



UNIVERZITET U NIŠU
PRIRODNO-MATEMATIČKI FAKULTET U NIŠU



Branislav D. Ivanov

**NAPREDNI GRADIJENTNI ALGORITMI ZA
REŠAVANJE PROBLEMA BEZUSLOVNE
OPTIMIZACIJE I NELINEARNIH MONOTONIH
SISTEMA JEDNAČINA VELIKIH DIMENZIJA**

DOKTORSKA DISERTACIJA

Niš, 2025.



UNIVERSITY OF NIŠ
FACULTY OF SCIENCES AND MATHEMATICS



Branislav D. Ivanov

**ADVANCED GRADIENT ALGORITHMS FOR
SOLVING UNCONSTRAINED OPTIMIZATION
PROBLEMS AND MONOTONIC NONLINEAR
SYSTEMS OF EQUATIONS OF LARGE
DIMENSIONS**

DOCTORAL DISSERTATION

Niš, 2025.

Подаци о докторској дисертацији

Ментор:	Проф. др Предраг С. Станимировић, редовни професор, Универзитет у Нишу, Природно-математички факултет у Нишу
Наслов:	Напредни градијентни алгоритми за решавање проблема безусловне оптимизације и нелинеарних монотоних система једначина великих димензија
Резиме:	Докторска дисертација представља нове ефикасне алгоритме за решавање проблема безусловне оптимизације и система нелинеарних монотоних једначина. Нови алгоритми су представници класа градијентних и коњуговано градијентних алгоритма. У дисертацији је представљено шест нових градијентних алгоритма: два из класе модификованих убрзаних градијентних метода, два из класе хибридних модификованих градијентних метода и два алгоритма као последица вишеструке употребе линијског претраживања у градијентним методама. Из класе коњуговано градијентних алгоритама представљен је један нови трочлани метод, два нова мешовита коњугована градијентна метода и два нова коњуговано градијентна метода Dai-Liao типа. Када је у питању решавање система нелинеарних монотоних једначина великих димензија представљен је нови метод Dai-Liao типа са побољшаним параметрима. Такође, у дисертацији су представљени градијентни алгоритми за решавање проблема безусловне оптимизације базирани на неутрософији. Урађена анализа конвергенције и нумеричка тестирања за нове итеративне шеме потврђују теоријски и практични значај нових градијентних и коњуговано градијентних метода и њихових алогоритама. Примена градијентних и коњуговано градијентних алгоритма дата је у 2D роботској контроли покрета, процесу рестаурације слика и регресионој анализи.
Научна област:	Рачунарске науке
Научна дисциплина:	Рачунарство и нумерички методи, Операциона истраживања
Кључне речи:	Нелинеарна оптимизација, математичко програмирање, алгоритми, градијентни методи, коњуговано градијентни методи, методи линијског претраживања, системи нелинеарних монотоних једначина, неутрософија
УДК:	004 + [005.31 + 519.6]
CERIF класификација:	P160 Статистика, операционо истраживање, програмирање, актуарска математика P170 Рачунарство, нумеричка анализа, системи, контрола
Тип лиценце Креативне заједнице:	CC BY-NC-ND

Data on Doctoral Dissertation

Doctoral Supervisor:	Predrag S. Stanimirović, Ph.D., full professor, University of Niš, Faculty of Sciences and Mathematics
Title:	Advanced gradient algorithms for solving unconstrained optimization problems and monotonic nonlinear systems of equations of large dimensions
Abstract:	The doctoral dissertation presents new efficient algorithms for solving unconstrained optimization problems and monotonic nonlinear systems of equations. The new algorithms are representatives of the classes of gradient and conjugate gradient algorithms. The dissertation presents six new gradient algorithms: two from the class of modified accelerated gradient methods, two from the class of hybrid modified gradient methods, and two algorithms as a consequence of the multiple use of line search in gradient methods. From the class of conjugate gradient algorithms, one new three-term method, two new mixed conjugate gradient methods, and two new conjugate gradient methods of the Dai-Liao type are presented. When it comes to solving systems of nonlinear monotonic equations of large dimensions, a new Dai-Liao type method with improved parameters is presented. Also, the dissertation presents gradient algorithms for solving unconstrained optimization problems based on neutrosophy. The convergence analysis and numerical testing of new iterative schemes confirm the theoretical and practical significance of new gradient and conjugate gradient methods and their algorithms. The application of gradient and conjugate gradient algorithms is given in 2D robotic motion control, image restoration processes, and regression analysis.
Scientific Field:	Computer science
Scientific Discipline:	Computer science and numerical methods, Operational research
Key Words:	Nonlinear optimization, mathematical programming, algorithms, gradient methods, conjugate gradient methods, line search methods, systems of monotone nonlinear equations, neutrosophy
UDC:	004 + [005.31 + 519.6]
CERIF Classification:	P160 Statistics, operations research, programming, actuarial mathematics P170 Computer science, numerical analysis, systems, control
Creative Commons License Type:	CC BY-NC-ND



ПРИРОДНО - МАТЕМАТИЧКИ ФАКУЛТЕТ НИШ

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР :	
Идентификациони број, ИБР :	
Тип документације, ТД :	монографска
Тип записа, ТЗ :	текстуални / графички
Врста рада, ВР :	докторска дисертација
Аутор, АУ :	Бранислав Д. Иванов
Ментор, МН :	Предраг С. Станимировић
Наслов рада, НР :	Напредни градијентни алгоритми за решавање проблема безусловне оптимизације и нелинеарних монотоних система једначина великих димензија
Језик публикације, ЈП :	српски
Језик извода, ЈИ :	енглески
Земља публикавања, ЗП :	Србија
Уже географско подручје, УГП :	Србија
Година, ГО :	2025.
Издавач, ИЗ :	ауторски репринт
Место и адреса, МА :	Ниш, Вишеградска 33.
Физички опис рада, ФО : (поглавља/страна/ цитата/табела/слика/графика/прилога)	9 поглавља / 233 страна / 195 референци / 51 табела / 40 слика / 27 алгоритма
Научна област, НО :	Рачунарске науке
Научна дисциплина, НД :	Рачунарство и нумерички методи, Операциона истраживања
Предметна одредница/Кључне речи, ПО :	Нелинеарна оптимизација, математичко програмирање, алгоритми, градијентни методи, коњуговано градијентни методи, методи линијског претраживања, системи нелинеарних монотоних једначина, неутрософија
УДК	004 + [005.31 + 519.6]
Чува се, ЧУ :	библиотека
Важна напомена, ВН :	

Извод, ИЗ:	Докторска дисертација представља нове ефикасне алгоритме за решавање проблема безусловне оптимизације и система нелинеарних монотоних једначина. Нови алгоритми су представници класа градијентних и коњуговано градијентних алгоритма. У дисертацији је презентовано шест нових градијентних алгоритма: два из класе модификованих убрзаних градијентних метода, два из класе хибридних модификованих градијентних метода и два алгоритма као последица вишеструке употребе линијског претраживања у градијентним методама. Из класе коњуговано градијентних алгоритама презентован је један нови трочлани метод, два нова мешовита коњугована градијентна метода и два нова коњуговано градијентна метода Dai-Liao типа. Када је у питању решавање система нелинеарних монотоних једначина великих димензија презентован је нови метод Dai-Liao типа са побољшаним параметрима. Такође, у дисертацији су презентовани градијентни алгоритми за решавање проблема безусловне оптимизације базирани на неутрософији. Урађена анализа конвергенције и нумеричка тестирања за нове итеративне шеме потврђују теоријски и практични значај нових градијентних и коњуговано градијентних метода и њихових алогоритама. Примена градијентних и коњуговано градијентних алгоритма дата је у 2D роботској контроли покрета, процесу рестаурације слика и регресионој анализи.
Датум прихватања теме, ДП:	08.11.2021.
Датум одбране, ДО:	
Чланови комисије, КО:	Председник:
	Члан:
	Члан:
	Члан:
	Члан, ментор:

Образац Q4.09.13 - Издање 1



ПРИРОДНО - МАТЕМАТИЧКИ ФАКУЛТЕТ НИШ

KEY WORDS DOCUMENTATION

Accession number, ANO :	
Identification number, INO :	
Document type, DT :	monograph
Type of record, TR :	textual / graphic
Contents code, CC :	doctoral dissertation
Author, AU :	Branislav D. Ivanov
Mentor, MN :	Predrag S. Stanimirović
Title, TI :	Advanced gradient algorithms for solving unconstrained optimization problems and monotonic nonlinear systems of equations of large dimensions
Language of text, LT :	Serbian
Language of abstract, LA :	English
Country of publication, CP :	Serbia
Locality of publication, LP :	Serbia
Publication year, PY :	2025.
Publisher, PB :	author's reprint
Publication place, PP :	Niš, Višegradaska 33.
Physical description, PD : (chapters/pages/ref./tables/pictures/graphs/appendixes)	9 chapters / 233 pages / 195 ref. / 51 tables / 40 pictures / 27 algorithms
Scientific field, SF :	Computer science
Scientific discipline, SD :	Computer science and numerical methods, Operational research
Subject/Key words, S/KW :	Nonlinear optimization, mathematical programming, algorithms, gradient methods, conjugate gradient methods, line search methods, systems of monotone nonlinear equations, neutrosophy
UC	004 + [005.31 + 519.6]
Holding data, HD :	library
Note, N :	

Abstract, AB :	The doctoral dissertation presents new efficient algorithms for solving unconstrained optimization problems and monotonic nonlinear systems of equations. The new algorithms are representatives of the classes of gradient and conjugate gradient algorithms. The dissertation presents six new gradient algorithms: two from the class of modified accelerated gradient methods, two from the class of hybrid modified gradient methods, and two algorithms as a consequence of the multiple use of line search in gradient methods. From the class of conjugate gradient algorithms, one new three-term method, two new mixed conjugate gradient methods, and two new conjugate gradient methods of the Dai-Liao type are presented. When it comes to solving systems of nonlinear monotonic equations of large dimensions, a new Dai-Liao type method with improved parameters is presented. Also, the dissertation presents gradient algorithms for solving unconstrained optimization problems based on neutrosophy. The convergence analysis and numerical testing of new iterative schemes confirm the theoretical and practical significance of new gradient and conjugate gradient methods and their algorithms. The application of gradient and conjugate gradient algorithms is given in 2D robotic motion control, image restoration processes, and regression analysis.	
Accepted by the Scientific Board on, ASB :	08.11.2021.	
Defended on, DE :		
Defended Board, DB :	President:	
	Member:	
	Member:	
	Member:	
	Member, Mentor:	

Образец Q4.09.13 - Издање 1

Sadržaj

Predgovor	5
1 Uvod	1
1.1 Pregled oznaka, definicija i teorema	1
1.2 Organizacija doktorske disertacije	7
2 Opšti gradijentni algoritmi i varijante linijskog pretraživanja	11
2.1 Opšti gradijentni algoritmi	11
2.2 Linijsko pretraživanje	15
2.2.1 Tačno linijsko pretraživanje	16
2.2.2 Netačno linijsko pretraživanje	17
3 Kvazi-Njutnovi metodi	21
3.1 Kvazi-Njutnova jednačina	21
3.2 Ažuriranje simetričnom matricom ranga 1	24
3.3 DFP ažuriranje	25
3.4 BFGS ažuriranje	26
3.5 Kvazi-Njutnovi metodi bazirani na konstantnoj aproksimaciji dijagonalne matrice	26
3.6 Gradijentni metodi ubrzani Picard-Mann hibridnim iterativnim procesom . .	30
4 Modifikovani gradijentni metodi	32
4.1 Modifikovani ubrzani gradijentni (MAGD) metod	32
4.2 Modifikovani poboljšani gradijentni (MSM) metod	33
4.3 Hibridni modifikovani gradijentni metodi	35
4.3.1 Hibridni modifikovani ubrzani gradijentni (HMAGD) metod	36
4.3.2 Hibridni modifikovani poboljšani gradijentni (HMSM) metod	37
4.4 Višestruka upotreba linijskog pretraživanja unazad u MSM metodi	39
4.5 Analiza konvergencije	42
4.6 Numerički rezultati	52
4.6.1 Numerički rezultati za MAGD, MSM, HMAGD i HMSM metode . . .	53
4.6.2 Numerički rezultati za TSM i DMSM metode	62

5	Konjugovani gradijentni metodi	73
5.1	Osnovne metode konjugovanog gradijenta	74
5.2	Dai–Liao konjugovani gradijentni metod i njegove varijante	78
5.3	Hibridni konjugovani gradijentni metodi	81
5.4	BFGS konjugovani gradijentni metodi	84
5.5	Tročlani H-BFGS-CG1 metod	84
5.5.1	Analiza konvergencije za H-BFGS-CG1 metod	85
5.5.2	Numerički rezultati za H-BFGS-CG1 metod	87
6	Napredni konjugovani gradijentni metodi	92
6.1	Mešoviti LS-CD konjugovano gradijentni metod	92
6.2	Mešoviti DHSDL–DLSDL konjugovano gradijentni metod	94
6.3	Konjugovano gradijentni metod Dai–Liao tipa sa poboljšanim parametrima .	94
6.3.1	Efikasan Dai-Liao konjugovano gradijentni metod	96
6.3.2	Modifikovani konjugovano gradijentni metod Dai-Liao tipa zasnovan na aproksimaciji Hesijanove matrice pomoću parametra ubrzanja . .	97
6.4	Analiza konvergencije	102
6.4.1	Analiza konvergencije MLSCD i MMDL metode	102
6.4.2	Analiza konvergencije EDL i MSMDL metode	106
6.5	Numerički rezultati i primena	118
6.5.1	Numerički rezultati za MLSCD i MMDL metodu	119
6.5.2	Numerički rezultati za EDL metod	122
6.5.3	Numerički rezultati za MSMDL metod	127
6.5.4	Primena MSMDL metode u 2D robotskoj kontroli pokreta	135
7	Konjugovani gradijentni metodi Dai-Liao tipa sa poboljšanim parametrima za rešavanje sistema nelinearnih monotonih jednačina velikih dimenzija	138
7.1	Dai–Liao projekivni metod	138
7.2	MSM Dai–Liao projekivni metod (MSMDLPM)	142
7.3	Analiza konvergencije	145
7.3.1	Globalna konvergencija	145
7.3.2	Analiza brzine konvergencije	152
7.4	Numerički rezultati i primena	153
7.4.1	Numeričkih rezultati za MSMDLPM algoritam	154
7.4.2	Primena MSMDLPM algoritma na probleme restauracija slika	162
8	Gradijentni algoritmi za rešavanje problema bezuslovne optimizacije bazi- rani na neutrosofiji	166
8.1	Neutrosofija i bezuslovna optimizacija	167
8.2	Fazi modifikovani SM metod	169
8.3	Fazi neutrosofski Dai–Liao konjugovano gradijentni metod	176
8.4	Analiza konvergencije	180

SADRŽAJ

8.4.1	Ispitivanje konvergencije za FMSM metod	180
8.4.2	Ispitivanje konvergencije za FDL metod	187
8.5	Numerički rezultati i primena	190
8.5.1	Numeričkih rezultati za FSM, FGD i FMSM metode	191
8.5.2	Rangiranje optimizacionih metoda	193
8.5.3	Primena fazi metoda optimizacije u regresionoj analizi	198
8.5.4	Numeričkih rezultati za FDL metod	201
9	Zaključna razmatranja i predlozi za buduća istraživanja	206
	Spisak algoritama, tabela i slika	209
	Literatura	214
	Biografija autora	230

Predgovor

Oduvek su naučnici kroz svoj naučni rad težili postizanju boljih rezultata u iterativnim procesima. Određivanje i podešavanje vrednosti parametara u svakom iterativnom koraku direktno utiče na iterativni proces, a samim tim i na postizanju manjih ili većih vrednosti funkcije cilja. Postoje brojni matematički algoritmi (metode) za pronalaženje vrednosti parametara u iterativnim procesima za koje funkcija cilja postiže optimalnu vrednost.

U iterativnim procesima se kreće od početne aproksimacije, a zatim korak po korak, preko vrednosti ciljne funkcije ide se ka traženoj optimalnoj vrednosti funkcije (najvećoj ili najmanjoj). Iterativni algoritmi mogu imati mali broj iteracija koji brzo dovode do tačnog rešenja, ali koji su računski skupi (zahtevni), jer se u svakoj iteraciji za pronalaženje gradijenta izračunava Hesijan. U cilju smanjenja broja računskih operacija ili u nedostatku podataka drugog reda, prelazi se na odgovarajuće modifikovane algoritme (metode).

Istraživanja predložene doktorske disertacije zasnovana su na teoriji nelinearne optimizacije, numeričkoj analizi i pronalaženju novih efikasnih algoritama za uspešno rešavanje problema bezuslovne optimizacije i sistema nelinearnih monotonih jednačina velikih dimenzija. Gradijentni metodi, odnosno konjugovano gradijentni metodi kao naučna grana operacionih istraživanja i primenjene matematike su našli veliku primenu, kako u raznim naučnim disciplinama, tako i u inženjerstvu, ekonomiji (dizajniranju portfolija za investicije), vojnoj i avio industriji, a u poslednje vreme u obnavljanju (restauraciji) slika, dekodiranju retkih signala i u robotici. Predmet ove disertacije su optimizacioni problemi koji ne sadrže ograničenja i čija je ciljna funkcija nelinearna. Dakle, osnovni fokus je na stvaranju efikasnih algoritama za rešavanje problema bezuslovne optimizacije i sistema nelinearnih monotonih jednačina velikih dimenzija.

Ova naučna oblast poslednjih decenija privlači posebnu pažnju istraživača zbog dobro razvijene teorije numeričke optimizacije i savremenih računarskih mogućnosti. Savremena računarska tehnika je omogućila eksperimentalno generisanje i testiranje problema bezuslovne optimizacije i sistema nelinearnih monotonih jednačina velikih dimenzija. Osnovni problem u implementaciji metoda optimizacije je računaska složenost koja time ograničava primenu metode na probleme određenih dimenzija.

Istraživanja predložene doktorske disertacije su usmerena ka algoritmima koji će biti značajno računski jeftiniji, ali će i dalje određivati (pronaziti) optimalnu vrednost funkcije cilja sa datim stepenom odstupanja. Pristup koji ćemo istraživati se zasniva na kombinaciji ideja iz numeričke optimizacije uz odgovarajuće izmene i kombinovanjem sa drugim naučnim

oblastima u cilju smanjenja broja iteracija, broja evaluacija funkcije cilja, procesorskog vremena (CPU vreme), odnosno vreme izvršavanja algoritma i očuvanja konvergencije.

Konkretno, doktorska disertacija se bavi gradijentnim i konjugovano gradijentnim metodama kao dve najznačajnije klase metoda za rešavanje optimizacionih problema bez ograničenja. Gradijentne i konjugovano gradijentne metode služe kao motivacija i baza za dalje razvijanje drugih algoritama (metoda) čija bi primena dala korisne efekte na zadatke nelinearnog programiranja.

Doktorska disertacija je nastala na osnovu originalnih objavljenih naučnih radova [83, 84, 85, 86, 87, 151, 152, 153, 154] i na osnovu rezultata koji se u tezi prvi put pojavljuju. Polazeći od postojećih opšte prihvaćenih gradijentnih i konjugovano gradijentnih metoda za rešavanje problema bezuslovne optimizacije i sistema nelinearnih monotonih jednačina velikih dimenzija u disertaciji su predloženi i opisani novi algoritmi specifične forme za rešavanje problema iz ove oblasti.

Sama struktura doktorske disertacije je organizovana u devet poglavlja. Prvo poglavlje sadrži pregled osnovnih oznaka, definicija, lema i teorema neophodnih za praćenje problema bezuslovne optimizacije.

U drugom poglavlju su prikazani bazični gradijentni algoritmi, podela linijskog pretraživanja i pregled najpopularnijih metoda linijskog pretraživanja.

Treće poglavlje posvećeno je kvazi-Njutnovim metodama i njihovim modifikacijama. Takođe, u okviru ovog poglavlja dat je pregled najznačajnijih metoda baziranih na konstantnoj aproksimaciji dijagonalne matrice, kao i primena Pikard-Manovog hibridnog iterativnog procesa u procesu ubrzanja gradijentnih metoda.

Četvrto poglavlje je bazirano na naučnim rezultatima iz radova [85, 86]. U okviru ovog poglavlja je prezentovano šest novih gradijentnih algoritama i to: dva algoritma iz klase modifikovanih ubrzanih gradijentnih metoda, dva iz klase hibridnih modifikovanih gradijentnih metoda i dva algoritma kao posledica višestruke upotrebe linijskog pretraživanja u gradijentnim metodama.

Peto poglavlje je posvećeno konjugovano gradijentnim metodama i njihovim modifikacijama za rešavanje problema bezuslovne optimizacije. Pored pregleda klasičnih konjugovano gradijentnih metoda, dat je pregled Dai-Liao klase konjugovano gradijentnih metoda. Takođe, u ovom poglavlju dat je pregled hibridnih konjugovano gradijentnih metoda kao i deo naučnih rezultata iz [151]. Tačnije, prezentovan je novi tročlani H-BFGS-CG1 algoritam.

Naučni rezultati iz radova [83, 87, 151] prezentovani su u šestom poglavlju. U ovom poglavlju prezentovana su dva nova hibridna konjugovano gradijentna metoda iz [151] i dve nove varijante Dai-Liao konjugovano gradijentnih metoda iz [83, 87]. Predloženi metod iz [83] je uspešno primenjen u rešavanju problema optimizacije koji nastaje u 2D robotskoj kontroli pokreta. Glavni rezultat dobijen u ovom istraživanju je korelacija između kvazi-Njutnovog i konjugovano gradijentnog pristupa. Konkretno, u ovom istraživanju proučavana je primena kvazi-Njutnovog metoda u poboljšanju konjugovano gradijentnih algoritama Dai-Liao tipa.

Sedmo poglavlje predstavlja ubrzani modifikovani Dai–Liao projektivni metod za rešavanje sistema nelinearnih monotonih jednačina sa primenom u problemima uklanjanja zamućenja slika. Kombinacija MSM metode i Dai–Liao projektivnog metoda (MSMDLPM metod), koji je definisan u ovom poglavlju, predstavlja naučne rezultate iz [84] i može se smatrati proširenjem Dai–Liao metode za probleme neograničene optimizacije u kombinaciji sa projektivnim metodom i parametrom ubrzanja koji predstavlja aproksimaciju Hesijanove matrice. Pod blagim pretpostavkama dokazana je globalna i linearna konvergencija MSM Dai–Liao projektivnog metoda.

MSM Dai–Liao projektivni metod je upoređen sa sličnim metodama i numerički rezultati potvrđuju efikasnost MSMDLPM metode u odnosu na posmatrane metode za rešavanje sistema nelinearnih monotonih jednačina. Takođe, MSMDLPM algoritam se uspešno primenjuje na probleme uklanjanja zamućenja slika. Numerički rezultati dobijeni prilikom uklanjanja zamućenja slike korišćenjem MSMDLPM algoritma upoređeni su sa numeričkim rezultatima posmatranih algoritma i može se zaključiti da je MSMDLPM algoritam postigao bolje rezultate u svim eksperimentima po svakom indikatoru (broj iteracija, CPU vreme, SSIM, SNR i PSNR). Eksperimenti su sprovedeni na različitim uzorcima slika i dobijeni rezultati jasno pokazuju da je MSMDLPM pristup efikasniji u odnosu na testirane metode.

Uopšteno govoreći, ovo poglavlje pokazuje da hibridizacija kvazi–Njutnovih metoda sa veoma popularnom klasom konjugovano gradijentnih metoda može pokrenuti efikasne metode optimizacije. Definisanje hibridnih metoda između brojnih i raznovrsnih kvazi–Njutnovih metoda i konjugovano gradijentnih metoda je otvorena oblast za buduća istraživanja.

Osmo poglavlje posvećeno je gradijentnim algoritmima baziranim na neutrosofiji za rešavanje problema bezuslovne optimizacije, odnosno predstavljeni su naučni rezultati iz [153, 154]. U ovom poglavlju primena neutrosofije je usmerena na dva pravca i oba su u cilju unapređenja metoda za rešavanje problema bezuslovne optimizacije.

Prvi pravac primene neutrosofije je usmeren na poboljšanju metode linijskog pretraživanja za rešavanje problema bezuslovne optimizacije, a drugi pravac je usmeren na poboljšanju Dai–Liao konjugovano gradijentnog metoda.

Poznato je da su iteracije za rešavanje problema bezuslovne optimizacije bazirane na dužini koraka koja se održuje pomoću netačnog linijskog pretraživanja. Ovako određena dužina koraka omogućava upravo dovoljno smanjenje vrednosti funkcije cilja. Međutim, postoje određene mogućnosti za dalja prilagođavanja na osnovu ponašanja funkcije cilja. U tom slučaju, cilj je da se uz upotrebu dodatnih parametara za dužinu koraka izvrši uticaj na poboljšanju iteracija za rešavanje problema bezuslovne optimizacije. Jedan od ovih parametara je parametar ubrzanja, koji se određuje na osnovu Tejlorovog razvoja funkcije cilja. Drugi parametar (fazi parametar), definisan je u ovom poglavlju korišćenjem neutrosofske logike i ponašanja funkcije cilja u dve uzastopne iteracije. Fazi parametar nastaje kao izlaz iz odgovarajuće definisanog neutrosofskog logičkog sistema i može se koristiti u različitim metodama opadajućeg gradijenta kao korektivna veličina dužine koraka.

Drugi pravac primene neutrosofije je usmeren ka novom pristupu određivanja parametara

t u Dai–Liao konjugovano gradijentnim iteracijama. Novi pristup se bazira na pronalaženju odgovarajućih vrednosti za nenegativni parametar t u Dai–Liao metodi korišćenjem neutrosofske logike. Vrednost parametra t se određuje na osnovu odgovarajuće definisanog neutrosofskog logičkog sistema i može se koristiti u različitim metodama Dai–Liao klase konjugovano gradijentnih metoda. Ugrađivanjem neutrosofije u Dai–Liao konjugovano gradijentni metod, definisana je nova Dai–Liao konjugovano gradijentna iteracija za rešavanje velikih problema bezuslovne optimizacije pod nazivom fazi neutrosofski Dai–Liao (FDL) konjugovano gradijentni metod.

Poslednje, deveto poglavlje je zaključna razmatranja i predlozi za buduća istraživanja. U okviru ovog poglavlja data je kratka analiza prethodno izloženih rezultata i predlog i pravac budućih istraživanja.



Koristim priliku da izrazim veliku i iskrenu zahvalnost svom mentoru dr Predragu Stanimiroviću, redovnom profesoru Prirodno-matematičkog fakulteta u Nišu, Univerziteta u Nišu, koji je svojim savetima, idejama i sugestijama pomogao izradu moje doktorske disertacije. Posebno mu se zahvaljujem na razumevanju i moralnoj podršci za vreme mog studiranja na doktorskim studijama kao i na iskustvu koje sam stekao tokom našeg zajedničkog rada.

Svoju zahvalnost dugujem dr Gradimiru Milovanoviću, akademiku i redovnom članu Srpske akademije nauke i umetnosti, dr Dragiši Stanujkiću, redovnom profesoru Tehničkog fakulteta u Boru, dr Predragu Rajkoviću, redovnom profesoru Mašinskog fakulteta u Nišu i dr Marku Miladinoviću, vanrednom profesoru Prirodno-matematičkog fakulteta u Nišu na korisnim sugestijama i primedbama koje su značajno uticale na kvalitet ove disertacije.

Želim da se zahvalim svojoj porodici, supruzi Dušici i majci Slavici na strpljenju i razumevanju, a zatim priljateljima i kolegama na pruženoj podršci.

Poglavlje 1

Uvod

Predmet istraživanja disertacije je rešavanje problema bezuslovne optimizacije i sistema nelinearnih monotonih jednačina velikih dimezija. Za rešavanje problema bezuslovne optimizacije koriste se algoritmi ubrzanog opadajućeg gradijenta, hibridnog ubrzanog opadajućeg gradijenta, algoritmi bazirani na Dai–Liao konjugovano gradijentnom metodu, kao i algoritmi koji kombinuju neutrosofsku logiku sa algoritmima ubrzanog opadajućeg gradijenta i Dai–Liao konjugovano gradijentnog metoda.

Kada je u pitanju rešavanje sistema nelinearnih monotonih jednačina velikih dimezija koriste se algoritmi koji kombinuju projektni metod sa pravcima pretraživanja Dai–Liao tipa konjugovanih gradijenata. Pomenuti algoritmi pripadaju klasi iterativnih postupaka za rešavanje sistema nelinearnih jednačina.

Razumevanje teorijske osnove na kojoj se zasniva projektni metod i teorije iterativnih postupaka za rešavanje problema bezuslovne optimizacije i sistema nelinearnih jednačina zahteva odgovarajuće znanje iz matematičke, numeričke i funkcionalne analize, kao i iz linearne algebre.

Da bi se obezbedilo jednostavnije praćenje doktorske disertacije, ovo poglavlje je posvećeno pregledu oznaka, definicija, lema i teorema iz publikacija [112, 121, 152, 162].

1.1 Pregled oznaka, definicija i teorema

U ovom odeljku uvodimo osnovne oznake, definicije i teoreme iz matematičke, numeričke i funkcionalne analize i iz linearne algebre koje su neophodne za razumevanje doktorske disertacije.

Skup prirodnih brojeva označen je sa $\mathbb{N}$, a sa $\mathbb{R}$ označen je skup realnih brojeva, dok $\mathbb{R}^n$ označava n -dimenzionalan vektorski prostor realnih brojeva, a $\mathbb{R}^{m \times n}$ označava prostor realnih matrica sa m vrsta i n kolona.

Vektor (uređena n -torka) $\mathbf{x} \in \mathbb{R}^n$ predstavlja se kao $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ i poistovećuje

sa matricom kolone

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}, \quad x_1, \dots, x_n \in \mathbb{R}.$$

Ako je $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{x}^T$ označava transponovani vektor vektora $\mathbf{x}$, dok ako je $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{A}^T$ označava transponovanu matricu matrice $\mathbf{A}$. Sa $\mathbf{I}_n$ označava se $n \times n$ jedinična matrica.

Definicija 1.1.1. Neka je $\mathbb{V}$ vektorski prostor nad $\mathbb{R}$. Preslikavanje $\langle \cdot, \cdot \rangle : \mathbb{V} \times \mathbb{V} \rightarrow \mathbb{R}$ je **skalarni proizvod** na prostoru $\mathbb{V}$ ako ima sledeće osobine:

- $(\forall \mathbf{x} \in \mathbb{V}) \quad \langle \mathbf{x}, \mathbf{x} \rangle \geq 0$ (nenegativnost);
- $\langle \mathbf{x}, \mathbf{x} \rangle = 0$ ako i samo ako $\mathbf{x} = 0$ (definitnost);
- $(\forall \mathbf{x}, \mathbf{y} \in \mathbb{V}) \quad \langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{y}, \mathbf{x} \rangle$ (simetričnost);
- $(\forall \mathbf{x}, \mathbf{y} \in \mathbb{V}) \quad (\forall \lambda \in \mathbb{R}) \quad \langle \lambda \mathbf{x}, \mathbf{y} \rangle = \lambda \langle \mathbf{x}, \mathbf{y} \rangle$ (homogenost);
- $(\forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{V}) \quad \langle \mathbf{x}, \mathbf{y} + \mathbf{z} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle + \langle \mathbf{x}, \mathbf{z} \rangle$ (aditivnost).

U disertaciji posmatra se Euklidski prostor n -dimenzionalnih realnih vektora, ukoliko nije drugačije naglašeno, u kojem je **skalarni proizvod** vektora $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ definisan sa

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y} = \sum_{i=1}^n x_i y_i.$$

Definicija 1.1.2. Neka je $\mathbb{V}$ vektorski prostor nad $\mathbb{R}$. Funkcija $\| \cdot \| : \mathbb{V} \rightarrow \mathbb{R}$ se naziva **norma** na prostoru $\mathbb{V}$ ako i samo ako zadovoljava sledeće uslove:

- $(\forall \mathbf{x} \in \mathbb{V}) \quad \|\mathbf{x}\| \geq 0$ (nenegativnost);
- $\|\mathbf{x}\| = 0$ ako i samo ako $\mathbf{x} = 0$;
- $(\forall \mathbf{x} \in \mathbb{V}) \quad (\forall \lambda \in \mathbb{R}) \quad \|\lambda \mathbf{x}\| = |\lambda| \cdot \|\mathbf{x}\|$ (homogenost);
- $(\forall \mathbf{x}, \mathbf{y} \in \mathbb{V}) \quad \|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$ (nejednakost trougla).

Prirodni način da se generiše norma na $\mathbb{R}^n$ jeste da se uzme skalarni proizvod $\langle \cdot, \cdot \rangle$ na $\mathbb{R}^n$ i definiše norma na sledeći način:

$$\|\mathbf{x}\|_2 = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{\mathbf{x}^T \mathbf{x}} = \sqrt{\sum_{i=1}^n x_i^2}, \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

Ovako definisana norma se naziva l_2 -norma ili **Euklidska norma**. Euklidska norma koja se podrazumeva u $\mathbb{R}^n$ jeste $\|\cdot\|_2$, te će indeks 2 najčešće biti izostavljen u disertaciji.

Bitna nejednakost koja povezuje skalarni proizvod dva vektora sa njihovom normom je takozvana Cauchy–Schwarz-ova nejednakost.

Lema 1.1.1. (Cauchy–Schwarz-ova nejednakost) *U Euklidskom n -dimenzionalnom prostoru za svako $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ važi nejednakost*

$$|\mathbf{x}^T \mathbf{y}| \leq \|\mathbf{x}\| \cdot \|\mathbf{y}\|. \quad (1.1)$$

Lema 1.1.2. *U Euklidskom n -dimenzionalnom prostoru za svako $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ važi nejednakost*

$$\mathbf{x}^T \mathbf{y} \leq \frac{1}{2}(\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2). \quad (1.2)$$

Definicija 1.1.3. Preslikavanje $\|\cdot\| : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$ se naziva **matrična norma** ako i samo ako zadovoljava sledeće osobine:

- $\|\mathbf{A}\| \geq 0$ za svaku matricu $\mathbf{A} \in \mathbb{R}^{m \times n}$ (nenegativnost);
- $\|\mathbf{A}\| = 0$ ako i samo ako $\mathbf{A} = \mathbf{0}$;
- $\|\lambda \mathbf{A}\| = |\lambda| \cdot \|\mathbf{A}\|$ za svaku matricu $\mathbf{A} \in \mathbb{R}^{m \times n}$ i $\lambda \in \mathbb{R}$ (homogenost);
- $\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$ za svaku matricu $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$ (nejednakost trougla).

Definicija 1.1.4. Matrica $\mathbf{A} \in \mathbb{R}^{n \times n}$ je **pozitivno semidefinitna** ako

$$\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0 \text{ za svaki vektor } \mathbf{x} \in \mathbb{R}^n,$$

a **pozitivno definitna** ako

$$\mathbf{x}^T \mathbf{A} \mathbf{x} > 0 \text{ za svaki vektor } \mathbf{x} \in \mathbb{R}^n \text{ i } \mathbf{x} \neq \mathbf{0}.$$

Definicija 1.1.5. Ako važi $\mathbf{A} \mathbf{x} = \lambda \mathbf{x}$, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{x} \neq \mathbf{0}$ i $\lambda \in \mathbb{R}$, tada je $\mathbf{x}$ **sopstveni vektor** matrice $\mathbf{A}$ koji odgovara **sopstvenoj vrednosti** λ .

Lema 1.1.3. (Rejljeva nejednakost) *Ako je $\mathbf{A} \in \mathbb{R}^{n \times n}$ simetrična matrica, a $\|\cdot\|$ norma vektora u $\mathbb{R}^n$, tada važe nejednakosti*

$$\lambda_{\min} \|\mathbf{x}\|^2 \leq \mathbf{x}^T \mathbf{A} \mathbf{x} \leq \lambda_{\max} \|\mathbf{x}\|^2, \quad \mathbf{x} \in \mathbb{R}^n, \quad (1.3)$$

gde je $\lambda_{\min}$ najmanja, a $\lambda_{\max}$ najveća sopstvena vrednost matrice $\mathbf{A}$.

Definicija 1.1.6. Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$. Vektor

$$\nabla f(x_1, \dots, x_n) = \left[\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right]^T$$

naziva se **gradijent** funkcije f u tački $(x_1, \dots, x_n) \in \mathbb{P}$.

Definicija 1.1.7. Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ dva puta neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$. Matricu

$$\nabla^2 f(\mathbf{x}) = \left[\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} \right]_{i,j=1,\dots,n}, \mathbf{x} \in \mathbb{P}$$

nazivamo **Hesijanova matrica** ili **Hesijan** funkcije f u tački $\mathbf{x} \in \mathbb{P}$.

Radi jednostavnijeg i lakšeg praćenja doktorske disertacije usvojene su sledeće oznake:

$$\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x}), \quad G(\mathbf{x}) = \nabla^2 f(\mathbf{x}), \quad \mathbf{g}_k = \nabla f(\mathbf{x}_k) \text{ i } G_k = \nabla^2 f(\mathbf{x}_k), \quad (1.4)$$

gde $\nabla f(\mathbf{x})$ označava gradijent, a $\nabla^2 f(\mathbf{x})$ Hesijan (ili Hesijanova matrica) funkcije f . Radi jednostavnosti, oznaka f_k će ukazivati na $f(\mathbf{x}_k)$.

Pored usvojenih oznaka uvode se i oznake koje se koriste u gradijentnim metodama optimizacije za definisanje odgovarajućih izraza (obrazaca) za izračunavanje i ažuriranje:

$$\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k, \quad \mathbf{y}_k = \mathbf{g}_{k+1} - \mathbf{g}_k. \quad (1.5)$$

Takođe, uvode se i sledeće oznake, koje se odnose na Hesijanovu matricu funkcije cilja:

- B_k označava aproksimaciju Hesijanove matrice G_k u k -toj iteraciji.
- H_k označava aproksimaciju inverza Hesijanove matrice G_k^{-1} u k -toj iteraciji.

Definicija 1.1.8. Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$ i neka je vektor $\mathbf{d} \in \mathbb{R}^n$ takav da važi

$$\nabla f(\mathbf{x})^T \mathbf{d} < 0 \quad (1.6)$$

Tada se vektor $\mathbf{d} \in \mathbb{R}^n$ naziva **pravac opadanja** funkcije f u tački $\mathbf{x} \in \mathbb{P}$.

Na osnovu Tejlorove aproksimacije funkcije f u tački $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{d}$ važi da je

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k + \alpha \mathbf{d}) \approx f(\mathbf{x}_k) + \alpha \mathbf{g}_k^T \mathbf{d} + o(\alpha). \quad (1.7)$$

Tada očigledno važi

$$\exists \delta > 0, \text{ takvo da je } f(\mathbf{x}_k + \alpha \mathbf{d}) < f(\mathbf{x}_k), \quad \forall \alpha \in (0, \delta),$$

ako i samo ako je $\mathbf{d}$ pravac opadanja funkcije f u tački $\mathbf{x}_k$, odnosno ako je uslov (1.6) ispunjen.

Teorema 1.1.1. *Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$. Tada gradijent funkcije f postoji i nezavisan je od izbora koordinatnog sistema u prostoru $\mathbb{R}^n$. U svakoj tački u kojoj je gradijent funkcije f različit od $\vec{0} = (0, 0, \dots, 0)$ važi da gradijent funkcije predstavlja smer najbržeg rasta skalarnog polja određenog funkcijom f .*

Napomena 1.1.1. *Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$. U svakoj tački u kojoj je gradijent funkcije različit od $\vec{0} = (0, 0, \dots, 0)$ važi da vektor koji je suprotan gradijentu funkcije predstavlja **smer najbržeg opadanja** skalarnog polja određenog funkcijom f .*

Definicija 1.1.9. Tačka $\mathbf{x}^*$ naziva se **lokalni minimum** funkcije f ako postoji $\delta > 0$ tako da je

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \text{ za svako } \mathbf{x} \in \mathbb{R}^n \text{ za koje je } \|\mathbf{x} - \mathbf{x}^*\| < \delta.$$

Definicija 1.1.10. Tačka $\mathbf{x}^*$ naziva se **strogi lokalni minimum** funkcije f ako postoji $\delta > 0$ tako da je

$$f(\mathbf{x}^*) < f(\mathbf{x}) \text{ za svako } \mathbf{x} \in \mathbb{R}^n \text{ za koje je } \|\mathbf{x} - \mathbf{x}^*\| < \delta \text{ i } \mathbf{x} \neq \mathbf{x}^*.$$

Definicija 1.1.11. Tačka $\mathbf{x}^*$ naziva se **globalni minimum** funkcije f ako je

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \text{ za svako } \mathbf{x} \in \mathbb{R}^n.$$

Definicija 1.1.12. Tačka $\mathbf{x}^*$ naziva se **strogi globalni minimum** funkcije f ako je

$$f(\mathbf{x}^*) < f(\mathbf{x}) \text{ za svako } \mathbf{x} \in \mathbb{R}^n \text{ i } \mathbf{x} \neq \mathbf{x}^*.$$

Teorema 1.1.2. (Potreban uslov optimalnosti prvog reda) *Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$. Ako je $\mathbf{x}^* \in \mathbb{P}$ rešenje minimizacionog problema*

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}), \tag{1.8}$$

tada važi da je

$$\nabla f(\mathbf{x}^*) = 0.$$

Definicija 1.1.13. Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$ i neka je $\mathbf{x}^* \in \mathbb{P}$. Tada je $\mathbf{x}^*$ **stacionarna tačka** funkcije f ako je $\nabla f(\mathbf{x}^*) = 0$.

Teorema 1.1.3. (Potreban uslov optimalnosti drugog reda) *Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ dva puta neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$ i neka je $\mathbf{x}^*$ stacionarna tačka. Ako je $\mathbf{x}^*$ tačka lokalnog minimuma funkcije f na $\mathbb{P}$, tada je $\nabla^2 f(\mathbf{x}^*)$ pozitivno semidefinitna matrica.*

Teorema 1.1.4. (Dovoljan uslov optimalnosti drugog reda) *Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ dva puta neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P}$. Ako je $\nabla f(\mathbf{x}^*) = 0$ i ako je $\nabla^2 f(\mathbf{x}^*)$ pozitivno definitna (pozitivno semidefinitna) matrica, tada je $\mathbf{x}^* \in \mathbb{P}$ tačka strogog lokalnog minimuma funkcije f .*

Uslov koji garantuje da je posmatrano rešenje optimalno naziva se **dovoljan uslov optimalnosti**. Rešenje je optimalno ukoliko ono zadovoljava uslov optimalnosti koji je istovremeno i potreban i dovoljan uslov, i obrnuto, samo optimalno rešenje zadovoljava potrebne i dovoljne uslove za optimalnost.

Teorema 1.1.5. *Neka je funkcija f dva puta neprekidno diferencijabilna funkcija i definisana na celom skupu $\mathbb{R}^n$. Pretpostavimo da je $\nabla^2 f(\mathbf{x}) \geq 0$ za svako $\mathbf{x} \in \mathbb{R}^n$. Neka je $\mathbf{x}^*$ stacionarna tačka funkcije f . Tada je $\mathbf{x}^*$ tačka globalnog minimuma funkcije f .*

Definicija 1.1.14. Funkcija $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ je **konveksna funkcija** ako je njen domen $\mathbb{P}$ konveksan skup i za svako $\mathbf{x}, \mathbf{y} \in \mathbb{P}$ važi

$$f(\alpha \mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y}), \text{ za svako } \alpha \in (0, 1).$$

Funkcija f je **strogo (striktno) konveksna funkcija** ako za svako $\mathbf{x}, \mathbf{y} \in \mathbb{P}$, $\mathbf{x} \neq \mathbf{y}$ važi

$$f(\alpha \mathbf{x} + (1 - \alpha)\mathbf{y}) < \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y}), \text{ za svako } \alpha \in (0, 1).$$

Funkcija f je **uniformno konveksna funkcija** ako postoji konstanta $c > 0$ takva da za svako $\mathbf{x}, \mathbf{y} \in \mathbb{P}$, važi

$$f(\alpha \mathbf{x} + (1 - \alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1 - \alpha)f(\mathbf{y}) - \frac{1}{2}c\alpha(1 - \alpha)\|\mathbf{x} - \mathbf{y}\|^2, \text{ za svako } \alpha \in (0, 1).$$

Definicija 1.1.15. Funkcija $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ je **monotona funkcija**, ako za svako $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ važi

$$(F(\mathbf{x}) - F(\mathbf{y}))^T(\mathbf{x} - \mathbf{y}) \geq 0.$$

Definicija 1.1.16. Funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ je **Lipšic neprekidna funkcija**, ako postoji konstanta $L > 0$ takva da je

$$\|f(\mathbf{x}) - f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \text{ za svako } \mathbf{x}, \mathbf{y} \in \mathbb{R}^n.$$

Ako je neka funkcija Lipšic neprekidna, tada je i uniformno neprekidna na $\mathbb{R}^n$.

Teorema 1.1.6. *Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$, $\mathbb{P} \subset \mathbb{R}^n$ gde je $\mathbb{P}$ neprazan konveksan skup i neka je $\mathbf{x}^* \in \mathbb{P}$ tačka lokalnog minimuma funkcije f :*

- *Ako je f konveksna funkcija onda je $\mathbf{x}^*$ globalni minimum funkcije f .*
- *Ako je f strogo (striktno) konveksna funkcija onda je $\mathbf{x}^*$ jedinstveni globalni minimum funkcije f .*

Definicija 1.1.17. Niz vektora $\{\mathbf{x}_k\} \subset \mathbb{R}^n$ konvergira ka vektoru $\mathbf{x}^*$, ako važi

$$\lim_{k \rightarrow \infty} \|\mathbf{x}_k - \mathbf{x}^*\| = 0.$$

Definicija 1.1.18. Niz vektora $\{\mathbf{x}_k\} \subset \mathbb{R}^n$ naziva se **Košijev niz**, ako važi

$$\lim_{i,j \rightarrow \infty} \|\mathbf{x}_i - \mathbf{x}_j\| = 0,$$

to jest, za dato $\delta > 0$, postoji ceo broj M , takav da važi $\|\mathbf{x}_i - \mathbf{x}_j\| < \delta$, za svako $i, j > M$.

Definicija 1.1.19. Neka iterativni niz $\{\mathbf{x}_k\} \subset \mathbb{R}^n$ konvergira ka vektoru $\mathbf{x}^*$. Ako postoje pozitivne konstante α i β , gde je $\alpha \geq 1$, a β nezavisna od broja k , tako da važi

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|^\alpha} = \beta,$$

tada niz $\{\mathbf{x}_k\}$ ima red konvergencije jednak α .

Specijalno:

- Ako je $\alpha = 1$ i $\beta \in (0, 1)$, onda niz $\{\mathbf{x}_k\}$ konvergira **linerano**.
- Ako je $\alpha = 1$ i $\beta = 0$ ili $1 < \alpha < 2$ i $\beta > 0$, onda niz $\{\mathbf{x}_k\}$ konvergira **superlinerano**.
- Ako je $\alpha = 2$, onda niz $\{\mathbf{x}_k\}$ konvergira **kvadratno**.

Definicija 1.1.20. Iterativni metod za određivanje rešenja $\mathbf{x}^* \in \mathbb{R}^n$ nelinearnog sistema

$$F(\mathbf{x}) = 0, \quad \mathbf{x} \in \mathbb{R}^n, \tag{1.9}$$

gde je $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ neprekidno preslikavanje je lokalno (linearno, superlinearno, kvadratno) konvergentan ako niz iteracija $\{\mathbf{x}_k\}$ konvergira ka $\mathbf{x}^*$ (linearno, superlinearno, kvadratno), pri čemu je početna iteracija dovoljno blizu rešenja $\mathbf{x}^*$.

1.2 Organizacija doktorske disertacije

Doktorska disertacija je podeljena u osam poglavlja, gde je svako poglavlje podeljeno na odeljke, a odeljci se sastoje od pododeljaka.

- U trenutnom, prvom i uvodnom poglavlju dat je pregled oznaka, definicija, lema i teorema neophodnih za izučavanje problema bezuslovne optimizacije.
- Drugo poglavlje ove disertacije posvećeno je opštim gradijentnim algoritmima i varijantama linijskog pretraživanja.

- U prvom odeljku ovog poglavlja biće definisani osnovni pojmovi vezani za nelinearnu optimizaciju i opšti pregled gradijentnih algoritama.
- U drugom odeljku biće opisano linijsko pretraživanje, njegova podela na tačno i netačno linijsko pretraživanje, kao i opis često upotrebljavanih linijskih pretraživanja za rešavanje problema bezuslovne optimizacije.
- Treće poglavlje doktorata posvećeno je kvazi–Njutnovim metodama.
 - U prvom odeljku je opisana kvazi–Njutnova jednačina i kvazi–Njutnov metod.
 - Naredna tri odeljka opisuju modele matrica koji predstavljaju aproksimacije inverza Hesijana i zadovoljavaju kvazi–Njutnovu jednačinu.
 - U petom odeljku data je podela i pregled najpoznatijih metoda baziranih na konstantnoj aproksimaciji dijagonalne matrice.
 - Na kraju ovog poglavlja u poslednjem odeljku data je primena Pikard–Manovog hibridnog iterativnog procesa u procesu ubrzanja gradijentnih metoda, kao i pregled najznačajnijih metoda ove klase.
- Četvrto poglavlje modifikovani gradijentni metodi sadrži originalne rezultate koji predstavljaju deo glavnih rezultata doktorske disertacije.
 - U prva tri odeljka dati su modifikovani gradijentni metodi, odnosno u prvom odeljku dat je modifikovani ubrzani gradijentni MAGD metod, u drugom modifikovani poboljšani gradijentni MSM metod i u trećem odeljku date su hibridne varijante HMAGD i HMSM modifikovanih gradijentnih metoda.
 - Četvrti odeljak posvećen je višestrukoj upotrebi linijskog pretraživanja unazad u MSM metodi, odnosno TMSM i DMSM metodama.
 - U petom odeljku data je analiza konvergenције predstavljenih metoda.
 - Poslednji odeljak prikazuje numeričke rezultate i ispituje efikasnost predloženih metoda.
- Peto poglavlje konjugovani gradijentni metodi prikazuje pregled konjugovano gradijentnih metoda za bezuslovnu optimizaciju, njihovu klasifikaciju po kategorijama i naučne rezultate.
 - U prvom odeljku prikazane su osnovne metode konjugovanih gradijenata.
 - Drugi odeljak je posvećen klasi Dai–Liao metoda.
 - Hibridne konjugovano gradijentne metode opisane su u trećem odeljku.
 - Četvrti odeljak prikazuje BFGS konjugovano gradijentni metod i njegove varijante.
 - U petom odeljku dati su naučni rezultati ostvareni urađenim istraživanjem, odnosno predstavljen je tročlani hibridni BFGS–CG metod, a u pododeljcima data je analiza konvergenције i numerički rezultati predstavljenog metoda.

- Šesto poglavlje napredni konjugovani gradijentni metodi sadrži originalne rezultate koji predstavljaju jedan deo glavnih rezultata doktorske disertacije. Prezentovana su dva hibridna konjugovano gradijentna metoda, od kojih je jedan nastao kombinacijom klasičnih konjugovano gradijentnih metoda, a drugi kombinacijom modifikovanih Dai–Liao konjugovano gradijentnih metoda. Takođe, prezentovane su i dve nove varijante Dai–Liao konjugovano gradijentnih metoda.
 - U prvom odeljku dat je hibridni konjugovano gradijentni MLSCD metod nastao kombinacijom klasičnih LS i CD konjugovano gradijentnih metoda.
 - Drugi odeljak prezentuje hibridni konjugovano gradijentni MMDL metod baziran na DHSDL i DLSDL konjugovano gradijentnim metodama Dai–Liao tipa.
 - U okviru trećeg odeljka opisane su dve nove varijante (EDL i MSMDL) konjugovano gradijentnih metoda Dai–Liao tipa.
 - Analiza konvergencije predstavljenih metoda data je u četvrtom odeljku.
 - U petom odeljku ispitana je efikasnost predloženih metoda i dati su numerički rezultati. U okviru ovog odeljka u poslednjem pododeljku data je primena MSMDL metode u 2D robotskoj kontroli pokreta.
- Sedmo poglavlje konjugovano gradijentni metodi Dai–Liao tipa sa poboljšanim parametrima za rešavanje sistema nelinearnih monotonih jednačina velikih dimenzija sadrži originalne rezultate koji se odnose na rešavanje sistema nelinearnih monotonih jednačina i koji predstavljaju deo glavnih rezultata. Predstavljen je postupak za rešavanje sistema nelinearnih monotonih jednačina velikih dimenzija koji kombinuje projekтивni metod i pravac pretraživanja Dai–Liao tipa konjugovanih gradijenata.
 - U prvom odeljku dat je Dai–Liao projekтивni metod.
 - Drugi odeljak posvećen je MSM Dai–Liao projekтивnom metodu (MSMDLPM). U ovom odeljku uspostavljena je veza između MSM i DLPM metode sa ciljem da se stvori efikasan metod za rešavanje sistema nelinearnih monotonih jednačina velikih dimenzija.
 - Analiza konvergencije za MSMDLPM metod data je u trećem odeljku.
 - Poslednji odeljak prikazuje numeričke rezultate, ispituje efikasnost MSMDLPM metoda i daje primenu MSMDLPM metoda u restauraciji slika.
- Osmo poglavlje gradijentni algoritmi za rešavanje problema bezuslovne optimizacije bazirani na neutrosofiji sadrži originalne rezultate koji predstavljaju deo glavnih rezultata doktorske disertacije. U ovom poglavlju predstavljeni su FMSM i FDL metod.
 - U prvom odeljku ovog poglavlja biće definisani osnovni pojmovi vezani za neutrosofiju i ideja o povezivanju neutrosofije sa gradijentnim algoritmima.
 - Drugi odeljak posvećen je fazi modifikovanom SM metodu (FMSM), gde je istražena primena neutrosofske logike u određivanju dodatne dužine koraka u metodama za rešavanje problema bezuslovne optimizacije.

- Treći odeljak posvećen je fazi neutrosofskom Dai–Liao konjugovano gradijentnom metodu (FDL), gde je istražena primena neutrosofske logike u poboljšanju performansi Dai–Liao metoda.
- Analiza konvergencije za FMSM i FDL metode data je u četvrtom odeljku.
- Poslednji odeljak prikazuje numeričke rezultate i primenu. U okviru ovog odeljka u pododeljcima dati su numerički rezultati, rangiranje optimizacionih metoda i primena fazi metoda optimizacije u regresionoj analizi.
- Deveto poglavlje zaključna razmatranja i predlozi za buduća istraživanja sadrži pregled i analizu prethodno izloženih rezultata u doktorskoj disertaciji, kao i predlog i smer budućih istraživanja.

Poglavlje 2

Opšti gradijentni algoritmi i varijante linijskog pretraživanja

U ovom poglavlju analiziraju se gradijentni algoritmi za bezuslovnu optimizaciju i varijante linijskog pretraživanja. Gradijentni algoritmi za bezuslovnu optimizaciju baziraju se na izvodima ciljne funkcije i mogu se podeliti u dve klase i to: gradijentni algoritmi prvog reda i gradijentni algoritmi drugog reda. Pod gradijentnim algoritmima prvog reda podrazumevaju se iterativne šeme koje u svojim formulacijama sadrže samo prvi izvod ciljne funkcije. Najpoznatiji među gradijentnim algoritmima prvog reda je Košijev (*Cauchy*) metod opadajućeg gradijenta. Iterativne šeme koje u svojim formulacijama sadrže i prvi i drugi izvod ciljne funkcije, ili neke njihove aproksimacije, nazivaju se gradijentni algoritmi drugog reda. Najpoznatiji među gradijentnim algoritmima ove klase je Njutnov (*Newton*) metod.

U nastavku poglavlja će biti opisani opšti gradijentni algoritmi, podela linijskog pretraživanja i detaljan opis tačnog i netačnog linijskog pretraživanja, kao i opis često upotrebljanih linijskih pretraživanja kao što su: linijsko pretraživanje unazad (*eng. backtracking*), Goldštajnovo (*Goldstein*) i Volf–Pauelovo (*Wolfe–Powell*) linijsko pretraživanje.

2.1 Opšti gradijentni algoritmi

U ovom odeljku biće dat opšti pregled gradijentnih algoritama kao i definicije i osobine metoda za rešavanje nelinearnog problema bezuslovne optimizacije

$$\min f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n, \quad (2.1)$$

gde je ciljna funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i čiji minimum želimo da odredimo. Opšte iterativno pravilo za rešavanje problema (2.1) počinje od početne aproksimacije $\mathbf{x}_0 \in \mathbb{R}^n$ i generiše niz $\{\mathbf{x}_k, k \geq 0\}$ koristeći opštu iterativnu šemu

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k, \quad k \geq 0, \quad (2.2)$$

gde je dužina koraka α_k pozitivan realan parametar određen linijskim pretraživanjem, $\mathbf{x}_k$ je poslednja generisana iterativna tačka, $\mathbf{x}_{k+1}$ je trenutna iterativna tačka i $\mathbf{d}_k$ je odgovarajući pravac pretraživanja. Opšta klasa algoritama oblika (2.2) poznata je kao **algoritmi linijskog pretraživanja**. Ovi algoritmi zahtevaju pronalaženje pravca pretraživanja $\mathbf{d}_k \in \mathbb{R}^n$ i dužine koraka $\alpha_k \in \mathbb{R}$.

Prema tome, odgovarajući opadajući pravac pretraživanja $\mathbf{d}_k$ mora biti određen na osnovu **uslova opadanja** (*eng. descent condition*)

$$\mathbf{g}_k^T \mathbf{d}_k < 0, \quad \forall k \in \mathbb{N} \cup \{0\}. \quad (2.3)$$

Primarni izbor za opadajući pravac je $\mathbf{d}_k = -\mathbf{g}_k$, koji svodi opštu iterativnu šemu (2.2) u iterativnu šemu **opadajućeg gradijenta** (*eng. gradient descent*)

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k. \quad (2.4)$$

Ovakav izbor pravca pretraživanja dovodi nas do klase algoritama (metoda) poznate kao **algoritmi opadajućeg gradijenta** ili jednostavnije **gradijentni algoritmi**. Jedna od negativnih karakteristika gradijentnih algoritama je relativno česta pojava, takozvanog, cik-cak fenomena, koji inicira veoma sporu konvergenciju algoritama opadajućeg gradijenta prema optimalnoj tački ili čak divergenciju [121].

U ovom poglavlju razmatraju se gradijentni algoritmi koji zadovoljavaju uslov opadanja (2.3). Ako postoji konstanta $c > 0$ takva da

$$\mathbf{g}_k^T \mathbf{d}_k \leq -c \|\mathbf{g}_k\|^2, \quad \forall k \in \mathbb{N} \cup \{0\}, \quad (2.5)$$

gde c ne zavisi od broja iteracija k , onda se kaže da vektor $\mathbf{d}_k$ zadovoljava **dovoljan uslov opadanja** (*eng. sufficient descent condition*).

Prednosti i nedostaci algoritama opadajućeg gradijenta mogu se sumirati na sledeći način:

1. Algoritmi opadajućeg gradijenta su globalno konvergentni, tj. konvergiraju ka lokalnom minimumu bez obzira na početnu tačku.
2. Većina algoritama optimizacije prelazi na pravilo opadajućeg gradijenta kada ne ostvare dovoljan napredak u konvergenciji.
3. Konvergencija je linearna i obično veoma spora.
4. Numerički algoritmi opadajućeg gradijenta često nisu konvergentni.

Drugi važan pravac pretraživanja koji treba izdvojiti je **Njutnov pravac** za gradijentne algoritme drugog reda. Njutnov pravac pretraživanja $\mathbf{d}_k = -G_k^{-1} \mathbf{g}_k$ je dobijen iz Tejlоровog razvoja drugog reda, pod pretpostavkom da je Hesijan G_k pozitivno definitna matrica.

Njutnov metod (bez linijskog pretraživanja) za minimizaciju ciljne funkcije $f : \mathbb{R}^n \rightarrow \mathbb{R}$ je definisan pomoću kvadratne aproksimacije $\Phi(\mathbf{d})$ ciljne funkcije $f(\mathbf{x}_{k+1})$:

$$\Phi(\mathbf{d}) := f(\mathbf{x}_k + \mathbf{d}) \approx f(\mathbf{x}_k) + \mathbf{g}_k^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T G_k \mathbf{d}, \quad (2.6)$$

pri čemu je $\mathbf{d} = \mathbf{x}_{k+1} - \mathbf{x}_k$.

Potrebno je da se funkcija $\Phi(\mathbf{d})$ minimizira. Nije teško uočiti da je izvod poslednjeg člana desne strane jednakosti (2.6) jednak

$$0 = \Phi'(\mathbf{d}) = \mathbf{g}_k + G_k \mathbf{d}. \quad (2.7)$$

Iz čega sledi da je rešenje $\mathbf{d}_k = \min_{\mathbf{d}}(\Phi(\mathbf{d}))$ dato sa

$$\mathbf{d}_k = -G_k^{-1} \mathbf{g}_k.$$

Dakle, Njutnov metod je definisan sa iterativnim pravilom

$$\mathbf{x}_{k+1} = \mathbf{x}_k - G_k^{-1} \mathbf{g}_k. \quad (2.8)$$

Jasno je da Njutnov pravac predstavlja pravac opadanja, zato što važi relacija

$$\mathbf{g}_k^T \mathbf{d}_k = -\mathbf{g}_k^T G_k^{-1} \mathbf{g}_k < 0, \quad \forall k \in \mathbb{N} \cup \{0\}, \quad (2.9)$$

pod uslovom da je G_k pozitivno definitna matrica.

Opšti algoritam Njutnovog metoda sa iterativnim pravilom (2.8) je dat Algoritmom 2.1.1.

Algoritam 2.1.1 Njutnov metod

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0 \in \mathbb{R}^n$ i tolerancija $0 < \varepsilon \ll 1$.

1: $k := 0$.

2: **while** $\|\mathbf{g}_k\| > \varepsilon$ **do**

3: Rešiti jednačinu $G_k \mathbf{d}_k = -\mathbf{g}_k$ po $\mathbf{d}_k$, odnosno odrediti pravac pretraživanja $\mathbf{d}_k$.

4: Izračunati $\mathbf{x}_{k+1} := \mathbf{x}_k + \mathbf{d}_k$.

5: $k := k + 1$.

6: **end while**

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

Na osnovu svega navedenog možemo zaključiti da Njutnov metod (Algoritam 2.1.1) predstavlja unapređenje algoritma opadajućeg gradijenta, zbog toga što osim informacije o gradijentu ciljne funkcije koristi i informaciju o Hesijanu.

Konvergencija Njutnovog metoda nije sigurna u slučajevima kada je početna tačka daleko od rešenja, jer nije sigurno da je matrica G_k pozitivno definitna, niti da je Njutnov pravac pretraživanja $\mathbf{d}_k = -G_k^{-1} \mathbf{g}_k$ pravac opadanja. Zbog toga se uvodi modifikacija dužine koraka

pravca pretraživanja kojom se garantuje globalna konvergencija.

Linijsko pretraživanje je globalna strategija koja se može primeniti na Njutnov metod i iz te primene dobija se **Njutnov metod sa linijskim pretraživanjem** za koga bi mogli da garantujemo globalnu konvergenciju. Treba naglasiti da je stopa konvergencije Njutnovog metoda kvadratna isključivo kada niz dužina koraka $\{\alpha_k\}$ teži jedinici.

Njutnov metod sa linijskim pretraživanjem koristi odgovarajuću dužinu koraka α_k u (2.8) sa ciljem da obezbedi globalnu stabilnost, a iteracije su oblika

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k G_k^{-1} \mathbf{g}_k, \quad (2.10)$$

pri čemu se dužina koraka α_k određuje nekom od metoda linijskog pretraživanja.

Iteracije oblika (2.10) dovode nas do Njutnovog metoda sa linijskim pretraživanjem (Algoritam 2.1.2).

Algoritam 2.1.2 Njutnov metod sa linijskim pretraživanjem

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0 \in \mathbb{R}^n$ i tolerancija $0 < \varepsilon \ll 1$.

1: $k := 0$.

2: **while** $\|\mathbf{g}_k\| > \varepsilon$ **do**

3: Rešiti jednačinu $G_k \mathbf{d}_k = -\mathbf{g}_k$ po $\mathbf{d}_k$, odnosno odrediti pravac pretraživanja $\mathbf{d}_k$.

4: Linijsko pretraživanje: Izračunati $\alpha_k > 0$ koristeći tačno ili netačno linijsko pretraživanje tako da važi

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) = \min_{\alpha \geq 0} f(\mathbf{x}_k + \alpha \mathbf{d}_k).$$

5: Izračunati $\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{d}_k$.

6: $k := k + 1$.

7: **end while**

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

Njutnov metod ispoljava tri glavna nedostatka u praktičnim primenama:

1. Konvergencija (opadanje) je neizvesna ukoliko su iteracije oblika (2.8) započete previše daleko od lokalnog minimuma.
2. Neophodnost da se u svakoj iteraciji izračuna Hesijanova matrica i njen inverz. U nekim slučajevima izračunavanje Hesijana i njegovog inverza je vrlo komplikovano.
3. Glavni nedostatak Njutnovog metoda je mogućnost da Hesijan G_k nije pozitivno definitna matrica.

Navedeni nedostaci Njutnovog metoda prouzrokovali su brojne modifikacije koje se globalno mogu podeliti u dve velike grupe: **modifikovani Njutnovi metodi** i **kvazi-Njutnovi metodi**.

Modifikovani Njutnovi metodi navedene probleme rešavaju svođenjem Njutnovog metoda na metod opadajućeg gradijenta ili na osnovni gradijentni metod prvog reda, dok kvazi-Njutnovi metodi imaju za cilj da reše sve navedene nedostatke Njutnovog metoda. Prvi nedostatak se prevazilazi uzimanjem odgovarajuće definisane pozitivno definitne matrice B_k koja aproksimira Hesijanovu matricu G_k ili odgovarajuće definisane pozitivno definitne matrice H_k koja aproksimira Hesijanov inverz G_k^{-1} , a zatim izvršavanjem linijskog pretraženja u svakoj iteraciji. Za datu početnu tačku $\mathbf{x}_0 \in \mathbb{R}^n$ i simetričnu pozitivno definitnu matricu H_0 , pravac pretraživanja u k -toj iteraciji kvazi-Njutnovog metoda je definisan sa $\mathbf{d}_k = -H_k \mathbf{g}_k$, gde je H_k simetrična pozitivno definitna matrica.

Kvazi-Njutnovi i modifikovani Njutnovi metodi spadaju u najmoćnije metode za rešavanje problema bezuslovne optimizacije i primenjene su u mnogim problemima nelinearne optimizacije. Pregled kvazi-Njutnovih metoda za rešavanje nelinearnih problema najmanjih kvadrata razmatran je u [117, 152]. Metode optimizacije su našle brojne primene u mehanici fluida [61], slobodnom površinskom toku i kontaktu sa čvrstim telom [25], pronalaženju optimalne putanje za avione ili robotske ruke [15, 83], dizajniranju investicionalnog portfolija [15] i u mnogim drugim oblastima. Takođe, data je primena algoritma optimizacije u veštačkim neuronskim mrežama [101]. Za više detalja o metodama optimizacije pogledajte [121].

2.2 Linijsko pretraživanje

Da bi se postigla globalna konvergencija iterativnih metoda, potrebna je odgovarajuća dužina koraka α_k . Pretpostavimo da je određen pravac pretraživanja $\mathbf{d}_k$. Dužinu koraka α_k određujemo iz uslova da funkcija $f(\mathbf{x}_k)$ ostvari što je moguće veće smanjenje vrednosti funkcije po pravcu pretraživanja $\mathbf{d}_k$. U tom cilju definisana je pomoćna jednoargumentna funkcija $\Phi : \mathbb{R} \rightarrow \mathbb{R}$ koja zavisi od dužine koraka α i koja je definisana na sledeći način

$$\Phi(\alpha) := f(\mathbf{x}_k + \alpha \mathbf{d}_k), \quad \alpha \geq 0, \quad (2.11)$$

pri čemu je dužina koraka definisana nakon optimizacije jednodimenzionalnog problema

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) = \min_{\alpha > 0} \Phi(\alpha). \quad (2.12)$$

Primenom iterativnog postupka (2.2), najlogičnije je izabrati novu tačku tako da dužina koraka α_k smanjuje vrednost funkcije cilja:

$$\Phi(\alpha_k) = f(\mathbf{x}_{k+1}) < \Phi(0) = f(\mathbf{x}_k). \quad (2.13)$$

Metod koji u svakom iterativnom koraku zahteva ispunjenje uslova (2.13), odnosno smanjenje vrednosti ciljne funkcije, definiše iteracije koje se u svakom koraku približavaju minimumu date funkcije. Ovako definisan metod pripada klasi metoda **monotonog linijskog pretraživanja** (eng. *monotone line search*).

Problem nalaženja dužine koraka kod metoda linijskog pretraživanja dovodi do dve glavne strategije linijskog pretraživanja i to: **tačno linijsko pretraživanje** (*eng. exact line search*) i **netačno linijsko pretraživanje** (*eng. inexact line search*).

Na prvi pogled više obećava tačno linijsko pretraživanje, jer može dati najprecizniji minimum. Međutim, tačno linijsko pretraživanje se u praksi ne primenjuje, jer je potpuna minimizacija funkcije Φ u svakoj iteraciji vrlo zahtevna operacija. Zbog toga se u praksi više koristi metod netačnog linijskog pretraživanja. Kod metoda netačnog linijskog pretraživanja se prilikom određivanja dužine koraka α_k ne ide do minimuma jednodimenzionalne funkcije $\Phi(\alpha)$, već se zahteva da budu zadovoljeni neki unapred zadati uslovi za funkciju $f(\mathbf{x}_k)$.

Za razliku od monotonom linijskog pretraživanja, u literaturi je poznato i **nemonotono linijsko pretraživanje** (*eng. non-monotonic line search*), kod koga nije izričito obavezno smanjivanje vrednosti funkcije cilja u svakoj iteraciji [69, 70, 71]. Iako nemonotone tehnike ne obezbeđuju približavanje minimumu funkciji u svakoj iteraciji, one se veoma često koriste u praktičnim primenama i imaju veoma dobra svojstva konvergencije. Nedavno je predložen veliki broj metoda nemonotonog linijskog pretraživanja, za više detalja pogledati [160, 164].

Metode netačnog linijskog pretraživanja se takođe dele na monotone i nemonotone metode tj. na metode koje dopuštaju da se vrednost ciljne funkcije $f(\mathbf{x}_k)$ kroz iteracije i poveća.

2.2.1 Tačno linijsko pretraživanje

Metod tačnog linijskog pretraživanja baziran je na rešavanju jednodimenzionog minimizacionog problema (za više detalja pogledati [146, 162])

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) = \min_{\alpha \geq 0} f(\mathbf{x}_k + \alpha \mathbf{d}_k). \quad (2.14)$$

Rešenje problema (2.14) predstavlja optimalnu dužinu koraka α_k u datoj iteraciji, pri čemu se podrazumeva da je određen ili odabran pravac pretraživanja $\mathbf{d}_k$. Ukoliko je moguće analitički odrediti minimum problema (2.14), u tom slučaju tehnika tačnog linijskog pretraživanja je pogodna za rešavanje postavljenog optimizacionog problema i daje dobra konvergentna svojstva. Nažalost, u praksi to nije uvek taj slučaj. Samim tim, primena tehnike tačnog linijskog pretraživanja postaje nepogodna, kako zbog nemogućnosti rešavanja jednodimenzionog problema (2.14), tako i zbog neefikasnosti ovog metoda. Neefikasnost metode proizilazi iz činjenice da je potrebno određeno procesorsko vreme za rešavanje problema (2.14) u svakoj iteraciji, a takođe potreban je i memorijski prostor. Međutim, metod tačnog linijskog pretraživanja ima veliki teorijski značaj. Ispitivanje konvergencije dovelo je do korisnih rezultata. Iako je cilj tehnike tačnog linijskog pretraživanja rešenje problema (2.14), koji se svodi na određivanje parametra dužine koraka α_k u tekućoj iteraciji, iz analize konvergencije ovog modela izvodi se zaključak da konvergencija metoda tačnog linijskog pretraživanja u mnogome zavisi od određivanja ili odabira pravca pretraživanja $\mathbf{d}_k$.

Opšti algoritam tačnog linijskog pretraživanja je dat Algoritmom 2.2.1.

Algoritam 2.2.1 Algoritam tačnog linijskog pretraživanja

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0 \in \mathbb{R}^n$ i tolerancija $0 < \varepsilon \ll 1$.

- 1: $k := 0$.
- 2: **while** $\|\mathbf{g}_k\| > \varepsilon$ **do**
- 3: Odrediti pravac pretraživanja $\mathbf{d}_k$.
- 4: Odrediti dužinu koraka $\alpha_k > 0$ tako da je ispunjen uslov

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) = \min_{\alpha \geq 0} f(\mathbf{x}_k + \alpha \mathbf{d}_k).$$

- 5: Izračunati $\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{d}_k$.
- 6: $k := k + 1$.
- 7: **end while**

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

Ako definišemo jednodimenzionu funkciju $\Phi : \mathbb{R} \rightarrow \mathbb{R}$ kao $\Phi(\alpha) = f(\mathbf{x}_k + \alpha \mathbf{d}_k)$, uzevši u obzir jednakost (2.14), očigledno važe relacije

$$\Phi(0) = f(\mathbf{x}_k), \Phi(\alpha) \leq \Phi(0).$$

Dok prvu stacionarnu tačku funkcije Φ definiše izraz

$$\alpha_k = \min\{\alpha \geq 0 \mid \nabla f(\mathbf{x}_k + \alpha \mathbf{d}_k)^T \mathbf{d}_k = 0\}. \quad (2.15)$$

Na kraju se može dati precizan zaključak da je metod tačnog linijskog pretraživanja određen relacijama (2.14) i (2.15), pri čemu relacija (2.14) definiše globalni minimum, a relacija (2.15) stacionarnu tačku funkcije Φ .

2.2.2 Netačno linijsko pretraživanje

U prethodnom odeljku opisan je metod tačnog linijskog pretraživanja, gde je naglašeno da se ne primenjuje u praksi, ali da ima veliki teorijski značaj. Takođe, ukoliko je tekuća iteracija daleko od optimalne tačke, nije efikasno tačno rešavati jednodimenzionalni problem

$$\alpha_k = \arg \min_{\alpha > 0} f(\mathbf{x}_k + \alpha \mathbf{d}_k). \quad (2.16)$$

Značaj metode sa netačnim linijskim pretraživanjem je višestruk. Sam postupak tačnog rešavanja jednodimenzione minimizacije (2.16) i vrlo česta kompleksnost ovog problema dovela je do potrebe za alterativnim rešenjem, a samim tim do nastanka netačnog linijskog pretraživanja. Pri tome, stopa konvergencije uglavnom nije smanjena u poređenju sa metodama tačnog linijskog pretraživanja. Mnogobrojnim istraživanjima je potvrđeno da konvergencija i red konvergencije metoda linijskog pretraživanja ne zavisi od toga da li se koristi tačno ili netačno linijsko pretraživanje.

Opšta iterativna šema u gradijentnim algoritmima za rešavanje problema bezuslovne optimizacije, bazirana na netačno linijsko pretraživanje, ista je kao i kod algoritama sa tačnim linijskim pretraživanjem i oblika je (2.2). Za pravac pretraživanja $\mathbf{d}_k$ može se odabrati bilo koji od predloženih izbora iz Odeljka 2.1 ovog poglavlja. Kako nam je fokus na rešavanju minimizacionog problema (2.16), važno je da odabrani pravac pretraživanja bude opadajući pravac pretraživanja, odnosno da zadovoljava uslov opadanja (2.3).

Za efikasnost algoritama sa netačnim linijskim pretraživanjem kao i za njihovu konvergenciju jednako je važan i parametar dužine koraka α_k , odnosno određivanje njegove optimalne dužine. Svi algoritmi sa netačnim linijskim pretraživanjem generalno imaju zajednički cilj, a to je izabrati dužinu koraka α_k tako da **dovoljno smanji** vrednost ciljne funkcije u svakoj iteraciji, odnosno da važi

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \ll f(\mathbf{x}_k), \quad \forall k \in \mathbb{N} \cup \{0\},$$

gde smo sa $\ll$ označili **dovoljno malu razliku** između posmatranih vrednosti.

Opšti algoritam netačnog linijskog pretraživanja je dat Algoritmom 2.2.2.

Algoritam 2.2.2 Algoritam netačnog linijskog pretraživanja

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0 \in \mathbb{R}^n$ i tolerancija $0 < \varepsilon \ll 1$.

- 1: $k := 0$.
- 2: **while** $\|\mathbf{g}_k\| > \varepsilon$ **do**
- 3: Odrediti pravac pretraživanja $\mathbf{d}_k$.
- 4: Odrediti dužinu koraka $\alpha_k > 0$ tako da važi uslov

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \ll f(\mathbf{x}_k).$$

- 5: Izračunati $\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{d}_k$.
- 6: $k := k + 1$.
- 7: **end while**

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

U nastavku su opisane varijante netačnog linijskog pretraživanja koje su dominantne u nelinearnoj optimizaciji i za cilj imaju određivanje što optimalnije dužine koraka α_k . Najčešće korišćene metode netačnog linijskog pretraživanja koje pripadaju klasi monotonog netačnog linijskog pretraživanja predložili su: Armijo (*Armijo*)[12], Goldštajn (*Goldstein*)[68], Volf (*Wolfe*)[171], Pael (*Powell*)[137], Flečer (*Fletcher*)[64] i mnogi drugi [6, 35, 73, 75, 146].

Za potrebe istraživanja u ovoj doktorskoj disertaciji najčešće je korišćen metod **linijsko pretraživanje unazad** (*eng. backtracking line search* ili *Armijo line search*) iz [6]. Pored pomenutog metoda u većini gradijentnih metoda, za izračunavanje dužine koraka α_k koristi se jedan od sledećih metoda netačnog linijskog pretraživanja: **Volfovo linijsko pretraživanje** (*eng. Wolfe line search*) razvijeno u [73, 75, 171, 172] ili **snažno Volfovo linijsko pretraživanje** (*eng. strong Wolfe line search*)[46].

Linijsko pretraživanje unazad (*backtracking*)

Linijsko pretraživanje unazad (*eng. backtracking line search*) bazirano je na Armija uslovima, i ima za cilj da odredi maksimalnu vrednost dužine koraka α_k tokom kretanja po pravcu vektora pretraživanja $\mathbf{d}_k$. Počinje sa relativno velikom procenom dužine koraka i iterativno smanjuje vrednost dužine koraka sve dok opada vrednost ciljne funkcije, u odnosu na lokalni gradijent ciljne funkcije. Neka su dati realni parametri $\sigma \in (0, \frac{1}{2})$, $\beta \in (0, 1)$ i $\alpha > 0$. Tada postoji najmanji nenegativni ceo broj m_k koji zadovoljava nejednačinu

$$f(\mathbf{x}_k + \beta^{m_k} \alpha \mathbf{d}_k) \leq f(\mathbf{x}_k) + \sigma \beta^{m_k} \alpha \mathbf{g}_k^T \mathbf{d}_k, \quad \alpha > 0. \quad (2.17)$$

Procedura za linijsko pretraživanje unazad predložena u [6] počinje od početne vrednosti $\alpha = 1$ i njene izlazne vrednosti su definisane tako da smanjuje funkciju cilja. Shodno tome, Algoritam 2.2.3 iz [155] se koristi u numeričkim eksperimentima kao implementacija netačnog linijskog pretraživanja koja definiše dužinu koraka α_k .

Algoritam 2.2.3 Linijsko pretraživanje unazad (*backtracking*)

Ulaz: Funkcija cilja $f(\mathbf{x})$, pravac pretraživanja $\mathbf{d}_k$, prethodna aproksimacija $\mathbf{x}_k$, i realni brojevi $0 < \sigma < 0.5$ i $0 < \beta < 1$.

- 1: $\alpha = 1$.
- 2: **while** $f(\mathbf{x}_k + \alpha \mathbf{d}_k) > f(\mathbf{x}_k) + \sigma \alpha \mathbf{g}_k^T \mathbf{d}_k$ **do**
- 3: $\alpha := \alpha \beta$.
- 4: **end while**

Izlaz: $\alpha_k = \alpha$.

Parametri $0 < \sigma < 0.5$ i $0 < \beta < 1$ igraju jako bitnu ulogu u praktičnoj primeni optimizacionih metoda i najčešće vrednosti parametara ne proističu iz teorijskih razmatranja nego iz praktičnih iskustava, odnosno iz numeričkih eksperimenata.

Goldštajnovo (*Goldstein*) linijsko pretraživanje

Da bi se obezbedilo dovoljno smanjenje ciljne funkcije, Goldštajnovo pravilo za netačno linijsko pretraživanje zahteva sledeće uslove:

$$f(\mathbf{x}_k + \alpha \mathbf{d}_k) \leq f(\mathbf{x}_k) + \rho \alpha \mathbf{g}_k^T \mathbf{d}_k \quad (2.18)$$

i

$$f(\mathbf{x}_k + \alpha \mathbf{d}_k) \geq f(\mathbf{x}_k) + (1 - \rho) \alpha \mathbf{g}_k^T \mathbf{d}_k, \quad (2.19)$$

gde je $0 < \rho < \frac{1}{2}$ i $\alpha > 0$. Uslovi (2.18) i (2.19) definišu Goldštajnovo pravilo za netačno linijsko pretraživanje.

Volf–Paueľovo (*Wolfe–Powel*) linijsko pretraživanje

Uslovi za Volfova linijska pretraživanja su dobro poznati i dati su u obliku

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \leq f(\mathbf{x}_k) + \eta \alpha_k \mathbf{g}_k^T \mathbf{d}_k, \quad (2.20)$$

$$\mathbf{g}(\mathbf{x}_k + \alpha_k \mathbf{d}_k)^T \mathbf{d}_k \geq \sigma_1 \mathbf{g}_k^T \mathbf{d}_k, \quad (2.21)$$

gde je $0 < \eta < \sigma_1 < 1$. Pored toga, uopšteni snažni Volfovi uslovi, koji su sastavljeni od (2.20) i

$$-\sigma_2 \mathbf{g}_k^T \mathbf{d}_k \geq g(\mathbf{x}_k + \alpha_k \mathbf{d}_k)^T \mathbf{d}_k \geq \sigma_1 \mathbf{g}_k^T \mathbf{d}_k \quad (2.22)$$

se često koriste, gde je $\sigma_1 > 0$. U slučaju $\sigma_1 = \sigma_2$, uopšteni snažni Volfovi uslovi se svode na snažne Volfove uslove, koji su sastavljeni od (2.20) i

$$|\mathbf{g}(\mathbf{x}_k + \alpha_k \mathbf{d}_k)^T \mathbf{d}_k| \leq -\sigma_1 \mathbf{g}_k^T \mathbf{d}_k. \quad (2.23)$$

Uslov (2.20) Volfovih uslova se naziva **Armijo uslov** i često se koristi odvojeno ili u obliku svojih varijanti.

Poglavlje 3

Kvazi-Njutnovi metodi

Značaj i uspeh Njutnovog metoda ogleda se u upotrebi Hesijana koji nam daje korisne informacije o zakrivljenosti ciljne funkcije. Međutim, upotreba Hesijana u Njutnovoj metodi predstavlja i njegovu slabost. Zbog toga što izračunavanje matrice Hesijana i njenog inverza u svakoj iteraciji može biti vrlo komplikovano, ili možda štaviše imamo slučaj da Hesijan nije analitički dostupan. Navedena slabost Njutnovog metoda dovela je do klase Njutnovih metoda koje koriste samo vrednosti ciljne funkcije i njenog gradijenta. U ovom poglavlju razmatramo kvazi-Njutnove metode koje imaju sličnu stopu konvergencije kao Njutnov metod, dok sa druge strane koriste vrednosti funkcije i gradijenata funkcije umesto matrice Hesijana i njenog inverza. Takođe, treba napomenuti da klasa kvazi-Njutnovi metodi pripada klasi gradijentnih metoda prvog reda, jer koristi samo prvi izvod ciljne funkcije.

Generalno većina problema koji se javljaju kod Njutnovih metoda prevazilaze se kod kvazi-Njutnovih metoda konstrukcijom niza aproksimacija Hesijana ili inverza Hesijana, pri čemu se zahteva da svaka od aproksimacija Hesijana ili inverza Hesijana zadovoljava kvazi-Njutnovu jednačinu koja garantuje dobru definisanost aproksimacija. U nastavku poglavlja je opisana kvazi-Njutnova jednačina i kvazi-Njutnov metod. Nakon toga opisani su modeli matrica koji predstavljaju aproksimacije inverza Hesijana i zadovoljavaju kvazi-Njutnovu jednačinu. Na kraju ovog poglavlja date su metode bazirane na konstantnoj aproksimaciji dijagonalne matrice, kao i primena Pikard-Manovog (*Picard-Mann*) hibridnog iterativnog procesa u procesu ubrzanja gradijentnih metoda.

3.1 Kvazi-Njutnova jednačina

Bazični Njutnov metod zahteva izračunavanje matrice drugog izvoda, međutim to se zaoobilazi u kvazi-Njutnovim metodama. Kao posledica toga, Njutnov metod je računski skup (zahtevan) metod i sporo se izračunavaju iterativni koraci, dok su kvazi-Njutnove metode računarski jeftine i brže se izračunavaju iterativni koraci. S druge strane, Njutnov metod zahteva manji broj iterativnih koraka i generiše precizniji put konvergencije od kvazi-Njutnovih metoda.

Osnovna ideja kvazi-Njutnovog metoda je da se definiše aproksimacija B_k Hesijana G_k , odnosno aproksimacija H_k inverza Hesijana G_k^{-1} . Pri čemu se očekuje da nizovi matrica B_k (odnosno H_k), zadovoljavaju određene uslove:

- matrice B_k i H_k su pozitivno definitne;
- pravac pretraživanja $\mathbf{d}_k = -B_k^{-1}\mathbf{g}_k$ (odnosno $\mathbf{d}_k = -H_k\mathbf{g}_k$) je pravac opadanja ciljne funkcije;
- izračunavanje matrica B_k i H_k nije zahtevno, odnosno računski skupo.

I na kraju postavlja se pitanje kako formirati niz matrica B_k (odnosno H_k)?

Da bi smo dali odgovor na ovo pitanje bavićemo se uslovima koje matrice B_k i H_k moraju da zadovolje. Neka je $f : \mathbb{P} \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna funkcija na otvorenom skupu $\mathbb{P} \subset \mathbb{R}^n$. Posmatramo kvadratnu aproksimaciju funkcije $f(\mathbf{x})$ u tački $\mathbf{x}_{k+1} \in \mathbb{P}$:

$$f(\mathbf{x}) \approx f(\mathbf{x}_{k+1}) + \mathbf{g}_{k+1}^T(\mathbf{x} - \mathbf{x}_{k+1}) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_{k+1})^T G_{k+1}(\mathbf{x} - \mathbf{x}_{k+1}), \quad k \in \mathbb{N}. \quad (3.1)$$

Diferenciranjem jednačine (3.1) po $\mathbf{x}$ dobija se

$$\mathbf{g}(\mathbf{x}) \approx \mathbf{g}_{k+1} + G_{k+1}(\mathbf{x} - \mathbf{x}_{k+1}), \quad k \in \mathbb{N}. \quad (3.2)$$

Zamenom $\mathbf{x} = \mathbf{x}_k$, $k \in \mathbb{N}$ u (3.2) i uvođenjem oznaka iz (1.5) dobija se

$$G_{k+1}^{-1}\mathbf{y}_k \approx \mathbf{s}_k. \quad (3.3)$$

U (3.3) važi znak jednakosti ukoliko je u pitanju kvadratna funkcija f , jer je Hesijanova matrica kvadratne ciljne funkcije f konstantna matrica G , to jest,

$$\mathbf{s}_k = G^{-1}\mathbf{y}_k \text{ ili } \mathbf{y}_k = G\mathbf{s}_k, \quad k \in \mathbb{N}. \quad (3.4)$$

Na osnovu relacija (3.3) i (3.4), zahteva se da aproksimacije H_{k+1} (odnosno B_{k+1}) inverznih matrica G_{k+1}^{-1} (odnosno matrica G_{k+1}) u kvazi-Njutnovom metodu zadovoljavaju naredne jednačine

$$H_{k+1}\mathbf{y}_k = \mathbf{s}_k, \quad (3.5)$$

odnosno

$$B_{k+1}\mathbf{s}_k = \mathbf{y}_k. \quad (3.6)$$

Jednačine (3.5) i (3.6) nazivaju se **kvazi-Njutnove jednačine** ili **kvazi-Njutnovi uslovi**, gde su $\mathbf{s}_k$ i $\mathbf{y}_k$ definisani u (1.5). Jednačina (3.5) definiše uslov koji treba da zadovolje aproksimacije inverza Hesijana H_{k+1} , dok jednačina (3.6) se odnosi na uslov koji treba da ispune aproksimacije Hesijana B_{k+1} . Iz jednačine (3.5) se može izvesti jednačina (3.6) i pri tome je jasno da je $H_{k+1} = B_{k+1}^{-1}$.

Ukoliko sa leve strane jednačinu (3.6) pomnožimo vektorom $\mathbf{s}_k^T$ dobija se

$$\mathbf{s}_k^T B_{k+1} \mathbf{s}_k = \mathbf{s}_k^T \mathbf{y}_k.$$

Iz ovog sledi da je matrica B_{k+1} pozitivno definitna ukoliko važi uslov

$$\mathbf{s}_k^T \mathbf{y}_k > 0. \quad (3.7)$$

Uslov (3.7) naziva se **uslov prevoja** (*eng. curvature condition*) za matrice H_{k+1} i B_{k+1} koje zadovoljavaju kvazi-Njutnove jednačine (3.5) i (3.6).

Opšti algoritam kvazi-Njutnovog metoda je dat Algoritmom 3.1.1.

Algoritam 3.1.1 Opšti algoritam kvazi-Njutnovog metoda

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0 \in \mathbb{R}^n$, početna aproksimacija inverza Hesijana $H_0 \in \mathbb{R}^{n \times n}$ i tolerancija $0 < \varepsilon \ll 1$.

1: $k := 0$.

2: **while** $\|\mathbf{g}_k\| > \varepsilon$ **do**

3: Odrediti pravac pretraživanja

$$\mathbf{d}_k = -H_k \mathbf{g}_k. \quad (3.8)$$

4: Izračunati $\alpha_k > 0$ koristeći tačno ili netačno linijsko pretraživanje.

5: Izračunati $\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{d}_k$.

6: Ažurirati H_k u H_{k+1} , tako da važi kvazi-Njutnova relacija (3.5).

7: $k := k + 1$.

8: **end while**

Izlaz: $\mathbf{x}_{k+1}$ i $f(\mathbf{x}_{k+1})$.

U Algoritmu 3.1.1 za inicijalnu vrednost matrice H_0 , kao jedna od mogućnosti, uzima se da je $H_0 = I_n$, gde je I_n jedinična matrica dimenzije n , pri čemu je n broj promenljivih ciljne funkcije f . U tom slučaju, prva iteracija je upravo iteracija opadajućeg gradijenta.

Ukoliko se kvazi-Njutnov metod primeni na jednačinu (3.6), odnosno ukoliko se umesto aproksimacije inverza Hesijanove matrice H_k koriste aproksimacije B_k Hesijanove matrice dolazimo do druge varijante Algoritma 3.1.1. U tom slučaju umesto početne aproksimacije inverza Hesijana $H_0 \in \mathbb{R}^{n \times n}$ koristi se aproksimacija $B_0 \in \mathbb{R}^{n \times n}$ Hesijanove matrice, a umesto Koraka **3.** i **6.** u Algoritmu 3.1.1 koriste se sledeći koraci:

- **Korak 3*:** Rešiti $B_k \mathbf{d}_k = -\mathbf{g}_k$ po $\mathbf{d}_k$ i na taj način odrediti pravac pretraživanja $\mathbf{d}_k$.
- **Korak 6*:** Ažurirati B_k u B_{k+1} tako da važi kvazi-Njutnova jednačina (3.6).

U nastavku sledi poređenje Njutnovog i kvazi-Njutnovih metoda:

- Kod kvazi-Njutnovog metoda, izborom matrica H_k i B_k koje zadovoljavaju uslove (3.5) i (3.6) zadržavaju i osobinu pozitivne definitnosti, dok se za matrice G_k i G_k^{-1} kod Njutnovog metoda to ne može tvrditi.

- Njutnov metod zahteva računanje vrednosti funkcije, gradijenta funkcije kao i matrice Hesijana i njenog inverza, dok kvazi-Njutnov metod zahteva samo računanje vrednosti funkcije i gradijenta funkcije. Neračunanje vrednosti matrice Hesijana i njenog inverza kod kvazi-Njutnovog metoda predstavlja najveću prednost kvazi-Njutnovog metoda u odnosu na Njutnov metod.
- Na kraju, navedimo i to da je kod Njutnovog metoda potrebno $O(n^3)$ operacija u svakoj iteraciji, dok je kod kvazi-Njutnovog metoda potrebno $O(n^2)$ operacija u svakoj iteraciji, gde je n broj promenljivih ciljne funkcije f .

Na osnovu svega navedenog očigledna je nadmoć kvazi-Njutnovog metoda nad Njutnovim metodom.

U nastavku ovog poglavlja razmotrimo neke od poznatih postupaka za ažuriranje matrica H_k (odnosno B_k) koji zadovoljavaju kvazi-Njutnovu jednačinu (3.5)(odnosno (3.6)).

3.2 Ažuriranje simetričnom matricom ranga 1

Neka je H_k aproksimacija inverza Hesijana G_k^{-1} u k -toj iteraciji. Ažuriranje H_k u H_{k+1} simetričnom matricom ranga 1 se definiše na sledeći način

$$H_{k+1} = H_k + E_k,$$

gde se prepostavlja da je E_k simetrična matrica ranga 1. Dakle,

$$H_{k+1} = H_k + \mathbf{u}_k \mathbf{v}_k^T, \quad (3.9)$$

pri čemu su $\mathbf{u}_k, \mathbf{v}_k \in \mathbb{R}^n$. Na osnovu kvazi-Njutnove jednačine (3.5) imamo

$$H_{k+1} \mathbf{y}_k = (H_k + \mathbf{u}_k \mathbf{v}_k^T) \mathbf{y}_k = \mathbf{s}_k,$$

odnosno

$$(\mathbf{v}_k^T \mathbf{y}_k) \mathbf{u}_k = \mathbf{s}_k - H_k \mathbf{y}_k. \quad (3.10)$$

Zaključak je da $\mathbf{u}_k$ mora biti u smeru $\mathbf{s}_k - H_k \mathbf{y}_k$. Pretpostavimo da je

$$\mathbf{s}_k - H_k \mathbf{y}_k \neq 0,$$

(u suprotnom bi H_k zadovoljavala kvazi-Njutnovu jednačinu) i da vektor $\mathbf{v}_k$ zadovoljava uslov $\mathbf{v}_k^T \mathbf{y}_k \neq 0$. Tada, na osnovu (3.9) i (3.10), sledi

$$H_{k+1} = H_k + \frac{1}{\mathbf{v}_k^T \mathbf{y}_k} (\mathbf{s}_k - H_k \mathbf{y}_k) \mathbf{v}_k^T. \quad (3.11)$$

Uslov da je aproksimacija inverza Hesijana H_{k+1} simetrična, zahteva da je vektor $\mathbf{v}_k$

definisan na sledeći način $\mathbf{v}_k = \mathbf{s}_k - H_k \mathbf{y}_k$, na osnovu čega se iz (3.11) dobija

$$H_{k+1} = H_k + \frac{(\mathbf{s}_k - H_k \mathbf{y}_k)(\mathbf{s}_k - H_k \mathbf{y}_k)^T}{(\mathbf{s}_k - H_k \mathbf{y}_k)^T \mathbf{y}_k}, \quad (3.12)$$

što predstavlja **ažuriranje simetričnom matricom ranga 1** (*eng. symmetric rank one update*).

Za opšte funkcije, Kon (*Conn*), Gould (*Gould*) i Toint (*Toint*) u [36] su dokazali da niz Hesijanovih aproksimacija generisan pomoću simetrične matrice ranga 1 konvergira pravom Hesijanu pod uslovom da su koraci uniformno linearno nezavisni.

3.3 DFP ažuriranje

DFP ažuriranje je prvi predložio Dejvidon (*Davidon*) u [55], a kasnije su ga popularizovali Flečer (*Fletcher*) i Pael (*Powell*) u [65]. DFP ažuriranje je poznato još i kao metod ažuriranja simetričnom matricom ranga 2, odnosno imamo da se H_{k+1} formira tako što se matrici H_k dodaju dve simetrične matrice, od kojih je svaka ranga 1:

$$H_{k+1} = H_k + a \mathbf{u}_k \mathbf{u}_k^T + b \mathbf{v}_k \mathbf{v}_k^T, \quad k \in \mathbb{N},$$

pri čemu su $\mathbf{u}_k, \mathbf{v}_k \in \mathbb{R}^n$, dok su a i b skalari koje je potreba odrediti. Zamenom H_{k+1} u kvazi-Njutnovoj jednačini (3.5) dobija se

$$H_k \mathbf{y}_k + a \mathbf{u}_k \mathbf{u}_k^T \mathbf{y}_k + b \mathbf{v}_k \mathbf{v}_k^T \mathbf{y}_k = \mathbf{s}_k. \quad (3.13)$$

Vektori $\mathbf{u}_k$ i $\mathbf{v}_k$ nisu jedinstveni, ali se mogu odrediti na sledeći način:

$$\mathbf{u}_k = \mathbf{s}_k \text{ i } \mathbf{v}_k = H_k \mathbf{y}_k.$$

Sada iz (3.13) određujemo veličine skalara a i b prema pravilima

$$a = \frac{1}{\mathbf{u}_k^T \mathbf{y}_k} = \frac{1}{\mathbf{s}_k^T \mathbf{y}_k}, \quad b = -\frac{1}{\mathbf{v}_k^T \mathbf{y}_k} = -\frac{1}{\mathbf{y}_k^T H_k \mathbf{y}_k},$$

a odavde dobijamo iterativnu formulu za DFP ažuriranje

$$H_{k+1}^{\text{DFP}} = H_k + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} - \frac{H_k \mathbf{y}_k \mathbf{y}_k^T H_k}{\mathbf{y}_k^T H_k \mathbf{y}_k}, \quad k \in \mathbb{N}. \quad (3.14)$$

Formulu (3.14) je predložio Dejvidon, a kasnije su razvili Flečer i Pael, tako da na osnovu početnih slova imena svojih autora sam metod je dobio naziv **DFP (Davidon-Fletcher-Powell) ažuriranje**.

3.4 BFGS ažuriranje

Jedna poznata i široko korišćena formula za ažuriranje je Broyden-Fletcher-Goldfarb-Shanno (BFGS) pravilo. Ažuriranje inverza Hesijana H_{k+1} je definisano kao rešenje problema

$$\min_H \|H - H_k\|, \text{ tako da je } H = H^T, H\mathbf{y}_k = \mathbf{s}_k. \quad (3.15)$$

Nesumljivo, BFGS ažuriranje je otvoreno i poznato kao

$$H_{k+1}^{\text{BFGS}} = (I - \rho_k \mathbf{s}_k \mathbf{y}_k^T) H_k^{\text{BFGS}} (I - \rho_k \mathbf{y}_k \mathbf{s}_k^T) + \rho_k \mathbf{s}_k \mathbf{s}_k^T, \quad \rho_k = \frac{1}{\mathbf{y}_k^T \mathbf{s}_k}, \quad (3.16)$$

odnosno

$$H_{k+1}^{\text{BFGS}} = H_k^{\text{BFGS}} - \frac{H_k \mathbf{y}_k \mathbf{s}_k^T + \mathbf{s}_k \mathbf{y}_k^T H_k}{\mathbf{s}_k^T \mathbf{y}_k} + \left(1 + \frac{\mathbf{y}_k^T H_k \mathbf{y}_k}{\mathbf{s}_k^T \mathbf{y}_k}\right) \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k}, \quad (3.17)$$

gde su $\mathbf{s}_k$ i $\mathbf{y}_k$ definisani u (1.5).

Formula za BFGS ažuriranje Hesijanove matrice može da se generiše korišćenjem Sherman-Morrison-Woodbury formule. Modifikacija ranga jedan $M = A + \mathbf{b}\mathbf{c}^*$ matrice $A \in \mathbb{C}^{m \times n}$ koristi dva vektora $\mathbf{b} \in \mathbb{C}^{m \times 1}$ i $\mathbf{c} \in \mathbb{C}^{n \times 1}$. Sherman-Morrison-ova formula uspostavlja relaciju između M^{-1} i A^{-1} na sledeći način [62]:

$$M^{-1} = A^{-1} - (1 + \mathbf{c}^* A^{-1} \mathbf{b})^{-1} A^{-1} \mathbf{b} \mathbf{c}^* A^{-1}. \quad (3.18)$$

Kao rezultat, dobija se sledeće ažuriranje za B_k :

$$B_{k+1}^{\text{BFGS}} = B_k^{\text{BFGS}} - \frac{B_k \mathbf{s}_k \mathbf{s}_k^T B_k}{\mathbf{s}_k^T B_k \mathbf{s}_k} + \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{s}_k^T \mathbf{y}_k}. \quad (3.19)$$

3.5 Kvazi-Njutnovi metodi bazirani na konstantnoj aproksimaciji dijagonalne matrice

Opšte iterativno pravilo za kvazi-Njutnove metode sa linijskim pretraživanjem je oblika

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k H_k \mathbf{g}_k, \quad (3.20)$$

gde se pretpostavlja da je B_k (odnosno $H_k = B_k^{-1}$) pozitivna definitivna aproksimacija matrice G_k (odnosno G_k^{-1}) [162]. Ažuriranje B_k u B_{k+1} (odnosno H_k u H_{k+1}) se definiše na osnovu kvazi-Njutnove jednačine (3.6) (odnosno (3.5)).

Prema Brezinskovo (Brezinski) klasifikaciji [28] ažuriranje matrica B_k može se podeliti u tri kategorije i to:

- Ažuriranje pomoću skalarnih matrica $B_k = \lambda_k I$.
- Ažuriranje pomoću dijagonalnih matrica $B_k = \text{diag}(\lambda_1, \dots, \lambda_n)$.
- Ažuriranje na osnovu određivanja matrice B_k u celosti.

Metode optimizacije koje su uključene u disertaciji bazirane su na najjednostavnijoj aproksimaciji Hesijana i njegovog inverza

$$B_k = \gamma_k I \approx G_k, \quad (3.21)$$

gde je I odgovarajuća $n \times n$ jedinična matrica, a $\gamma_k > 0$ je parametar. Takav izbor aproksimacija Hesijana i njegovog inverza vodi ka iterativnom pravilu

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k^{-1} \alpha_k \mathbf{g}_k, \quad (3.22)$$

gde se dužina koraka α_k određuje korišćenjem netačnog linijskog pretraživanja, a γ_k se određuje na osnovu Tejlorovog razvoja funkcije $f(\mathbf{x})$. Iteracije oblika (3.22) se u [86] nazivaju algoritmom (metodom) **poboljšanog opadajućeg gradijenta** (*eng. improved gradient descent* (IGD)).

Postoje brojne modifikacije iterativne šeme (3.22), neke od njih su date u radovima [126, 131, 155, 156]. Stanimirović i Miladinović u radu [155] su definisali SM (Stanimirović-Miladinović) metod sa iteracijama oblika (3.22) u kojima je korišćena skalarna $\gamma_k I$ aproksimacija Hesijana, gde je $\gamma_k = \gamma(\mathbf{x}_k, \mathbf{x}_{k-1}) > 0$ parametar. Iterativna šema u SM metodi je oblika

$$\mathbf{x}_{k+1}^{\text{SM}} = \mathbf{x}_k^{\text{SM}} - \alpha_k (\gamma_k^{\text{SM}})^{-1} \mathbf{g}_k^{\text{SM}}, \quad (3.23)$$

gde je $\gamma_k^{\text{SM}} > 0$ **parametar ubrzanja** određen korišćenjem Tejlorovog razvoja ciljne funkcije f u tački $\mathbf{x}_{k+1}^{\text{SM}}$ i definisan sledećom jednakošću:

$$\gamma_{k+1}^{\text{SM}} = 2\gamma_k^{\text{SM}} \frac{\gamma_k^{\text{SM}} [f(\mathbf{x}_{k+1}^{\text{SM}}) - f(\mathbf{x}_k^{\text{SM}})] + \alpha_k \|\mathbf{g}_k^{\text{SM}}\|^2}{\alpha_k^2 \|\mathbf{g}_k^{\text{SM}}\|^2}. \quad (3.24)$$

Algoritam 3.5.1 je poznat kao SM metod i uveden je u [155].

Algoritam 3.5.1 SM (Stanimirović-Miladinović) metod.

Ulaz: Ciljna funkcija $f(\mathbf{x})$ i odabrana početna tačka $\mathbf{x}_0 \in \text{dom}(f)$.

- 1: Postaviti $k = 0$, izračunati $f(\mathbf{x}_0)$, $\mathbf{g}_0 = \nabla f(\mathbf{x}_0)$ i uzeti $\gamma_0 = 1$.
- 2: Ako su ispunjeni kriterijumi za kraj ciklusa tada prekinuti iteracije; inače preći na sledeći korak.
- 3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3 za $\mathbf{d}_k = -\gamma_k^{-1} \mathbf{g}_k$.
- 4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \gamma_k^{-1} \mathbf{g}_k$, $f(\mathbf{x}_{k+1})$ i $\mathbf{g}_{k+1} = \nabla f(\mathbf{x}_{k+1})$.
- 5: Odrediti γ_{k+1} pomoću (3.24).
- 6: Ako je $\gamma_{k+1} < 0$, tada postaviti $\gamma_{k+1} = 1$.
- 7: Postaviti $k := k + 1$ i preći na **korak 2**.

Izlaz: $\mathbf{x}_{k+1}$ i $f(\mathbf{x}_{k+1})$.

Dvokoračni (*eng. double step-size*) i **dvosmerni** (*eng. double direction*) ubrzani metodi

nastali su u [126, 131]. Iteracije oblika

$$\mathbf{x}_{k+1}^{\text{ADSS}} = \mathbf{x}_k^{\text{ADSS}} - (\alpha_k(\gamma_k^{\text{ADSS}})^{-1} + l_k) \mathbf{g}_k^{\text{ADSS}}, \quad (3.25)$$

u [126] poznate su kao ubrzane dvokoračne (ADSS) iteracije, gde su α_k i l_k dužine koraka određene pomoću dva nezavisna linijska pretraživanja unazad. Dok su iteracije oblika

$$\mathbf{x}_{k+1}^{\text{ADD}} = \mathbf{x}_k^{\text{ADD}} - \alpha_k(\gamma_k^{\text{ADD}})^{-1} \mathbf{g}_k^{\text{ADD}} + \alpha_k^2 \mathbf{d}_k^{\text{ADD}}, \quad (3.26)$$

u [131] poznate kao ubrzane dvosmerne (ADD) iteracije, gde je α_k dužine koraka određene pomoću linijskog pretraživanja unazad, a $(\gamma_k^{\text{ADD}})^{-1} \mathbf{g}_k^{\text{ADD}}$ i $\mathbf{d}_k^{\text{ADD}}$ generišu odgovarajuće pravce pretraživanja.

Transformisani ubrzani dvokoračni (*eng. transformed accelerated double step size (TADSS)*) metod u [156] je predložen korišćenjem pretpostavke $\alpha_k + l_k = 1$ u (3.25), što daje iteracije oblika

$$\mathbf{x}_{k+1}^{\text{TADSS}} = \mathbf{x}_k^{\text{TADSS}} - (\alpha_k((\gamma_k^{\text{TADSS}})^{-1} - 1) + 1) \mathbf{g}_k^{\text{TADSS}}.$$

Kasnije je u [92] istraživana primena TADSS iteracija u vazduhoplovnoj industriji.

Parametar ubrzanja γ_{k+1} u metodama ADD, ADSS i TADSS definisan je sledećim jednakostima:

$$\gamma_{k+1}^{\text{ADD}} = 2 \frac{f(\mathbf{x}_{k+1}^{\text{ADD}}) - f(\mathbf{x}_k^{\text{ADD}}) - \alpha_k(\mathbf{g}_k^{\text{ADD}})^T (\alpha_k \mathbf{d}_k^{\text{ADD}} - (\gamma_k^{\text{ADD}})^{-1} \mathbf{g}_k^{\text{ADD}})}{(\alpha_k \mathbf{d}_k^{\text{ADD}} - (\gamma_k^{\text{ADD}})^{-1} \mathbf{g}_k^{\text{ADD}})^T (\alpha_k \mathbf{d}_k^{\text{ADD}} - (\gamma_k^{\text{ADD}})^{-1} \mathbf{g}_k^{\text{ADD}})}, \quad (\text{ADD metod [131]})$$

$$\gamma_{k+1}^{\text{ADSS}} = 2 \frac{f(\mathbf{x}_{k+1}^{\text{ADSS}}) - f(\mathbf{x}_k^{\text{ADSS}}) + (\alpha_k(\gamma_k^{\text{ADSS}})^{-1} + l_k) \|\mathbf{g}_k^{\text{ADSS}}\|^2}{(\alpha_k(\gamma_k^{\text{ADSS}})^{-1} + l_k)^2 \|\mathbf{g}_k^{\text{ADSS}}\|^2}, \quad (\text{ADSS metod [126]})$$

$$\gamma_{k+1}^{\text{TADSS}} = 2 \frac{f(\mathbf{x}_{k+1}^{\text{TADSS}}) - f(\mathbf{x}_k^{\text{TADSS}}) + (\alpha_k((\gamma_k^{\text{TADSS}})^{-1} - 1) + 1) \|\mathbf{g}_k^{\text{TADSS}}\|^2}{(\alpha_k((\gamma_k^{\text{TADSS}})^{-1} - 1) + 1)^2 \|\mathbf{g}_k^{\text{TADSS}}\|^2}, \quad (\text{TADSS metod [156]}).$$

Efikasnost metoda poboljšanog opadajućeg gradijenta je numerički testirana u [128].

Specifičan izbor $\gamma_k = 1$ transformiše iteracije poboljšanog opadajućeg gradijenta (3.22) u iterativnu šemu opadajućeg gradijenta (2.4). Dalje, iteracije poboljšanog opadajućeg gradijenta (3.22) u slučaju $\alpha_k = 1$ mogu se posmatrati kao iteracije opadajućeg gradijenta

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k^{-1} \mathbf{g}_k, \quad (3.27)$$

gde γ_k postaje primarna dužina koraka koja treba da bude na odgovarajući način definisana.

Andrej (*Andrei*) je u [6, 10] definisao iteracije oblika

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \theta_k \alpha_k \mathbf{g}_k, \quad (3.28)$$

gde je $\theta_k > 0$ parametar koji poboljšava ponašanje algoritma opadajućeg gradijenta. U [10]

je predložio korišćenje slučajnih vrednosti za θ_k ravnomerno raspoređenih između 0 i 1, dok je u [6] predložio Algoritam 3.5.2 za određivanje vrednosti θ_k u (3.28). Na osnovu iterativno pravilo (3.28) i Algoritam 3.5.2 definisan je algoritam **ubranog opadajućeg gradijenta** (*eng. accelerated gradient descent* (AGD)) u [6].

Algoritam 3.5.2 Izračunavanje vrednosti parametra θ_k u (3.28).

Ulaz: Funkcija cilja $f(\mathbf{x})$, gradijent $\mathbf{g}_k$ i prethodna aproksimacija $\mathbf{x}_k$.

1: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.

2: Izračunati $z = \mathbf{x}_k - \alpha_k \mathbf{g}_k$, $g_z = \nabla f(z)$ i $y_k = g_z - \mathbf{g}_k$.

3: Izračunati $a_k = \alpha_k \mathbf{g}_k^T \mathbf{g}_k$ i $b_k = -\alpha_k y_k^T \mathbf{g}_k$.

Izlaz: $\theta_k = \frac{a_k}{b_k}$.

Barzilai (*Barzilai*) i Borvein (*Borwein*) u [22] su pokrenuli dve dobro poznate varijante metode opadajućeg gradijenta, poznate kao BB metode, sa dužinom koraka $\gamma_k^{BB} := \gamma_k^{-1}$ u (3.27). Dužina koraka γ_k^{BB} u prvom slučaju je definisana minimizacijom vektora $\min_{\gamma} \|\mathbf{s}_{k-1} - \gamma \mathbf{y}_{k-1}\|^2$, što daje

$$\gamma_k^{BB} = \frac{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}{\mathbf{y}_{k-1}^T \mathbf{y}_{k-1}}. \quad (3.29)$$

Simetrični slučaj pretpostavlja minimizaciju $\|\gamma \mathbf{s}_{k-1} - \mathbf{y}_{k-1}\|^2$, koji proizvodi

$$\gamma_k^{BB} = \frac{\mathbf{s}_{k-1}^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}. \quad (3.30)$$

Iteracije BB tipa su definisane korišćenjem γ_k^{BB} na sledeći način:

$$\mathbf{x}_{k+1}^{BB} = \mathbf{x}_k^{BB} - \gamma_k^{BB} \mathbf{g}_k^{BB}.$$

Zbog svoje efikasnosti i jednostavnosti BB metod je privukao veliku pažnju brojnih istraživača širom sveta, što je uticalo na pojavu brojnih modifikacija BB metode u brojnim istraživačkim člancima [38, 49, 50, 51, 52, 140, 141, 142, 183].

Još jedan član klase iteracija poboljšanog opadajućeg gradijenta je metod **skalarne korekcije** (*eng. scalar correction* (SC)) [113], definisan iterativnom šemom (3.27) i pravilom

$$\gamma_{k+1}^{SC} = \begin{cases} \frac{\mathbf{s}_k^T \mathbf{r}_k}{\mathbf{y}_k^T \mathbf{r}_k}, & \mathbf{y}_k^T \mathbf{r}_k > 0, \\ \frac{\|\mathbf{s}_k\|}{\|\mathbf{y}_k\|}, & \mathbf{y}_k^T \mathbf{r}_k \leq 0, \end{cases} \quad \mathbf{r}_k = \mathbf{s}_k - \gamma_k \mathbf{y}_k. \quad (3.31)$$

Shodno tome, iteracije skalarne korekcije su definisane relacijom

$$\mathbf{x}_{k+1}^{SC} = \mathbf{x}_k^{SC} - \gamma_k^{SC} \mathbf{g}_k^{SC}.$$

U radu [59] autor predlaže dva iterativna pravila za **relaksirani opadajući gradijent kvazi-Njutnovog** (*eng. relaxed gradient descent quasi Newton* (RGDQN i RGDQN1))

metoda

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \xi_k \alpha_k \gamma_k^{-1} \mathbf{g}_k, \quad (3.32)$$

tako da je ξ_k odgovarajuća realna vrednost. RGDQN iteracije su definisane sa nasumično generisanim $\xi_k \in (0, 1)$, dok RGDQN1 algoritam koristi $\xi_k = \frac{\gamma_k}{\alpha_k \gamma_{k+1}}$.

3.6 Gradijentni metodi ubrzani Picard-Mann hibridnim iterativnim procesom

Primena Pikard-Manovog (*Picard-Mann*) hibridnog iterativnog procesa iz [90] je još jedan način da se ubrzaju iteracije za rešavanje problema nelinearne optimizacije i data je u [129]. Pikard-Manov hibridni iterativni proces je definisan pomoću sledeće tri relacije

$$\begin{cases} \mathbf{x}_1 = \mathbf{x} \in \mathbb{C}, \\ \mathbf{x}_{k+1} = T\mathbf{y}_k, \\ \mathbf{y}_k = (1 - \Omega_k)\mathbf{x}_k + \Omega_k T\mathbf{x}_k, \quad k \in \mathbb{N}, \end{cases} \quad (3.33)$$

gde je $\Omega_k \in (0, 1)$ realan broj koji se naziva **parametar korekcije** (*eng. correction parameter*) u [129]. Umesto (3.33) može se koristiti ekvivalentna iteracija koja nastaje spajanjem druge i treće relacije

$$\mathbf{x}_{k+1} = T[(1 - \Omega_k)\mathbf{x}_k + \Omega_k T\mathbf{x}_k], \quad k \in \mathbb{N}. \quad (3.34)$$

Funkcija $T : \mathbb{C} \rightarrow \mathbb{C}$ u (3.33) definisana je na konveksnom podskupu $\mathbb{C}$ normiranog prostora $\mathbb{E}$, a $\{\mathbf{x}_k\}$ i $\{\mathbf{y}_k\}$ su nizovi određeni iteracijama (3.33).

U [90] je predložen skup konstantnih vrednosti $\Omega = \Omega_k \in (0, 1), \forall k \in \mathbb{N}$ u numeričkim eksperimentima i zaključak je da proces (3.33) konvergira brže od Pikard, Man i Išikava (*Ishikava*) iterativnih procesa iz [82, 110, 134]. Iteracija (3.34) je primenjena u [129] za hibridizaciju SM metode, i tom prilikom je nastao metod poznat kao HSM. Preciznije, preslikavanje T u (3.33) ili (3.34) je definisano SM iterativnom šemom (3.23), tj. $T\mathbf{y}_k = \mathbf{y}_k - \alpha_k \gamma_k^{-1} \mathbf{g}_k$, a zatim je izveden sledeći skup relacija

$$\begin{cases} \mathbf{x}_1 = \mathbf{x} \in \mathbb{R}, \\ \mathbf{x}_{k+1} = T\mathbf{y}_k = \mathbf{y}_k - \alpha_k \gamma_k^{-1} \mathbf{g}_k, \\ \mathbf{y}_k = (1 - \Omega_k)\mathbf{x}_k + \Omega_k T\mathbf{x}_k = (1 - \Omega_k)\mathbf{x}_k + \Omega_k (\mathbf{x}_k - \alpha_k \gamma_k^{-1} \mathbf{g}_k) = \mathbf{x}_k - \Omega_k \alpha_k \gamma_k^{-1} \mathbf{g}_k, \end{cases} \quad (3.35)$$

gde je $k \in \mathbb{N}$. Zamenom treće jednačine iz (3.35) u drugu, dobija se HSM iterativno pravilo

$$\mathbf{x}_{k+1}^{\text{HSM}} = \mathbf{x}_k^{\text{HSM}} - (\Omega_k + 1)\alpha_k (\gamma_k^{\text{HSM}})^{-1} \mathbf{g}_k^{\text{HSM}}, \quad (3.36)$$

gde je parametar ubrzanja $\gamma_k^{\text{HSM}} > 0$ definisan sa

$$\gamma_{k+1}^{\text{HSM}} = 2\gamma_k^{\text{HSM}} \frac{\gamma_k^{\text{HSM}} [f(\mathbf{x}_{k+1}^{\text{HSM}}) - f(\mathbf{x}_k^{\text{HSM}})] + (\Omega_k + 1)\alpha_k \|\mathbf{g}_k^{\text{HSM}}\|^2}{(\Omega_k + 1)^2 \alpha_k^2 \|\mathbf{g}_k^{\text{HSM}}\|^2}.$$

Modifikovani HSM (MHSM) metod je definisan u [124] tako što se predlaže odgovarajuća početna vrednost u Algoritmu 2.2.3 linijskog pretraživanja unazad.

Postoje brojne primene Pikard-Manovog hibridnog iterativnog procesa u rešavanju problema bezuslovne optimizacije, neke od njih su date u radovima [127, 130, 132]. Hibridizacija ADD metode je razmatrana u [132] u obliku

$$\mathbf{x}_{k+1}^{\text{HADD}} = \mathbf{x}_k^{\text{HADD}} - (\Omega_k + 1)\alpha_k(\gamma_k^{\text{HADD}})^{-1}\mathbf{g}_k^{\text{HADD}} + (\Omega_k + 1)\alpha_k^2\mathbf{d}_k,$$

gde je

$$\gamma_{k+1}^{\text{HADD}} = 2 \frac{f(\mathbf{x}_{k+1}^{\text{HADD}}) - f(\mathbf{x}_k^{\text{HADD}}) - (\Omega_k + 1)(\mathbf{g}_k^{\text{HADD}})^T (\alpha_k^2\mathbf{d}_k - \alpha_k(\gamma_k^{\text{HADD}})^{-1}\mathbf{g}_k^{\text{HADD}})}{(\Omega_k + 1)^2\alpha_k^2 (\alpha_k\mathbf{d}_k - (\gamma_k^{\text{HADD}})^{-1}\mathbf{g}_k^{\text{HADD}})^T (\alpha_k\mathbf{d}_k - (\gamma_k^{\text{HADD}})^{-1}\mathbf{g}_k^{\text{HADD}})}.$$

Hibridizacija ADSS metode je razmatrana u [127], gde su definisane iteracije oblika

$$\mathbf{x}_{k+1}^{\text{HADSS}} = \mathbf{x}_k^{\text{HADSS}} - (\Omega_k + 1) (\alpha_k(\gamma_k^{\text{HADSS}})^{-1} + l_k) \mathbf{g}_k^{\text{HADSS}}, \quad (3.37)$$

pri čemu je

$$\gamma_{k+1}^{\text{HADSS}} = 2 \frac{f(\mathbf{x}_{k+1}^{\text{HADSS}}) - f(\mathbf{x}_k^{\text{HADSS}}) + (\Omega_k + 1) (\alpha_k(\gamma_k^{\text{HADSS}})^{-1} + l_k) \|\mathbf{g}_k^{\text{HADSS}}\|^2}{(\Omega_k + 1)^2 (\alpha_k(\gamma_k^{\text{HADSS}})^{-1} + l_k)^2 \|\mathbf{g}_k^{\text{HADSS}}\|^2}.$$

Nedavno, hibridizacija TADSS metode je predložena, ispitana i testirana u [130] i date su iteracije oblika

$$\mathbf{x}_{k+1}^{\text{HTADSS}} = \mathbf{x}_k^{\text{HTADSS}} - (\Omega_k + 1) (\alpha_k ((\gamma_k^{\text{HTADSS}})^{-1} - 1) + 1) \mathbf{g}_k^{\text{HTADSS}}, \quad (3.38)$$

gde je

$$\gamma_{k+1}^{\text{HTADSS}} = 2 \frac{f(\mathbf{x}_{k+1}^{\text{HTADSS}}) - f(\mathbf{x}_k^{\text{HTADSS}}) + (\Omega_k + 1) (\alpha_k ((\gamma_k^{\text{HTADSS}})^{-1} - 1) + 1) \|\mathbf{g}_k^{\text{HTADSS}}\|^2}{(\Omega_k + 1)^2 (\alpha_k ((\gamma_k^{\text{HTADSS}})^{-1} - 1) + 1)^2 \|\mathbf{g}_k^{\text{HTADSS}}\|^2}.$$

Poglavlje 4

Modifikovani gradijentni metodi

Predstojeće poglavlje sadrži deo glavnih publikovanih naučnih rezultata. Ovi rezultati se pre svega odnose na četiri gradijentna algoritama prezentovana u [86] (MAGD, MSM, HMAGD i HMSM) i na dva gradijentna algoritama (TMSM i DMSM) prezentovana u [85] za rešavanje problema bezuslovne optimizacije.

Samo istraživanje u ovom poglavlju možemo da usmerimo u četiri pravca. Prvi pravac istraživanja kreće od ubrzanog opadajućeg gradijenta, gde se kroz uvođenje određenih modifikacija u iteracijama dolazi do modifikovanog ubrzanog opadajućeg gradijentnog metoda. Drugi pravac istraživanja se nadovezuje na prvi pravac istraživanja i preko poboljšanog opadajućeg gradijenta dolazi se do modifikovanog poboljšanog opadajućeg gradijentnog metoda. Treći pravac istraživanja nadovezuje se na prvi i drugi pravac istraživanja i uz pomoć Picard-Mann hibridnog iterativnog procesa dolazi se do hibridnog modifikovanog ubrzanog gradijentnog (HMAGD) metoda i hibridnog modifikovanog poboljšanog gradijentnog (HMSM) metoda. Što se tiče četvrtog pravca istraživanja on se nadovezuje na drugi pravac istraživanja i uz pomoć višestruke upotrebe linijskog pretraživanja unazad radi se na poboljšanju efikasnosti MSM metode.

Na kraju ovog poglavlja ispitivano je ponašanje predloženih algoritama za rešavanje problema bezuslovne optimizacije. Najvažniji pomak u pogledu poboljšanja performansi gradijentnih metoda koji je postignut konstruisanjem modifikovanog poboljšanog opadajućeg gradijentnog metoda ogleda se u višestrukom poboljšanju praćenih indikatora: broj iteracija, broj evaluacija funkcije i CPU vreme.

4.1 Modifikovani ubrzani gradijentni (MAGD) metod

Modifikacija iteracija opadajućeg gradijenta (2.4) je predložena u [86] i označena sa MGD, gde je definisana iterativnim pravilom

$$\mathbf{x}_{k+1}^{\text{MGD}} = \mathbf{x}_k^{\text{MGD}} - (\alpha_k + \alpha_k^2 - \alpha_k^3) \mathbf{g}_k^{\text{MGD}}. \quad (4.1)$$

Glavna ideja korišćena u definisanju iteracija (4.1) je zamena osnovne dužine koraka α_k u metodi opadajućeg gradijenta (2.4) za nešto većom dužinom koraka $\alpha_k^{mod} = \alpha_k + \alpha_k^2 - \alpha_k^3$.

Ideja za definisanje dužine koraka α_k^{mod} proizašla je iz istraživanja datog u [131], gde je iterativna šema definisana relacijom (3.26) i metodom opadajućeg gradijenta. Imajući u vidu da je $\alpha_k^{mod} \geq \alpha_k$, očigledno pokušavamo da koristimo nešto veću dužinu koraka u cilju stvaranja metode koja je bolja od nekih postojećih metoda u ovoj oblasti. Pretpostavlja se da je α_k definisano procedurom za linijsko pretraživanje unazad (Algoritam 2.2.3), što podrazumeva da je $\alpha_k \in (0, 1]$. Kao posledica toga, obrazloženje za ovu modifikaciju leži u nejednakostima

$$\alpha_k \leq \alpha_k + \alpha_k^2 - \alpha_k^3 \leq \alpha_k + \alpha_k^2.$$

Generalno, iterativno pravilo (4.1) se bazira na relativno malom povećanju dužine koraka α_k unutar intervala $[\alpha_k, \alpha_k + \alpha_k^2]$.

Koristeći predhodno datu notaciju algoritam ubrzanog opadajućeg gradijenta sa iteracijama (3.28) se transformiše u **modifikovani AGD (MAGD) metod** sa iterativnom šemom

$$\mathbf{x}_{k+1}^{MAGD} = \mathbf{x}_k^{MAGD} - \theta_k(\alpha_k + \alpha_k^2 - \alpha_k^3)\mathbf{g}_k^{MAGD}. \quad (4.2)$$

Na ovaj način uvodimo iterativni metod za безусловnu optimizaciju, koji se naziva **modifikovani ubrzani opadajući gradijentni metod** (MAGD metod).

Algoritam 4.1.1 Modifikovani ubrzani opadajući gradijentni metod (MAGD metod)

Ulaz: Ciljna funkcija $f(\mathbf{x})$ i početna tačka $\mathbf{x}_0^{MAGD} \in \text{dom}(f)$.

- 1: Postaviti $k = 0$ i izračunati $f(\mathbf{x}_0^{MAGD})$ i $\mathbf{g}_0^{MAGD} = \nabla f(\mathbf{x}_0^{MAGD})$.
- 2: Ako su ispunjeni kriterijumi za kraj ciklusa tada prekinuti iteracije; inače preći na sledeći korak.
- 3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.
- 4: Izračunati $\mathbf{z}^{MAGD} = \mathbf{x}_k^{MAGD} - \alpha_k \mathbf{g}_k^{MAGD}$, $\mathbf{g}_z^{MAGD} = \nabla f(\mathbf{z}^{MAGD})$ i $\mathbf{y}_k^{MAGD} = \mathbf{g}_z^{MAGD} - \mathbf{g}_k^{MAGD}$.
- 5: Izračunati $a_k = \alpha_k(\mathbf{g}_k^{MAGD})^T \mathbf{g}_k^{MAGD}$, $b_k = -\alpha_k(\mathbf{y}_k^{MAGD})^T \mathbf{g}_k^{MAGD}$ i $\theta_k = a_k/b_k$.
- 6: Izračunati $\mathbf{x}_{k+1}^{MAGD} = \mathbf{x}_k^{MAGD} - \theta_k(\alpha_k + \alpha_k^2 - \alpha_k^3)\mathbf{g}_k^{MAGD}$.
- 7: Izračunati $f(\mathbf{x}_{k+1}^{MAGD})$ i $\mathbf{g}_{k+1}^{MAGD} = \nabla f(\mathbf{x}_{k+1}^{MAGD})$.
- 8: Postaviti $k := k + 1$ i preći na korak 2.

Izlaz: $\mathbf{x}_{k+1}^{MAGD}$ i $f(\mathbf{x}_{k+1}^{MAGD})$.

4.2 Modifikovani poboljšani gradijentni (MSM) metod

Modifikovanje iteracija metode poboljšanog opadajućeg gradijenta (3.22), proizvodi klasu iteracija opšteg oblika

$$\mathbf{x}_{k+1}^{MIGD} = \mathbf{x}_k^{MIGD} - \gamma_k^{-1}(\alpha_k + \alpha_k^2 - \alpha_k^3)\mathbf{g}_k^{MIGD}, \quad (4.3)$$

gde je γ_k definisano korišćenjem odgovarajućeg Tejlorovog razvoja. Očigledno, pošto je $\gamma_k > 0$ sledi da važi nejednakost

$$\gamma_k^{-1} \alpha_k \leq \gamma_k^{-1} (\alpha_k + \alpha_k^2 - \alpha_k^3),$$

koja implicira da (4.3) definiše odgovarajuću klasu **modifikovanih poboljšanih opadajućih gradijentnih metoda**, koja je označena kao MIGD klasa.

Moguće je razmotriti modifikacije različitih metoda poboljšanog opadajućeg gradijenta, kao što su SM, ADSS i TADSS. Ove modifikacije će biti definisane redom kao MSM, MADSS i MTADSS metode, a detaljno će biti opisan samo MSM metod. Primena modifikacije na SM metod dovodi do iteracija

$$\mathbf{x}_{k+1}^{MSM} = \mathbf{x}_k^{MSM} - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM}. \quad (4.4)$$

U tom slučaju, iz Tejlorovog razvoja drugog reda, aproksimacija funkcije $f(\mathbf{x}_{k+1}^{MSM})$ može se dati na sledeći način:

$$\begin{aligned} f(\mathbf{x}_{k+1}^{MSM}) \approx & f(\mathbf{x}_k^{MSM}) - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} (\mathbf{g}_k^{MSM})^T \mathbf{g}_k^{MSM} \\ & + \frac{1}{2} (\alpha_k + \alpha_k^2 - \alpha_k^3)^2 ((\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM})^T \nabla^2 f(\xi) (\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM}. \end{aligned} \quad (4.5)$$

Parametar ξ u (4.5) ispunjava uslov $\xi \in [\mathbf{x}_k^{MSM}, \mathbf{x}_{k+1}^{MSM}]$, gde je

$$\xi = \mathbf{x}_k^{MSM} + \delta(\mathbf{x}_{k+1}^{MSM} - \mathbf{x}_k^{MSM}) = \mathbf{x}_k^{MSM} - \delta(\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM}, \quad 0 \leq \delta \leq 1. \quad (4.6)$$

Sledeći [155], matrica $\nabla^2 f(\xi)$ je zamenjena dijagonalnom matricom $\gamma_{k+1}^{MSM} I$. Na osnovu prethodnog, izraz (4.5) postaje

$$\begin{aligned} f(\mathbf{x}_{k+1}^{MSM}) \approx & f(\mathbf{x}_k^{MSM}) - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \|\mathbf{g}_k^{MSM}\|^2 \\ & + \frac{1}{2} (\alpha_k + \alpha_k^2 - \alpha_k^3)^2 \gamma_{k+1}^{MSM} (\gamma_k^{MSM})^{-2} \|\mathbf{g}_k^{MSM}\|^2. \end{aligned} \quad (4.7)$$

Tada se γ_{k+1}^{MSM} može izraziti iz (4.7) na sledeći način:

$$\gamma_{k+1}^{MSM} = 2\gamma_k^{MSM} \frac{\gamma_k^{MSM} [f(\mathbf{x}_{k+1}^{MSM}) - f(\mathbf{x}_k^{MSM})] + (\alpha_k + \alpha_k^2 - \alpha_k^3) \|\mathbf{g}_k^{MSM}\|^2}{(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 \|\mathbf{g}_k^{MSM}\|^2}. \quad (4.8)$$

Slično kao u [155] i ovde, neophodan uslov koji parametar γ_{k+1}^{MSM} definisan sa (4.8) treba ispuniti je $\gamma_{k+1}^{MSM} > 0$, jer u suprotnom potreban i dovoljan uslov optimalnosti drugog reda (*eng. Second-Order Necessary Condition and Second-Order Sufficient Condition*) neće biti ispunjeni.

Ukoliko je tačka $\mathbf{x}_k$ lokalni minimum glatke funkcije, onda gradijent $\mathbf{g}_k$ funkcije teži nuli u tački $\mathbf{x}_k$ i matrica Hesijana je pozitivno semidefinitna (**potreban uslov optimalnosti drugog reda**) [162]. Sa druge strane, ukoliko gradijent $\mathbf{g}_k$ teži nuli u nekoj tački $\mathbf{x}_k$ i

ukoliko je matrica Hesijana pozitivno definitna, onda funkcija dostiže minimum u toj tački (**dovoljan uslov optimalnosti drugog reda**) [162]. Dakle, zadovoljenje potrebnog i dovoljnog uslova optimalnosti zavisi od pozitivnosti parametra γ_{k+1}^{MSM} .

Ukoliko se neželjena situacija $\gamma_{k+1}^{MSM} < 0$ desi u bilo kom iterativnom koraku, onda se problem može rešiti uzimanjem $\gamma_{k+1}^{MSM} = 1$. Obrazloženje za ovakav izbor parametra γ_{k+1}^{MSM} proizilazi iz činjenice da ukoliko se dogodi da matrica Hesijan G_k nije pozitivno definitna onda se izborom $\gamma_{k+1}^{MSM} = 1$ iterativna šema (4.4) svodi na iterativnu šemu modifikovanog opadajućeg gradijentnog metoda (4.1). Na taj način se vrednost funkcije u narednoj iteraciji određuje metodom modifikovanog opadajućeg gradijenta.

Za kraj ovog odeljka, u Algoritmu 4.2.1 prikazujemo modifikovani SM metod (MSM metod):

Algoritam 4.2.1 Modifikovani SM metod (MSM metod)

Ulaz: Ciljna funkcija $f(\mathbf{x})$ i početna tačka $\mathbf{x}_0^{MSM} \in \text{dom}(f)$.

- 1: Postaviti $k = 0$, izračunati $f(\mathbf{x}_0^{MSM})$, $\mathbf{g}_0^{MSM} = \nabla f(\mathbf{x}_0^{MSM})$ i uzeti $\gamma_0^{MSM} = 1$.
- 2: Ako su ispunjeni kriterijumi za kraj ciklusa tada prekinuti iteracije; inače preći na sledeći korak.
- 3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.
- 4: Izračunati $\mathbf{x}_{k+1}^{MSM} = \mathbf{x}_k^{MSM} - (\gamma_k^{MSM})^{-1}(\alpha_k + \alpha_k^2 - \alpha_k^3)\mathbf{g}_k^{MSM}$.
- 5: Izračunati $f(\mathbf{x}_{k+1}^{MSM})$ i $\mathbf{g}_{k+1}^{MSM} = \nabla f(\mathbf{x}_{k+1}^{MSM})$.
- 6: Odrediti skalarnu aproksimaciju γ_{k+1}^{MSM} Hesijana funkcije f u tački $\mathbf{x}_{k+1}^{MSM}$ koristeći (4.8).
- 7: Ako je $\gamma_{k+1}^{MSM} < 0$, tada postaviti $\gamma_{k+1}^{MSM} = 1$.
- 8: Postaviti $k := k + 1$, i preći na korak 2.

Izlaz: $\mathbf{x}_{k+1}^{MSM}$ i $f(\mathbf{x}_{k+1}^{MSM})$.

4.3 Hibridni modifikovani gradijentni metodi

Naredna klasa iteracija je definisana hibridnom korekcijom iteracija opadajućeg gradijenta (2.4). Hibridni oblik metode opadajućeg gradijenta je definisan u prostoru $\mathbb{C} := \mathbb{R}^n$, pod pretpostavkom da je preslikavanje $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ u (3.33) definisano iteracijom opadajućeg gradijenta, tj. $T\mathbf{y}_k = GD(\mathbf{y}_k) = \mathbf{y}_k - \alpha_k \mathbf{g}_k$. Zatim, koristeći (3.33), možemo da izvedemo sledeći skup iteracija:

$$\begin{cases} \mathbf{x}_1 = \mathbf{x} \in \mathbb{R}^n, \\ \mathbf{x}_{k+1} = GD(\mathbf{y}_k) = \mathbf{y}_k - \alpha_k \mathbf{g}_k, \\ \mathbf{y}_k = (1 - \Omega_k)\mathbf{x}_k + \Omega_k GD(\mathbf{x}_k) \\ \quad = (1 - \Omega_k)\mathbf{x}_k + \Omega_k (\mathbf{x}_k - \alpha_k \mathbf{g}_k) \\ \quad = \mathbf{x}_k - \Omega_k \alpha_k \mathbf{g}_k, \quad k \in \mathbb{N}. \end{cases} \quad (4.9)$$

Zamenom $\mathbf{y}_k$ iz treće jednačine u drugoj jednačini za $\mathbf{x}_{k+1}$ u (4.9), dobijamo hibridni metod opadajućeg gradijenta (HGD metod) koja je definisana iteracijom

$$\mathbf{x}_{k+1}^{HGD} = \mathbf{x}_k^{HGD} - (\Omega_k + 1)\alpha_k \mathbf{g}_k^{HGD}. \quad (4.10)$$

Kako je $\alpha_k \in (0, 1]$ i $\Omega_k + 1 \geq 1$, sledi da važi nejednakost

$$\alpha_k \leq (\Omega_k + 1)\alpha_k,$$

iz koje sledi da iteracije hibridnog opadajućeg gradijenta (4.10) definišu još jedno povećanje dužine koraka u klasi metoda opadajućeg gradijenta.

Takođe, istražujemo modifikaciju opadajućeg gradijenta (4.1) i hibridizaciju opadajućeg gradijenta (4.10) spojeni u jedno iterativno pravilo

$$\mathbf{x}_{k+1}^{HMGD} = \mathbf{x}_k^{HMGD} - (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3)\mathbf{g}_k^{HMGD}. \quad (4.11)$$

Klasa iteracija (4.11) se naziva klasom **hibridne modifikacije metode opadajućeg gradijenta** (HMGD klasa). Kako je $\alpha_k \in (0, 1]$ i $\Omega_k + 1 \geq 1$, sledi da važi nejednakost

$$\alpha_k \leq \alpha_k + \alpha_k^2 - \alpha_k^3 \leq (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3),$$

iz koje sledi da iteracije hibridne modifikacije metode opadajućeg gradijenta (4.11) definišu povećanje dužine koraka u MGD klasi, a MGD klasa je bazirana na povećanju dužine koraka u metodi opadajućeg gradijenta.

4.3.1 Hibridni modifikovani ubrzani gradijentni (HMAGD) metod

Da bi definisali hibridni oblik ubrzanog opadajućeg gradijentnog metoda (AGD metod), pretpostavimo da je $\mathbb{C} = \mathbb{R}^n$ i preslikavanje $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ u (3.33) definisano AGD iteracijom (3.28), tj., $T\mathbf{y}_k = AGD(\mathbf{y}_k) = \mathbf{y}_k - \theta_k \alpha_k \mathbf{g}_k$. Zatim, koristeći (3.33), dobijene su sledeće iteracije:

$$\begin{cases} \mathbf{x}_1 = \mathbf{x} \in \mathbb{R}^n, \\ \mathbf{x}_{k+1} = AGD(\mathbf{y}_k) = \mathbf{y}_k - \theta_k \alpha_k \mathbf{g}_k, \\ \mathbf{y}_k = (1 - \Omega_k)\mathbf{x}_k + \Omega_k AGD(\mathbf{x}_k) \\ \quad = (1 - \Omega_k)\mathbf{x}_k + \Omega_k (\mathbf{x}_k - \theta_k \alpha_k \mathbf{g}_k) \\ \quad = \mathbf{x}_k - \Omega_k \theta_k \alpha_k \mathbf{g}_k, \quad k \in \mathbb{N}. \end{cases} \quad (4.12)$$

Zamenom $\mathbf{y}_k$ iz treće jednačine u $\mathbf{x}_{k+1}$ u drugoj jednačini u (4.12), dobija se **hibridni ubrzani gradijentni** (HAGD) metod

$$\mathbf{x}_{k+1}^{HAGD} = \mathbf{x}_k^{HAGD} - (\Omega_k + 1)\theta_k \alpha_k \mathbf{g}_k^{HAGD}. \quad (4.13)$$

Identičnom procedurom dobija se **hibridni modifikovani ubrzani gradijentni metod**

(HMAGD metod)

$$\mathbf{x}_{k+1}^{HMAGD} = \mathbf{x}_k^{HMAGD} - (\Omega_k + 1)\theta_k(\alpha_k + \alpha_k^2 - \alpha_k^3)\mathbf{g}_k^{HMAGD}. \quad (4.14)$$

Parametar dužine koraka α_k se izračunava korišćenjem Algoritma 2.2.3 za netačno linijsko pretraživanje unazad. Dok u svakoj iteraciji za parametar korekcije Ω_k uzimamo konstantnu vrednost iz intervala $(0, 1)$, kao što je predloženo u [90]. U nastavku predstavljamo Algoritam 4.3.1 za HMAGD metod.

Algoritam 4.3.1 Hibridni modifikovani ubrzani gradijentni (HMAGD) metod

Ulaz: Ciljna funkcija $f(\mathbf{x})$, $\Omega_k \in (0, 1)$ i početna tačka $\mathbf{x}_0^{HMAGD} \in \text{dom}(f)$.

- 1: Postaviti $k = 0$ i izračunati $f(\mathbf{x}_0^{HMAGD})$ i $\mathbf{g}_0^{HMAGD} = \nabla f(\mathbf{x}_0^{HMAGD})$.
- 2: Ako su ispunjeni kriterijumi za kraj ciklusa tada prekinuti iteracije; inače preći na sledeći korak.
- 3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritma 2.2.3.
- 4: Izračunati $\mathbf{z}^{HMAGD} = \mathbf{x}_k^{HMAGD} - \alpha_k \mathbf{g}_k^{HMAGD}$, $\mathbf{g}_z^{HMAGD} = \nabla f(\mathbf{z}^{HMAGD})$ i $\mathbf{y}_k^{HAAGD} = \mathbf{g}_z^{HMAGD} - \mathbf{g}_k^{HMAGD}$.
- 5: Izračunati $a_k = \alpha_k(\mathbf{g}_k^{HMAGD})^T \mathbf{g}_k^{HMAGD}$, $b_k = -\alpha_k(\mathbf{y}_k^{HAAGD})^T \mathbf{g}_k^{HMAGD}$ i $\theta_k = a_k / b_k$.
- 6: Izračunati $\mathbf{x}_{k+1}^{HMAGD} = \mathbf{x}_k^{HMAGD} - (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3)\theta_k \mathbf{g}_k^{HMAGD}$.
- 7: Izračunati $f(\mathbf{x}_{k+1}^{HMAGD})$ i $\mathbf{g}_{k+1}^{HMAGD} = \nabla f(\mathbf{x}_{k+1}^{HMAGD})$.
- 8: Postaviti $k := k + 1$ i preći na korak 2.

Izlaz: $\mathbf{x}_{k+1}^{HMAGD}$ i $f(\mathbf{x}_{k+1}^{HMAGD})$.

4.3.2 Hibridni modifikovani poboljšani gradijentni (HMSM) metod

Naredna klasa iteracija je definisana hibridnom korekcijom iteracija poboljšanog opadajućeg gradijenta (3.22) i definisana je iteracijom

$$\mathbf{x}_{k+1}^{HIGD} = \mathbf{x}_k^{HIGD} - (\Omega_k + 1)\gamma_k^{-1}\alpha_k \mathbf{g}_k^{HIGD}, \quad (4.15)$$

gde je $0 < \Omega_k < 1$ odabrani odgovarajući realni parametar definisan pomoću (3.33).

Takođe, razmatramo definisanje iterativnog pravila na osnovu modifikovanog poboljšanog opadajućeg gradijenta (4.3) i hibridnog poboljšanog opadajućeg gradijenta (HIGD) (4.15) dato u sledećem obliku

$$\mathbf{x}_{k+1}^{HMIGD} = \mathbf{x}_k^{HMIGD} - (\Omega_k + 1)\gamma_k^{-1}(\alpha_k + \alpha_k^2 - \alpha_k^3)\mathbf{g}_k^{HMIGD}. \quad (4.16)$$

Budući da je $\alpha_k \in (0, 1]$, $\gamma_k > 0$ i $\Omega_k + 1 \geq 1$, sledi važi nejednakost

$$\alpha_k \gamma_k^{-1} \leq (\alpha_k + \alpha_k^2 - \alpha_k^3) \gamma_k^{-1} \leq (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3) \gamma_k^{-1},$$

iz koje sledi da HIGD definisan u (4.15) definiše još jedno povećanje dužine koraka u IGD klasi, a HMIGD definisan u (4.16) je ubrzanje od HIGD i MIGD, i naravno od IGD. Klasa iteracija (4.15) naziva se klasom **hibridnog poboljšanog opadajućeg gradijenta (HIGD)**. Konačno, klasa iteracija (4.16) naziva se klasom **hibridnog modifikovanog poboljšanog opadajućeg gradijenta (HMIGD)**.

Na osnovu uvedene klasifikacije možemo zaključiti da HSM metod (3.36) pripada HIGD klasi.

Konkretno, hibridizacija MSM iteracija (4.4) biće označena sa HMSM i definisana na sledeći način

$$\mathbf{x}_{k+1}^{HMSM} = \mathbf{x}_k^{HMSM} - (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{HMSM})^{-1} \mathbf{g}_k^{HMSM}. \quad (4.17)$$

Da bismo završili definisanje **metode hibridnog modifikovanog poboljšanog gradijenta (HMSM)**, potrebno je izračunati vrednost parametra ubrzanja γ_k^{HMSM} . Aproksimacija matrice Hesijana ciljne funkcije f data je dijagonalnom matricom

$$\nabla^2 f(\xi) \approx \gamma_{k+1}^{HMSM} I, \quad (4.18)$$

u kojoj je $\gamma_{k+1}^{HMSM} = \gamma(\mathbf{x}_{k+1}^{HMSM}, \mathbf{x}_k^{HMSM})$ odgovarajući realan broj baziran na Tejlorovom razvoju drugog reda ciljne funkcije f u tački $\mathbf{x}_{k+1}^{HMSM}$:

$$\begin{aligned} f(\mathbf{x}_{k+1}^{HMSM}) &\approx f(\mathbf{x}_k^{HMSM}) - (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3)(\mathbf{g}_k^{HMSM})^T (\gamma_k^{HMSM})^{-1} \mathbf{g}_k^{HMSM} \\ &\quad + \frac{1}{2}(\Omega_k + 1)^2(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 ((\gamma_k^{HMSM})^{-1} \mathbf{g}_k^{HMSM})^T \nabla^2 f(\xi) (\gamma_k^{HMSM})^{-1} \mathbf{g}_k^{HMSM}. \end{aligned} \quad (4.19)$$

U prethodnoj jednačini, promenljiva ξ ispunjava uslov $\xi \in [\mathbf{x}_k^{HMSM}, \mathbf{x}_{k+1}^{HMSM}]$. Pošto je tačka $\mathbf{x}_k^{HMSM}$ dovoljno blizu tački $\mathbf{x}_{k+1}^{HMSM}$, razumno je uzeti $\xi = \mathbf{x}_{k+1}^{HMSM}$. Sada, na osnovu jednakosti (4.18), imamo

$$\begin{aligned} f(\mathbf{x}_{k+1}^{HMSM}) &\approx f(\mathbf{x}_k^{HMSM}) - (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{HMSM})^{-1} \|\mathbf{g}_k^{HMSM}\|^2 \\ &\quad + \frac{1}{2}(\Omega_k + 1)^2(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 \gamma_{k+1}^{HMSM} (\gamma_k^{HMSM})^{-2} \|\mathbf{g}_k^{HMSM}\|^2. \end{aligned} \quad (4.20)$$

Iz (4.20) može se izraziti γ_{k+1}^{HMSM} na sledeći način:

$$\gamma_{k+1}^{HMSM} = 2\gamma_k^{HMSM} \frac{\gamma_k^{HMSM} [f(\mathbf{x}_{k+1}^{HMSM}) - f(\mathbf{x}_k^{HMSM})] + (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3) \|\mathbf{g}_k^{HMSM}\|^2}{(\Omega_k + 1)^2(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 \|\mathbf{g}_k^{HMSM}\|^2}. \quad (4.21)$$

Slično kao i za MSM metod i ovde za HMSM metod zadovoljenje neophodnog i dovoljnog uslova optimalnosti zavisi od pozitivnosti parametra γ_{k+1}^{HMSM} . Ukoliko se neželjena situacija $\gamma_{k+1}^{HMSM} < 0$ desi u bilo kom iterativnom koraku, onda se problem rešava uzimanjem vrednosti $\gamma_{k+1}^{HMSM} = 1$.

Sada je sve spremno da predstavimo Algoritam 4.3.2 za HMSM metod.

Algoritam 4.3.2 Hibridni modifikovani poboljšani gradijentni metod (HMSM metod)

Ulaz: Ciljna funkcija $f(\mathbf{x})$, $\Omega_k \in (0, 1)$ i početna tačka $\mathbf{x}_0^{HMSM} \in \text{dom}(f)$.

- 1: Postaviti $k = 0$, izračunati $f(\mathbf{x}_0^{HMSM})$, $\mathbf{g}_0^{HMSM} = \nabla f(\mathbf{x}_0^{HMSM})$ i uzeti $\gamma_0^{HMSM} = 1$.
- 2: Ako su ispunjeni kriterijumi za kraj ciklusa tada prekinuti iteracije; inače preći na sledeći korak.
- 3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.
- 4: Izračunati $\mathbf{x}_{k+1}^{HMSM} = \mathbf{x}_k^{HMSM} - (\Omega_k + 1)(\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{HMSM})^{-1} \mathbf{g}_k^{HMSM}$.
- 5: Izračunati $f(\mathbf{x}_{k+1}^{HMSM})$ i $\mathbf{g}_{k+1}^{HMSM} = \nabla f(\mathbf{x}_{k+1}^{HMSM})$.
- 6: Odrediti γ_{k+1}^{HMSM} pomoću (4.21).
- 7: Ako je $\gamma_{k+1}^{HMSM} < 0$, tada postaviti $\gamma_{k+1}^{HMSM} = 1$.
- 8: Postaviti $k := k + 1$ i preći na korak 2.

Izlaz: $\mathbf{x}_{k+1}^{HMSM}$ i $f(\mathbf{x}_{k+1}^{HMSM})$.

4.4 Višestruka upotreba linijskog pretraživanja unazad u MSM metodi

Istraživanja opisana u prethodnim odeljcima ovog poglavlja stvorila su ideju o proučavanju uticaja višestruke upotrebe linijskog pretraživanja unazad u MSM metodi. Zadatak proučavanja ovakve ideje je stvaranje novih efikasnih algoritama koji će dovesti do poboljšanja karakteristika MSM metode. Istraživanja obavljena u [85] dala su neke odgovore u pogledu definisanja novih algoritama koji će dovesti do poboljšanja karakteristika MSM metode. U ovom i narednim odeljcima dat je pregled i analiza rezulta iz [85] o istraživanju višestruke upotrebe linijskog pretraživanja unazad u MSM metodi koja je definisana iterativnom šemom (4.4), odnosno

$$\mathbf{x}_{k+1}^{MSM} = \mathbf{x}_k^{MSM} - \alpha_k^{MSM} (\gamma_k^{MSM})^{-1} \mathbf{g}_k, \quad (4.22)$$

gde je $\alpha_k^{MSM} = \alpha_k + \alpha_k^2 - \alpha_k^3$.

Istraživanje u [85] je bazirano na poboljšanju ponašanja iteracija (4.22) korišćenjem dodatna dva ili tri odgovarajuće definisana parametra za dužinu koraka. Prateći ovu ideju, metod zasnovan na trostrukoj upotrebi linijskog pretraživanja unazad u MSM metodi se dobija kada se α_k^2 zameni sa l_k^2 , a α_k^3 zameni sa j_k^3 u (4.22), gde su α_k , l_k i j_k definisani nezavisnim procedurama linijskog pretraživanja: prvo linijsko pretraživanje unazad (Algoritam 2.2.3) izračunava α_k , drugo linijsko pretraživanje unazad (Algoritam 4.4.1) izračunava l_k , dok treće linijsko pretraživanje unazad (Algoritam 4.4.2) određuje j_k .

Primenom navedenih izmena u (4.22) dobijamo **trostruku upotrebu linijskog pretraživanja unazad u MSM metodi (TMSM metod)**:

$$\mathbf{x}_{k+1}^{TMSM} = \mathbf{x}_k^{TMSM} - \alpha_k^{TMSM} (\gamma_k^{TMSM})^{-1} \mathbf{g}_k, \quad (4.23)$$

gde je

$$\alpha_k^{TMSM} = \begin{cases} \alpha_k + l_k^2 - j_k^3, & \alpha_k + l_k^2 - j_k^3 > \alpha_k \\ \alpha_k, & \alpha_k + l_k^2 - j_k^3 \leq \alpha_k. \end{cases} \quad (4.24)$$

Parametar ubrzanja γ_k^{TMSM} u (4.23) određen je primenom Tejlorov razvoj drugog reda na funkciju $f(\mathbf{x}_{k+1}^{TMSM})$ u tački $\mathbf{x}_{k+1}^{TMSM}$:

$$\begin{aligned} f(\mathbf{x}_{k+1}^{TMSM}) &\approx f(\mathbf{x}_k^{TMSM}) - \alpha_k^{TMSM} (\gamma_k^{TMSM})^{-1} \mathbf{g}_k^T \mathbf{g}_k \\ &\quad + \frac{1}{2} (\alpha_k^{TMSM})^2 ((\gamma_k^{TMSM})^{-1} \mathbf{g}_k)^T \nabla^2 f(\xi) (\gamma_k^{TMSM})^{-1} \mathbf{g}_k. \end{aligned} \quad (4.25)$$

Parametar ξ u (4.25) ispunjava sledeći uslov $\xi \in [\mathbf{x}_k^{TMSM}, \mathbf{x}_{k+1}^{TMSM}]$ i

$$\begin{aligned} \xi &= \mathbf{x}_k^{TMSM} + \varphi (\mathbf{x}_{k+1}^{TMSM} - \mathbf{x}_k^{TMSM}) \\ &= \mathbf{x}_k^{TMSM} - \varphi \alpha_k^{TMSM} (\gamma_k^{TMSM})^{-1} \mathbf{g}_k, \quad 0 \leq \varphi \leq 1. \end{aligned} \quad (4.26)$$

Na osnovu [155], Hesijan $\nabla^2 f(\xi)$ je aproksimiran sa $\gamma_{k+1}^{TMSM} I$. Prem tome, (4.25) se svodi na

$$\begin{aligned} f(\mathbf{x}_{k+1}^{TMSM}) &\approx f(\mathbf{x}_k^{TMSM}) - \alpha_k^{TMSM} (\gamma_k^{TMSM})^{-1} \|\mathbf{g}_k\|^2 \\ &\quad + \frac{1}{2} (\alpha_k^{TMSM})^2 \gamma_{k+1}^{TMSM} (\gamma_k^{TMSM})^{-2} \|\mathbf{g}_k\|^2. \end{aligned} \quad (4.27)$$

Tada γ_{k+1}^{TMSM} može se izraziti iz (4.27) na sledeći način

$$\gamma_{k+1}^{TMSM} = 2\gamma_k^{TMSM} \frac{\gamma_k^{TMSM} [f(\mathbf{x}_{k+1}^{TMSM}) - f(\mathbf{x}_k^{TMSM})] + \alpha_k^{TMSM} \|\mathbf{g}_k\|^2}{(\alpha_k^{TMSM})^2 \|\mathbf{g}_k\|^2}. \quad (4.28)$$

Slično kao za MSM i HMSM metode i ovde za TMSM metod zadovoljenje potrebnog i dovoljnog uslova optimalnosti zavisi od pozitivnosti parametra γ_{k+1}^{TMSM} . U slučaju da imamo situaciju $\gamma_{k+1}^{TMSM} < 0$ u nekom iterativnom koraku, onda se problem rešava uzimanjem vrednosti $\gamma_{k+1}^{TMSM} = 1$.

Metod linijskog pretraživanja unazad je dat pomoću Algoritma 2.2.3 u kome se određuje α_k počevši od $\alpha_k = 1$ i nakon toga smanjuje vrednost promenljive α_k tako da dovoljno smanji vrednost ciljne funkcije f . Po istom principu funkcionišu i Algoritmi 4.4.1 i 4.4.2.

Algoritam 4.4.1 Drugo linijsko pretraživanje unazad za izračunavanje l_k .

Ulaz: Funkcija cilja $f(\mathbf{x})$, pravac pretraživanja $\mathbf{d}_k$ u tački $\mathbf{x}_k$ i pozitivni realni brojevi $0 < \sigma_l < 0.5$ i $\beta_l \in (0, 1)$.

1: $l = 1$.

2: Dok važi: $f(\mathbf{x}_k + l\mathbf{d}_k) > f(\mathbf{x}_k) + \sigma_l \mathbf{g}_k^T \mathbf{d}_k$, postavi $l := l\beta_l$.

Izlaz: $l_k = l$.

Konačno, trostruka upotreba linijskog pretraživanja unazad u MSM metodi, odnosno skraćeno TMSM metod je opisan pomoću Algoritma 4.4.3.

Algoritam 4.4.2 Treće linijsko pretraživanje unazad za izračunavanje j_k .

Ulaz: Funkcija cilja $f(\mathbf{x})$, pravac pretraživanja $\mathbf{d}_k$ u tački $\mathbf{x}_k$ i pozitivni realni brojevi $0 < \sigma_j < 0.5$ i $\beta_j \in (0, 1)$.

1: $j = 1$.

2: Dok važi: $f(\mathbf{x}_k + j\mathbf{d}_k) > f(\mathbf{x}_k) + \sigma_j j \mathbf{g}_k^T \mathbf{d}_k$, postaviti $j := j\beta_j$.

Izlaz: $j_k = j$.

Algoritam 4.4.3 Trostruka upotreba linijskog pretraživanja unazad u MSM metodi (TMSM metod)

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0^{TMSM} \in \text{dom}(f)$ i parametri $0 < \lambda < 1$, $0 < \nu < 1$.

1: Postaviti $k = 0$, izračunati $f(\mathbf{x}_0^{TMSM})$, $\mathbf{g}_0 = \nabla f(\mathbf{x}_0^{TMSM})$ i postaviti $\gamma_0^{TMSM} = 1$.

2: Ako je

$$\|\mathbf{g}_k\| \leq \lambda \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}^{TMSM}) - f(\mathbf{x}_k^{TMSM})|}{1 + |f(\mathbf{x}_k^{TMSM})|} \leq \nu,$$

STOP; inače preći na korak 3.

3: Izračunati $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.

4: Izračunati $l_k \in (0, 1]$ koristeći Algoritam 4.4.1.

5: Izračunati $j_k \in (0, 1]$ koristeći Algoritam 4.4.2.

6: Odrediti α_k^{TMSM} pomoću (4.24).

7: Izračunati $\mathbf{x}_{k+1}^{TMSM} = \mathbf{x}_k^{TMSM} - (\gamma_k^{TMSM})^{-1} \alpha_k^{TMSM} \mathbf{g}_k$.

8: Izračunati $f(\mathbf{x}_{k+1}^{TMSM})$ i $\mathbf{g}_{k+1} = \nabla f(\mathbf{x}_{k+1}^{TMSM})$.

9: Odrediti γ_{k+1}^{TMSM} pomoću (4.28).

10: Ako je $\gamma_{k+1}^{TMSM} < 0$, tada postaviti $\gamma_{k+1}^{TMSM} = 1$.

11: Postaviti $k := k + 1$, i preći na korak 2.

Izlaz: $\mathbf{x}_{k+1}^{TMSM}$ i $f(\mathbf{x}_{k+1}^{TMSM})$.

Očekivano je da će ukupan broj iteracija koje zahteva TMSM metod biti manji od broja iteracija MSM metode, ali povećanje broja evaluacija funkcije cilja i CPU vremena je takođe očekivano u TMSM metodi u odnosu na MSM metod. Ova konstatacija direktno proizilazi iz **4.** i **5. koraka** u Algoritmu 4.4.3 za TMSM metod. Dodatno korišćenje Algoritama 4.4.1 i 4.4.2 direktno povećava broj evaluacija funkcije cilja i CPU vreme.

Na osnovu ovih pokazatelja došli smo na ideju da se u TMSM metodi izostavi jedno linijsko pretraživanje. Time bismo drastično smanjili broj evaluacija funkcije cilja i CPU vreme. Prateći ovu ideju, dobijen je metod **dvostruke upotrebe linijskog pretraživanja unazad u MSM metodi (DMSM metod)**, čije iteracije su oblika:

$$\mathbf{x}_{k+1}^{DMSM} = \mathbf{x}_k^{DMSM} - \alpha_k^{DMSM} (\gamma_k^{DMSM})^{-1} \mathbf{g}_k, \quad (4.29)$$

gde je

$$\alpha_k^{DMSM} = \begin{cases} \alpha_k + \alpha_k^2 - j_k^3, & \alpha_k + \alpha_k^2 - j_k^3 > \alpha_k \\ \alpha_k, & \alpha_k + \alpha_k^2 - j_k^3 \leq \alpha_k. \end{cases} \quad (4.30)$$

Na potpuno identičan način kao i za TMSM metod dolazimo do obrazca za određivanje vrednosti parametra ubrzanja

$$\gamma_{k+1}^{DMSM} = 2\gamma_k^{DMSM} \frac{\gamma_k^{DMSM} [f(\mathbf{x}_{k+1}^{DMSM}) - f(\mathbf{x}_k^{DMSM})] + \alpha_k^{DMSM} \|\mathbf{g}_k\|^2}{(\alpha_k^{DMSM})^2 \|\mathbf{g}_k\|^2}. \quad (4.31)$$

Kao u prethodnim slučajevima za MSM, HMSM i TMSM metode tako i ovde za DMSM metod zadovoljenje potrebnog i dovoljnog uslova optimalnosti zavisi od pozitivnosti parametra γ_{k+1}^{DMSM} . U slučaju da imamo situaciju $\gamma_{k+1}^{DMSM} < 0$ u nekom iterativnom koraku, onda se problem rešava uzimanjem vrednost $\gamma_{k+1}^{DMSM} = 1$.

Metod dvostruke upotrebe linijskog pretraživanja unazad u MSM metodi (DMSM metod) je predstavljen Algoritmom 4.4.4.

Algoritam 4.4.4 Dvostruka upotreba linijskog pretraživanja unazad u MSM metodi (DMSM metod)

Ulaz: Ciljna funkcija $f(\mathbf{x})$, odabrana početna tačka $\mathbf{x}_0^{DMSM} \in \text{dom}(f)$ i parametri $0 < \lambda < 1$, $0 < \nu < 1$.

- 1: Postaviti $k = 0$, izračunati $f(\mathbf{x}_0^{DMSM})$, $\mathbf{g}_0 = \nabla f(\mathbf{x}_0^{DMSM})$ i postaviti $\gamma_0^{DMSM} = 1$.
- 2: Ako je

$$\|\mathbf{g}_k\| \leq \lambda \quad \text{ i } \quad \frac{|f(\mathbf{x}_{k+1}^{DMSM}) - f(\mathbf{x}_k^{DMSM})|}{1 + |f(\mathbf{x}_k^{DMSM})|} \leq \nu,$$

STOP; inače preći na korak 3.

- 3: Izračunati $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.
 - 4: Izračunati $j_k \in (0, 1]$ koristeći Algoritam 4.4.2.
 - 5: Odrediti α_k^{DMSM} pomoću (4.30).
 - 6: Izračunati $\mathbf{x}_{k+1}^{DMSM} = \mathbf{x}_k^{DMSM} - (\gamma_k^{DMSM})^{-1} \alpha_k^{DMSM} \mathbf{g}_k$.
 - 7: Izračunati $f(\mathbf{x}_{k+1}^{DMSM})$ i $\mathbf{g}_{k+1} = \nabla f(\mathbf{x}_{k+1}^{DMSM})$.
 - 8: Odrediti skalarnu aproksimaciju $\gamma_{k+1}^{DMSM} I$ Hesijana funkcije f u tački $\mathbf{x}_{k+1}^{DMSM}$ koristeći (4.31).
 - 9: Ako je $\gamma_{k+1}^{DMSM} < 0$, tada postaviti $\gamma_{k+1}^{DMSM} = 1$.
 - 10: Postaviti $k := k + 1$, i preći na korak 2.
- Izlaz:** $\mathbf{x}_{k+1}^{DMSM}$ i $f(\mathbf{x}_{k+1}^{DMSM})$.
-

4.5 Analiza konvergencije

U okviru ovog odeljka, pod standardnim prepostavkama izvršićemo anazlizu konvergencije za MAGD (Algoritam 4.1.1), MSM (Algoritam 4.2.1), HMAGD (Algoritam 4.3.1), HMSM (Algoritam 4.3.2), TMSM (Algoritam 4.4.3) i DMSM (Algoritam 4.4.4) metodu. Osobine konvergencije za navedene metode najpre ispitujemo na skupu uniformno konveksnih funkcija, a zatim uz određena ograničenja, i na podskupu strogo konveksnih kvadratnih funkcija.

U narednom Stavu i Lemi navodimo neke osnovne iskaze potrebne za analizu i dokaz

konvergenције metoda opisanih pomoću Algoritama 4.1.1, 4.2.1, 4.3.1, 4.3.2, 4.4.3 i 4.4.4 za uniformno konveksne funkcije. Dokazi za Stav 4.5.1 i Lemu 4.5.1 mogu se pronaći u [122, 143].

Stav 4.5.1. *Neka je funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i uniformno konveksna na $\mathbb{R}^n$. Tada važi:*

(H_1) *funkcija f ima donju granicu na skupu $\mathcal{B}_0 = \{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$, gde je $\mathbf{x}_0 \in \mathbb{R}^n$ dopustiva tačka;*

(H_2) *gradient $\mathbf{g}$ funkcije f je Lipšić neprekidan na otvorenom konveksnom skupu $\mathcal{B} \supseteq \mathcal{B}_0$, odnosno postoji konstanta $L > 0$ takva da je*

$$\|\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{B}. \quad (4.32)$$

Lema 4.5.1. *Ukoliko važe pretpostavke iz Stava 4.5.1, onda postoje realni brojevi m i M za koje je ispunjeno*

$$0 < m \leq 1 \leq M, \quad (4.33)$$

tako da funkcija $f(\mathbf{x})$ ima jedinstvenu tačku minimuma $\mathbf{x}^$ i pri tom važi:*

$$m\|\mathbf{y}\|^2 \leq \mathbf{y}^T \nabla^2 f(\mathbf{x}) \mathbf{y} \leq M\|\mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n; \quad (4.34)$$

$$\frac{1}{2}m\|\mathbf{x} - \mathbf{x}^*\|^2 \leq f(\mathbf{x}) - f(\mathbf{x}^*) \leq \frac{1}{2}M\|\mathbf{x} - \mathbf{x}^*\|^2, \quad \forall \mathbf{x} \in \mathbb{R}^n; \quad (4.35)$$

$$m\|\mathbf{x} - \mathbf{y}\|^2 \leq (\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{y}))^T (\mathbf{x} - \mathbf{y}) \leq M\|\mathbf{x} - \mathbf{y}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n. \quad (4.36)$$

Sledeći rezultat, preuzet iz [6], biće korišćeni za dalju analizu konvergenције.

Stav 4.5.2. *Neka je $\mathbf{d}_k$ vektor pretraživanja koji ima opadajući pravac i neka gradijent $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$ zadovoljava Lipšićov uslov (4.32). Ako je korišćeno linijsko pretraživanje unazad iz Algoritma 2.2.3, tada važi*

$$\alpha_k \geq \min \left\{ 1, -\frac{\beta(1-\sigma)}{L} \frac{\mathbf{g}_k^T \mathbf{d}_k}{\|\mathbf{d}_k\|^2} \right\}. \quad (4.37)$$

Dokaz. Sledi direktno na osnovu rezultata iz [6]. $\square$

Lema 4.5.2 je predložena u [155] za iterativnu šemu poboljšanog opadajućeg gradijenta. Procenjuje iterativno smanjenje ciljne funkcije prilikom svake iteracije kada se primenjuje metod poboljšanog opadajućeg gradijenta.

Lema 4.5.2. *Za dva puta neprekidno diferencijabilnu i uniformno konveksnu funkciju $f : \mathbb{R}^n \mapsto \mathbb{R}$, i za niz $\{\mathbf{x}_k\}$ generisan algoritmom poboljšanog opadajućeg gradijenta (3.22) važi sledeća nejednakost*

$$f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) \geq \mu \|\mathbf{g}_k\|^2, \quad (4.38)$$

pri čemu je

$$\mu = \min \left\{ \frac{\sigma}{M}, \frac{\sigma(1-\sigma)}{L}\beta \right\}. \quad (4.39)$$

Dokaz. Sledi direktno na osnovu rezultata iz [155]. $\square$

U nastavku se pretpostavlja da je $\mathbf{d}_k$ pravac pretraživanja, a takođe važna činjenica je da skalarna aproksimacija Hesijana omogućava da se izostavi pretpostavka da je funkcija f dva puta neprekidno diferencijabilna. Prema tome, umesto (4.34) i (4.33) koji pretpostavljaju da je funkcija f dva puta neprekidno diferencijabilna i uniformno konveksna, korišćićemo sledeći jednostavniji zahtev za γ_k :

$$m \leq \gamma_k \leq M, \quad 0 < m \leq 1 \leq M, \quad m, M \in \mathbb{R}. \quad (4.40)$$

U slučaju $\gamma_k < 0$ koristiti $\gamma_k = 1$, dok ćemo u slučaju $\gamma_k > M$ koristiti $\gamma_k = M$.

Lema 4.5.3 istražuje konvergenciju modifikovanog poboljšanog opadajućeg gradijentnog metoda za uniformno konveksne funkcije pod pretpostavkama (H_1) i (H_2) .

Lema 4.5.3. *Neka su zadovoljene pretpostavke (H_1) i (H_2) , neka važi (4.40) i neka je $f : \mathbb{R}^n \mapsto \mathbb{R}$ uniformno konveksna funkcija. Tada niz $\{\mathbf{x}_k\}$ generisan pomoću iteracija (4.3) zadovoljava nejednakost oblika (4.38)–(4.39).*

Dokaz. Klasa iteracija modifikovanog poboljšanog opadajućeg gradijentnog metoda je oblika $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$, gde je $\mathbf{d}_k = -\gamma_k^{-1}(1 + \alpha_k - \alpha_k^2)\mathbf{g}_k$. Na osnovu izlaznog uslova u Algoritmu 2.2.3 linijskog pretraživanja unazad, važi sledeća nejednakost

$$f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) \geq -\sigma\alpha_k \mathbf{g}_k^T \mathbf{d}_k. \quad \forall k \in \mathbb{N}. \quad (4.41)$$

Za slučaj $\alpha_k < 1$, uzimajući u obzir da je $\mathbf{d}_k = -\gamma_k^{-1}(1 + \alpha_k - \alpha_k^2)\mathbf{g}_k$ i zamenom u nejednakosti (4.41) može se izvesti sledeća nejednakost

$$\begin{aligned} f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) &\geq -\sigma\alpha_k \mathbf{g}_k^T \mathbf{d}_k = -\sigma\alpha_k \mathbf{g}_k^T (-\gamma_k^{-1}(1 + \alpha_k - \alpha_k^2)\mathbf{g}_k) \\ &= \sigma\alpha_k \frac{(1 + \alpha_k - \alpha_k^2)\mathbf{g}_k^T \mathbf{g}_k}{\gamma_k}. \end{aligned}$$

Sada, iz (4.37) imamo da je

$$\alpha_k \geq \frac{(1-\sigma)\beta}{L} \cdot \frac{\gamma_k}{1 + \alpha_k - \alpha_k^2},$$

zamenom u prethodnoj nejednačini dobija se

$$\begin{aligned} f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) &\geq \sigma \frac{(1-\sigma)\beta\gamma_k}{L(1 + \alpha_k - \alpha_k^2)} \cdot \frac{(1 + \alpha_k - \alpha_k^2)\mathbf{g}_k^T \mathbf{g}_k}{\gamma_k} \\ &\geq \sigma \frac{(1-\sigma)\beta}{L} \|\mathbf{g}_k\|^2. \end{aligned}$$

U skladu sa (4.40), sledeća nejednakost važi za slučaj $\alpha_k = 1$:

$$\begin{aligned} f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) &\geq -\sigma \mathbf{g}_k^T \mathbf{d}_k = -\sigma \mathbf{g}_k^T (-\gamma_k^{-1}(1 + \alpha_k - \alpha_k^2) \mathbf{g}_k) \\ &= \frac{\sigma}{\gamma_k} \|\mathbf{g}_k\|^2 \\ &\geq \frac{\sigma}{M} \|\mathbf{g}_k\|^2. \end{aligned}$$

Konačno, iz prethodne dve nejednačine izvedene za oba moguća slučaja, $\alpha_k < 1$ i $\alpha_k = 1$, dobija se (4.39) čime je dokaz završen. $\square$

Napomena 4.5.1. (a) Rezultat u Lemi 4.5.3 može se direktno primeniti na MSM metod. (b) Na sličan način Lema 4.5.3 može se dokazati za MAGD metod. Da biste dokazali ovu tvrdnju potrebno je izvršiti zamenu γ_k^{-1} sa θ_k . (c) Lema 4.5.3 može se dokazati na sličan način i za DMSM metod. Da biste dokazali ovu tvrdnju, potrebno je za $\mathbf{d}_k$ koristiti sledeći oblik $\mathbf{d}_k = -\frac{\alpha_k^{DMSM}}{\alpha_k} (\gamma_k^{DMSM})^{-1} \mathbf{g}_k$. Dokaz Leme 4.5.3 za DMSM metod može se pronaći u [85, Theorem 3.1]. (d) Dokaz Leme 4.5.3 za TMSM metod sličan je dokazu datom u [85, Theorem 3.1]. Da biste dokazali ovu tvrdnju, potrebno je za $\mathbf{d}_k$ koristiti sledeći oblik $\mathbf{d}_k = -\frac{\alpha_k^{TMSM}}{\alpha_k} (\gamma_k^{TMSM})^{-1} \mathbf{g}_k$.

Lema 4.5.4 istražuje granice iterativnog opadanja funkcije cilja kada se primenjuje na klasu hibridnih modifikovanih poboljšanih opadajućih gradijentnih metoda.

Lema 4.5.4. Neka su zadovoljene pretpostavke (H_1) i (H_2) , neka veži (4.40) i neka je $f : \mathbb{R}^n \mapsto \mathbb{R}$ uniformno konveksna funkcija. Tada nejednakost oblika (4.38) važi za niz $\{\mathbf{x}_k\}$ generisan iterativnom šemom (4.16), gde je

$$\mu = \min \left\{ \frac{\sigma}{M} (\Omega_k + 1), \frac{\sigma(1 - \sigma)}{L} \beta \right\}. \quad (4.42)$$

Dokaz. Analiziramo sledeća dva slučaja koja se odnose na vrednost dužine iterativnog koraka: $\alpha_k < 1$ i $\alpha_k = 1$. Na osnovu izlaznog uslova u Algoritmu 2.2.3 linijskog pretraživanja unazad, važi nejednakost (4.41).

U slučaju $\alpha_k < 1$, uzimajući u obzir (4.37), dobijamo

$$\alpha_k \geq -\frac{\beta(1 - \sigma)}{L} \frac{\mathbf{g}_k^T \mathbf{d}_k}{\|\mathbf{d}_k\|^2}.$$

Iz (4.16) imamo da je $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$, gde je

$$\mathbf{d}_k = -(\Omega_k + 1) \gamma_k^{-1} (1 + \alpha_k - \alpha_k^2) \mathbf{g}_k, \quad (4.43)$$

na osnovu čega sledi

$$\alpha_k \geq \frac{\beta(1 - \sigma) \gamma_k}{L(\Omega_k + 1)(1 + \alpha_k - \alpha_k^2)}. \quad (4.44)$$

Primenom (4.44) i (4.43) u (4.41) dobijaju se sledeće nejednakosti:

$$\begin{aligned} f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) &\geq -\sigma \alpha_k \mathbf{g}_k^T (-(\Omega_k + 1) \gamma_k^{-1} (1 + \alpha_k - \alpha_k^2) \mathbf{g}_k) \\ &\geq \frac{\sigma(1-\sigma)\beta\gamma_k}{L(\Omega_k + 1)(1 + \alpha_k - \alpha_k^2)} \cdot \frac{(\Omega_k + 1)(1 + \alpha_k - \alpha_k^2) \mathbf{g}_k^T \mathbf{g}_k}{\gamma_k} \\ &\geq \frac{\sigma(1-\sigma)\beta}{L} \|\mathbf{g}_k\|^2. \end{aligned}$$

S druge strane, u skladu sa (4.40), važi sledeća nejednakost za $\alpha_k = 1$:

$$f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) \geq -\sigma \mathbf{g}_k^T \mathbf{d}_k \geq \frac{\sigma}{M} (\Omega_k + 1) \|\mathbf{g}_k\|^2.$$

Konačno, iz prethodne dve nejednačine izvedene za $\alpha_k < 1$ i $\alpha_k = 1$, imamo

$$f(\mathbf{x}_k) - f(\mathbf{x}_{k+1}) \geq \min \left\{ \frac{\sigma}{M} (\Omega_k + 1), \frac{\sigma(1-\sigma)}{L} \beta \right\} \|\mathbf{g}_k\|^2,$$

čime je dokaz završen. $\square$

Napomena 4.5.2. (a) Rezultat u Lemi 4.5.4 može se direktno primeniti na HMSM metod. (b) Na sličan način Lema 4.5.4 može se dokazati za HMAGD metod. Da biste dokazali ovu tvrdnju potrebno je izvršiti zamenu γ_k^{-1} sa θ_k .

Napomena 4.5.3. Nakon upoređivanja nejednačina (4.39) i (4.42), može se zaključiti da HMIGD iteracije opravdavaju veće smanjenje $f(\mathbf{x}_{k+1})$ u odnosu na $f(\mathbf{x}_k)$ u odnosu na MIGD i IGD pravila u slučaju $\alpha_k = 1$, zbog $\frac{\sigma}{M} (\Omega_k + 1) > \frac{\sigma}{M}$.

U Teoremama 4.5.1, 4.5.2 i 4.5.3 dokazuje se linearna konvergencija modifikovanih MSM i MAGD metoda, hibridnih HMSM i HMAGD metoda, kao i DMSM i TMSM metoda koje koriste višestruku upotrebu linijskog pretraživanja unazada za uniformno konveksne funkcije.

Teorema 4.5.1. Neka su zadovoljene pretpostavke (H_1) i (H_2) u vezi sa (4.40) i $f : \mathbb{R}^n \mapsto \mathbb{R}$ je uniformno konveksna funkcija.

(a) Ako je niz $\{\mathbf{x}_k^{MSM}\}$ generisan Algoritmom 4.2.1, tada

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{MSM}\| = 0 \tag{4.45}$$

i niz $\{\mathbf{x}_k^{MSM}\}$ konvergira ka $\mathbf{x}^*$ najmanje linearno.

(b) Ako je niz $\{\mathbf{x}_k^{MAGD}\}$ generisan Algoritmom 4.1.1, tada

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{MAGD}\| = 0 \tag{4.46}$$

i niz $\{\mathbf{x}_k^{MAGD}\}$ konvergira ka $\mathbf{x}^*$ najmanje linearno.

Dokaz. Dokaz je sličan dokazu u [155, Teorema 4.1] i biće izostavljen. $\square$

Teorema 4.5.2. *Neka su zadovoljene pretpostavke (H_1) i (H_2) u vezi sa (4.40) i $f : \mathbb{R}^n \mapsto \mathbb{R}$ je uniformno konveksna funkcija.*

(a) *Ako je niz $\{\mathbf{x}_k^{HMSM}\}$ generisan Algoritmom 4.3.2, tada*

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{HMSM}\| = 0 \quad (4.47)$$

i niz $\{\mathbf{x}_k^{HMSM}\}$ konvergira ka $\mathbf{x}^$ najmanje linearno.*

(b) *Ako je niz $\{\mathbf{x}_k^{HMSMAGD}\}$ generisan Algoritmom 4.3.1, tada*

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{HMSMAGD}\| = 0 \quad (4.48)$$

i niz $\{\mathbf{x}_k^{HMSMAGD}\}$ konvergira ka $\mathbf{x}^$ najmanje linearno.*

Dokaz. **(a)** Kako je funkcija cilja f odozdo ograničena i kako je niz $\{f(\mathbf{x}_k^{HMSM})\}$ opadajući imamo da važi

$$\lim_{k \rightarrow \infty} (f(\mathbf{x}_k^{HMSM}) - f(\mathbf{x}_{k+1}^{HMSM})) = 0,$$

gde zajedno sa (4.38) i (4.42) implicira

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{HMSM}\| = 0.$$

Ako je u (4.36) uvedena smena $\mathbf{y} = \mathbf{x}^*$ i primenjena teorema o srednjoj vrednosti kao i Cauchy–Schwartzova nejednakost, imamo da je

$$m\|\mathbf{x}_k^{HMSM} - \mathbf{x}^*\| \leq \|\mathbf{g}(\mathbf{x}_k^{HMSM})\| \leq M\|\mathbf{x}_k^{HMSM} - \mathbf{x}^*\|, \quad \forall \mathbf{x}_k^{HMSM} \in \mathbb{R}^n.$$

Iz ove nejednakosti i (4.35) dobijamo sledeću nejednakost

$$\mu\|\mathbf{g}(\mathbf{x}_k^{HMSM})\|^2 \geq \mu m^2\|\mathbf{x}_k^{HMSM} - \mathbf{x}^*\|^2 \geq 2\mu \frac{m^2}{M}(f(\mathbf{x}_k^{HMSM}) - f(\mathbf{x}^*)).$$

Odavde sledi da $\lim_{k \rightarrow \infty} \|\mathbf{x}_k^{HMSM} - \mathbf{x}^*\| = 0$, odnosno da niz $\{\mathbf{x}_k^{HMSM}\}$ konvergira ka $\mathbf{x}^*$.

Potrebno je dokazati još da je konvergencija linearna. Dokaz linearne konvergenije sledi iz pretpostavke da je $M \geq \sqrt{2}$ i dokaza da važi

$$\rho = \sqrt{2\mu \frac{m^2}{M}} < 1,$$

gde je μ dato pomoću (4.42), a m i M su brojevi definisani sa (4.33) u Lemi 4.5.1. Iz (4.42) mogu se uočiti dva slučaja za μ .

Ako je $\mu = \frac{\sigma}{M}(\Omega_k + 1)$ i kako je $\Omega_k \in (0, 1)$ imamo da važi

$$\rho^2 = 2\mu \frac{m^2}{M} = 2\frac{\sigma}{M}(\Omega_k + 1) \frac{m^2}{M} = 2\sigma(\Omega_k + 1) \frac{m^2}{M^2} < 2 \cdot \sigma \cdot 2 \frac{m^2}{M^2} < 2 \cdot 0.5 \cdot 2 \frac{1^2}{\sqrt{2}^2} = 1.$$

U drugom slučaju, ako je $\mu = \frac{\sigma(1-\sigma)}{L}\beta$ imamo da važi

$$\rho^2 = 2\mu \frac{m^2}{M} = 2\frac{\sigma(1-\sigma)}{L}\beta \frac{m^2}{M} < \frac{m^2}{LM} < 1,$$

jer na osnovu (4.36) važi da je $m \leq L$.

Na kraju, uzimajući rezultat iz [146, Teorema 4.1] postizemo konačnu procenu

$$\|\mathbf{x}_k^{HMSM} - \mathbf{x}^*\| \leq \sqrt{\frac{2(f(\mathbf{x}_0^{HMSM}) - f(\mathbf{x}^*))}{m}} \sqrt{1 - \rho^2}^k$$

i time završavamo dokaz o linerarnoj konvergenciji niza $\{\mathbf{x}_k^{HMSM}\}$.

(b) Dokazuje se na sličan način. $\square$

Teorema 4.5.3. *Neka su zadovoljene pretpostavke (H_1) i (H_2) u vezi sa (4.40) i f je uniformno konveksna funkcija.*

(a) *Niz $\{\mathbf{x}_k^{DMSM}\}$ generisan Algoritmom 4.4.4 zadovoljava*

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{DMSM}\| = 0$$

i $\{\mathbf{x}_k^{DMSM}\}$ konvergira ka $\mathbf{x}^$.*

(b) *Niz $\{\mathbf{x}_k^{TMSM}\}$ generisan Algoritmom 4.4.3 zadovoljava*

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{TMSM}\| = 0$$

i $\{\mathbf{x}_k^{TMSM}\}$ konvergira ka $\mathbf{x}^$.*

Dokaz. Analogno dokazu u [155, Teorema 4.1]. $\square$

Posledica 4.5.1. *Ako su zadovoljene pretpostavke (H_1) i (H_2) u vezi sa (4.40) i $f : \mathbb{R}^n \mapsto \mathbb{R}$ je uniformno konveksna funkcija, nejednakost oblika (4.38) važi za nizove $\{\mathbf{x}_k\}$ generisani HGD (4.10), HSM (3.36) i HAGD (4.13) metodama, gde je μ definisano kao u (4.42). Takođe,*

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k\| = 0,$$

i niz $\{\mathbf{x}_k\}$ konvergira ka optimalnoj vrednosti $\mathbf{x}^$ najmanje linearno.*

U nastavku odeljka želimo da dokažemo konvergenciju MSM metode na klasi strogo konveksnih kvadratnih funkcija. Zato je potrebno postaviti dodatan uslov za najmanju i

najveću sopstvenu vrednost matrice A koji garantuje uspešnu primenu MSM iterativne šeme (4.4) na strogo konveksne kvadratne funkcije opšteg oblika

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T A \mathbf{x} - b^T \mathbf{x}, \quad (4.49)$$

gde je A realna $n \times n$ simetrična pozitivno definitna matrica, a $b \in \mathbb{R}^n$ vektor. Označimo sopstvene vrednosti simetrične pozitivno definitne matrice A sa $\lambda_1, \lambda_2, \dots, \lambda_n$ i pri tom neka važi $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$.

Lema 4.5.5. *Nejednakost*

$$\lambda_1 \leq \frac{\gamma_{k+1}^{MSM}}{\alpha_{k+1}} \leq \frac{2\lambda_n}{\sigma}, k \in \mathbb{N}, \quad (4.50)$$

gde su λ_1 i λ_n najmanja i najveća sopstvena vrednost matrice A , važi za strogo konveksnu kvadratnu funkciju f opšteg oblika (4.49) koji uključuje simetričnu pozitivno definitnu matricu $A \in \mathbb{R}^{n \times n}$, metod opadajućeg gradijenta (4.4) sa parametrom γ_k^{MSM} određenim u skladu sa (4.8) i dužinu primarnog koraka α_k određenu Algoritmom 2.2.3.

Dokaz. Razlika između dve uzastopne vrednosti funkcije f definisane u (4.49) iznosi

$$f(\mathbf{x}_{k+1}^{MSM}) - f(\mathbf{x}_k^{MSM}) = \frac{1}{2} (\mathbf{x}_{k+1}^{MSM})^T A \mathbf{x}_{k+1}^{MSM} - b^T \mathbf{x}_{k+1}^{MSM} - \frac{1}{2} (\mathbf{x}_k^{MSM})^T A \mathbf{x}_k^{MSM} + b^T \mathbf{x}_k^{MSM}. \quad (4.51)$$

Zamenom (4.4) u (4.51) i sređivanjem izraza dobija se

$$\begin{aligned} f(\mathbf{x}_{k+1}^{MSM}) - f(\mathbf{x}_k^{MSM}) &= \frac{1}{2} [\mathbf{x}_k^{MSM} - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM}]^T A \\ &\quad [\mathbf{x}_k^{MSM} - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM}] \\ &\quad - b^T [\mathbf{x}_k^{MSM} - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM}] \\ &\quad - \frac{1}{2} (\mathbf{x}_k^{MSM})^T A \mathbf{x}_k^{MSM} + b^T \mathbf{x}_k^{MSM} \\ &= -\frac{1}{2} (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} (\mathbf{x}_k^{MSM})^T A \mathbf{g}_k^{MSM} \\ &\quad - \frac{1}{2} (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} (\mathbf{g}_k^{MSM})^T A \mathbf{x}_k^{MSM} \\ &\quad + \frac{1}{2} (\alpha_k + \alpha_k^2 - \alpha_k^3)^2 (\gamma_k^{MSM})^{-2} (\mathbf{g}_k^{MSM})^T A \mathbf{g}_k^{MSM} \\ &\quad + (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} b^T \mathbf{g}_k^{MSM}. \end{aligned}$$

Kako je $\mathbf{g}_k^{MSM} = A \mathbf{x}_k^{MSM} - b$ gradijent funkcije (4.49) i kako važi $(\mathbf{g}_k^{MSM})^T A \mathbf{g}_k^{MSM} =$

$\mathbf{g}_k^{MSM} A (\mathbf{g}_k^{MSM})^T$, može se pokazati da važi:

$$\begin{aligned}
 f(\mathbf{x}_{k+1}^{MSM}) - f(\mathbf{x}_k^{MSM}) &= (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} [b^T \mathbf{g}_k^{MSM} - (\mathbf{x}_k^{MSM})^T A \mathbf{g}_k^{MSM}] \\
 &\quad + \frac{1}{2}(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 (\gamma_k^{MSM})^{-2} (\mathbf{g}_k^{MSM})^T A \mathbf{g}_k^{MSM} \\
 &= (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} [b^T - (\mathbf{x}_k^{MSM})^T A] \mathbf{g}_k^{MSM} \\
 &\quad + \frac{1}{2}(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 (\gamma_k^{MSM})^{-2} (\mathbf{g}_k^{MSM})^T A \mathbf{g}_k^{MSM} \\
 &= -(\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} (\mathbf{g}_k^{MSM})^T \mathbf{g}_k^{MSM} \\
 &\quad + \frac{1}{2}(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 (\gamma_k^{MSM})^{-2} (\mathbf{g}_k^{MSM})^T A \mathbf{g}_k^{MSM}.
 \end{aligned} \tag{4.52}$$

Nakon zamene (4.52) u (4.8), parametar γ_{k+1}^{MSM} postaje

$$\begin{aligned}
 \gamma_{k+1}^{MSM} &= 2\gamma_k^{MSM} \frac{\gamma_k^{MSM} [f(\mathbf{x}_{k+1}^{MSM}) - f(\mathbf{x}_k^{MSM})] + (\alpha_k + \alpha_k^2 - \alpha_k^3) \|\mathbf{g}_k^{MSM}\|^2}{(\alpha_k + \alpha_k^2 - \alpha_k^3)^2 \|\mathbf{g}_k^{MSM}\|^2} \\
 &= \frac{(\mathbf{g}_k^{MSM})^T A \mathbf{g}_k^{MSM}}{\|\mathbf{g}_k^{MSM}\|^2}.
 \end{aligned}$$

Poslednja jednačina za γ_{k+1}^{MSM} predstavlja Rejljev količnik (*eng. Rayleigh quotient*) realne simetrične matrice A i vektora $\mathbf{g}_k^{MSM}$. Prema tome, sledeća nejednakost važi:

$$\lambda_1 \leq \gamma_{k+1}^{MSM} \leq \lambda_n, \quad k \in \mathbb{N}. \tag{4.53}$$

Dokaz leve nejednakosti u (4.50) sledi neposredno iz (4.53), budući da je $0 < \alpha_{k+1} \leq 1$. Da bismo dokazali desnu nejednakost u (4.50), koristimo gornju granicu iniciranu linijskim pretraživanjem

$$\alpha_k \geq \frac{\beta(1 - \sigma)\gamma_k}{L},$$

što dovodi do sledeće nejednakosti:

$$\frac{\gamma_{k+1}^{MSM}}{\alpha_{k+1}} < \frac{L}{\beta(1 - \sigma)}. \tag{4.54}$$

Uzimajući u obzir da je $\mathbf{g}(\mathbf{x}) = A\mathbf{x} - b$ gradijent funkcije (4.49) u kombinaciji sa pretpostavkom da je realna matrica A simetrična, sledi da je:

$$\|\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{y})\| = \|A\mathbf{x} - A\mathbf{y}\| = \|A(\mathbf{x} - \mathbf{y})\| \leq \|A\| \|\mathbf{x} - \mathbf{y}\| = \lambda_n \|\mathbf{x} - \mathbf{y}\|. \tag{4.55}$$

Na osnovu poslednje jednačine možemo zaključiti da Lipšicovu konstantu L u (4.54) možemo zameniti vrednošću najveće sopstvene vrednosti λ_n matrice A . U Algoritmu 2.2.3 linijskog pretraživanja unazad parametri σ i β su izabrani tako da važi $\sigma \in (0, 0.5)$ i $\beta \in (\sigma, 1)$.

Imajući u vidu ova ograničenja, sledi da je

$$\frac{\gamma_{k+1}^{MSM}}{\alpha_{k+1}} < \frac{L}{\beta(1-\sigma)} = \frac{\lambda_n}{\beta(1-\sigma)} < \frac{2\lambda_n}{\sigma}. \quad (4.56)$$

Prema tome, dokazana je i desna strana nejednakosti (4.50), čime je dokaz završen. $\square$

Konvergencija MSM metode pod dodatnim pretpostavkama $\lambda_n < 2\lambda_1$ se dokazuje u Teoremi 4.5.4 za strogo konveksne kvadratne funkcije.

Teorema 4.5.4. *Neka je f strogo konveksna kvadratna funkcija oblika (4.49). Ako sopstvene vrednosti matrice A zadovoljavaju dodatnu pretpostavku $\lambda_n < 2\lambda_1$, onda MSM metod (4.4) zadovoljava*

$$(d_i^{k+1})^2 \leq \delta^2 (d_i^k)^2, \quad (4.57)$$

gde je

$$\delta = \max \left\{ 1 - \frac{\sigma\lambda_1}{2\lambda_n}, \frac{\lambda_n}{\lambda_1} - 1 \right\}. \quad (4.58)$$

Pored toga

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{MSM}\| = 0. \quad (4.59)$$

Dokaz. Neka je $\{\mathbf{x}_k^{MSM}\}$ niz vrednosti dobijen primenom Algoritma 4.2.1. Pretpostavimo da su $\{v_1, v_2, \dots, v_n\}$ ortonormirani sopstveni vektori simetrične pozitivno definitne matrice A . Tada za proizvoljni vektor $\mathbf{x}_k^{MSM}$, koristeći $\mathbf{g}_k^{MSM} = A\mathbf{x}_k^{MSM} - b$, postoje realne konstante $d_1^k, d_2^k, \dots, d_n^k$ takve da

$$\mathbf{g}_k^{MSM} = \sum_{i=1}^n d_i^k v_i. \quad (4.60)$$

Sada, koristeći (4.4), možete jednostavno potvrditi sledeće

$$\begin{aligned} \mathbf{g}_{k+1}^{MSM} &= A\mathbf{x}_{k+1}^{MSM} - b \\ &= A(\mathbf{x}_k^{MSM} - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \mathbf{g}_k^{MSM}) - b \\ &= \mathbf{g}_k^{MSM} - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} A\mathbf{g}_k^{MSM} \\ &= (I - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} A) \mathbf{g}_k^{MSM}. \end{aligned}$$

Koristeći jednostavnu linearnu reprezentaciju za $\mathbf{g}_{k+1}^{MSM}$ oblika (4.60), dobija se

$$\mathbf{g}_{k+1}^{MSM} = \sum_{i=1}^n d_i^{k+1} v_i = \sum_{i=1}^n (1 - (\alpha_k + \alpha_k^2 - \alpha_k^3)(\gamma_k^{MSM})^{-1} \lambda_i) d_i^k v_i. \quad (4.61)$$

Da bi dokazali (4.57), dovoljno je pokazati da $\left| 1 - \frac{\lambda_i}{\gamma_k^{MSM}(\alpha_k + \alpha_k^2 - \alpha_k^3)^{-1}} \right| \leq \delta$. Postoje dva slučaja. U prvom slučaju pretpostavimo da je $\lambda_i \leq \frac{\gamma_k^{MSM}}{\alpha_k + \alpha_k^2 - \alpha_k^3}$. Primenivši (4.50), možemo

zaključiti sledeće:

$$1 > \frac{\lambda_i}{\gamma_k^{MSM}(\alpha_k + \alpha_k^2 - \alpha_k^3)^{-1}} \geq \frac{\sigma \lambda_1}{2\lambda_n} \implies 1 - \frac{\lambda_i}{\gamma_k^{MSM}(\alpha_k + \alpha_k^2 - \alpha_k^3)^{-1}} \leq 1 - \frac{\sigma \lambda_1}{2\lambda_n} \leq \delta. \quad (4.62)$$

Ispitajmo sada drugi slučaj $\frac{\gamma_k^{MSM}}{\alpha_k + \alpha_k^2 - \alpha_k^3} < \lambda_i$. Kako važi

$$1 < \frac{\lambda_i}{\gamma_k^{MSM}(\alpha_k + \alpha_k^2 - \alpha_k^3)^{-1}} \leq \frac{\lambda_n}{\lambda_1}, \quad (4.63)$$

sledi da

$$\left| 1 - \frac{\lambda_i}{\gamma_k^{MSM}(\alpha_k + \alpha_k^2 - \alpha_k^3)^{-1}} \right| \leq \frac{\lambda_n}{\lambda_1} - 1 \leq \delta. \quad (4.64)$$

Sada da bi dokazali $\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{MSM}\| = 0$, koristimo ortonormalnost sopstvenih vektora $\{v_1, v_2, \dots, v_n\}$ kao i (4.60) i dobija se

$$\|\mathbf{g}_k^{MSM}\|^2 = \sum_{i=1}^n (d_i^k)^2. \quad (4.65)$$

Kako je (4.57) zadovoljeno i važi $0 < \delta < 1$, s obzirom na (4.65) sledi da važi (4.60), čime je dokaz završen. $\square$

Svojstva konvergencije hibridnih metoda (HMAGD i HMSM) i metoda sa višestrukom upotrebom linijskog pretraživanja unazad u MSM metodi (DMSM i TMSM) u slučaju strogo konveksnih kvadratnih funkcija mogu se dokazati analogno na osnovu Leme 4.5.5 i Teoreme 4.5.4.

4.6 Numerički rezultati

U ovom odeljku ispitujemo efikasnost predloženih MAGD, MSM, HMAGD, HMSM, DMSM i TMSM metoda. Odeljak se sastoji od dva pododeljka, gde se u jednom ispituje efikasnost MAGD, MSM, HMAGD i HMSM metoda, a u drugom se ispituje efikasnost višestruke upotrebe linijskog pretraživanja unazad u DMSM i TMSM metodama.

Sve navedene metode su implementirane u Matlab R2017a programskom jeziku i izvršene na laptop računaru sa Intel Core i3 2.0 GHz CPU-om, 8 GB RAM-a i Windows 10 operativnim sistemom. U numeričkim eksperimentima analizirana su tri indikatora (svojstva) za svaku testiranu metodu: broj iterativnih koraka (broj iteracija), ukupan broj izračunavanja vrednosti funkcije u različitim tačkama (broj evaluacija funkcije) i vreme izvršavanja algoritma (CPU vreme). Numerička testiranja su bazirana na test funkcijama iz [7], gde su mnoge test funkcije preuzete iz CUTER kolekcije [26].

Za svaki testirani algoritam korišćeni su isti kriterijumi zaustavljanja programa:

$$\|\mathbf{g}_k\| \leq 10^{-6} \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq 10^{-16}.$$

Za ulazne parametre u Algoritmu 2.2.3 linijskog pretraživanja unazad za sve testirane algoritme korišćene su vrednosti $\sigma = 0.0001$ i $\beta = 0.8$. Što znači da prihvatamo malo smanjenje ciljne funkcije f predviđeno linearnom aproksimacijom u trenutnoj tački. Dok za hibridne metode (*HMAGD*, *HMSM* i *HSM*) korišćena je vrednost parametra korekcije $\Omega_k = 0.1$.

Da bi imali jasnu vizualizaciju efikasnosti predloženih algoritama, koristimo profile performansi Dolana (*Dolan*) i Morea (*More*) iz [58]. Ovi profili se mogu smatrati alatom za procenu i poređenje performansi iterativnih metoda, gde se profil svake metode meri na osnovu odnosa njenog proračunskog ishoda u odnosu na računski ishod najbolje predstavljene metode. Gornja kriva odgovara metodi koja pokazuje najbolje performanse u odnosu na izabrani profil performansi.

4.6.1 Numerički rezultati za MAGD, MSM, HMAGD i HMSM metode

U ovom odeljku prezentovani su i analizirani numerički rezultati dobijeni testiranjem MAGD, MSM, HMAGD, HMSM, SM i HSM metoda. Prezentovanje i analiza numeričkih rezultata je izvršena na osnovu tri numerička eksperimenta.

Važno je napomenuti da je HSM metod u radu [129] postigao mnogo bolje rezultate od SM metode iz [155]. Pored toga, SM metod u [155] je postigao mnogo bolje rezultate od AGD i GD metoda. Na osnovu navedenih činjenica, izabran je HSM metod za upoređenje sa MAGD, MSM, HMAGD i HMSM metodama u numeričkom Eksperimentu 4.6.1.

Eksperiment 4.6.1. Numerički eksperiment je baziran na 28 test funkcija iz [7] za testiranje optimizacionih metoda velikih dimenzija. Za svaku od testiranih funkcija urađeno je 12 numeričkih testova sa različitim izborom broja promenljivih: 100, 200, 300, 500, 1000, 2000, 3000, 5000, 7000, 8000, 10000 i 15000. Zbirni numerički rezultati za MAGD, MSM, HMAGD, HMSM i HSM metod, testirani na 28 test funkcijama velikih dimenzija, predstavljani su u Tabelama 4.1, 4.2 i 4.3.

U Tabeli 4.4 prikazani su prosečni numerički rezultati dobijeni na osnovu 336 numeričkih testova za broj iteracija, broj evaluacija ciljne funkcije i CPU vreme.

Na osnovu rezultata datih u Tabeli 4.4, očigledno je da MSM metod postiže drastično bolje rezultate u poređenju sa MAGD, HMAGD, HMSM i HSM metodama. Ovaj zaključak potvrđuju i profili performansi za broj iteracija, broj evaluacija funkcije i CPU vreme.

Slika 4.1a prikazuje profile performansi MAGD, HMAGD, MSM, HMSM i HSM metoda u odnosu na broj iteracija. Slika 4.1b ilustruje profile performansi MAGD, HMAGD, MSM, HMSM i HSM metoda u odnosu na broj evaluacija funkcije, dok Slika 4.2 prikazuje profile performansi MAGD, HMAGD, MSM, HMSM i HSM metoda u odnosu na CPU vreme. Kako

Tabela 4.1: Zbirni numerički rezultati MAGD, HMAGD, MSM, HMSM i HSM metoda za broj iterativnih koraka.

Test funkcija	Broj iteracija				
	<i>MAGD</i>	<i>HMAGD</i>	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Extended Penalty	341	391	689	744	684
Perturbed Quadratic	352325	356402	34828	84420	79198
Raydan 2	60	119	108	160	160
Diagonal 3	119719	124750	7030	21287	20626
Generalized Tridiagonal 1	647	653	346	443	422
Extended Tridiagonal 1	692219	755340	1370	82248	74898
Extended TET	455	455	156	261	286
Diagonal 4	8084	10590	96	1681	2055
Diagonal 5	48	109	72	120	120
Extended Himmelblau	302	400	260	693	358
Perturbed quadratic diagonal	1060824	5965848	37454	196373	155484
Quadratic QF1	362896	368183	36169	89026	78932
Extended quadratic penalty QP1	229	275	369	340	374
Extended quadratic penalty QP2	356357	84634	1674	22385	20432
Quadratic QF2	71647	388352	32727	90357	89593
Extended quadratic exponential EP1	67	128	100	268	193
Extended Tridiagonal 2	1665	1721	659	710	778
ARWHEAD (CUTE)	12834	71741	430	4261	4151
Almost Perturbed Quadratic	354369	358466	33652	84546	79701
LIARWHD (CUTE)	925138	1963100	3029	271705	244509
ENGVAL1 (CUTE)	822	821	461	561	522
QUARTC (CUTE)	177	165	217	292	256
Diagonal 6	60	119	108	162	160
Generalized Quartic	229	270	181	209	226
Diagonal 7	159	216	147	266	209
Diagonal 8	154	216	120	202	177
Full Hessian FH3	63	153	63	207	186
Diagonal 9	325609	614270	10540	79802	63237

gornja kriva odgovara metodi koja pokazuje najbolje performanse u odnosu na izabrani profil performansi, ovde je očigledno da MSM metod postiže najbolje rezultate.

Na Slici 4.1a se može primetiti da svih pet metoda uspešno rešavaju sve probleme, a MSM metod je najbolji u 75.0% testiranih problema u poređenju sa MAGD(25.0%), HMAGD(3.6%), HMSM(0%) i HSM(0%).

Na Slici 4.1b očigledno je da svih pet metoda uspešno rešavaju sve probleme, dok je MSM metod najbolji u 75.0% testiranih problema u poređenju sa MAGD(10.7%), HMAGD(3.6%), HMSM(7.1%) i HSM(3.6%).

Grafici na Slici 4.2 pokazuju da svih pet metoda uspešno rešavaju sve probleme, a MSM

Tabela 4.2: Zbirni numerički rezultati MAGD, HMAGD, MSM, HMSM i HSM metoda za broj evaluacija testirane funkcije.

Test funkcija	Broj evaluacija funkcije				
	<i>MAGD</i>	<i>HMAGD</i>	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Extended Penalty	9085	11702	3479	3638	3460
Perturbed Quadratic	13855459	14193163	200106	366943	334564
Raydan 2	132	250	228	332	332
Diagonal 3	4244404	4482972	38158	93632	88698
Generalized Tridiagonal 1	9057	9616	1191	1396	1330
Extended Tridiagonal 1	2077341	3021492	10989	425411	387939
Extended TET	4130	4168	528	753	818
Diagonal 4	133440	185397	636	8140	9517
Diagonal 5	108	230	156	253	253
Extended Himmelblau	5192	7164	976	2754	1172
Perturbed quadratic diagonal	38728371	236316190	341299	1018378	807185
Quadratic QF1	13192789	13541108	208286	387021	332928
Extended quadratic penalty QP1	2939	3747	2196	1846	2141
Extended quadratic penalty QP2	8846145	2282567	11491	116071	105841
Quadratic QF2	2810965	16640880	183142	394364	378921
Extended quadratic exponential EP1	1513	2878	894	2500	1716
Extended Tridiagonal 2	9613	10916	2866	2793	3010
ARWHEAD (CUTE)	468970	2847637	5322	27050	28015
Almost Perturbed Quadratic	13936462	14275979	194876	367586	336419
LIARWHD (CUTE)	41619197	90302744	27974	1409648	1269240
ENGVAL1 (CUTE)	8332	8531	2285	2956	2700
QUARTC (CUTE)	414	402	494	644	572
Diagonal 6	132	275	270	362	356
Generalized Quartic	1244	1696	493	526	592
Diagonal 7	745	1187	504	756	672
Diagonal 8	740	1064	383	753	589
Full Hessian FH3	1955	5508	566	1898	1541
Diagonal 9	12984028	25166521	68189	392059	307951

metod je najbolji u 78.6% testiranih problema u poređenju sa MAGD(10.7%), HMAGD(0%), HMSM(10.7%) i HSM(3.6%).

Na osnovu zbirnih numeričkih rezultata prikazanih u Tabelama 4.1, 4.2 i 4.3 i na osnovu profila performansi odgovarajućih metoda prikazanih na Slikama 4.1 i 4.2, možemo zaključiti sledeće:

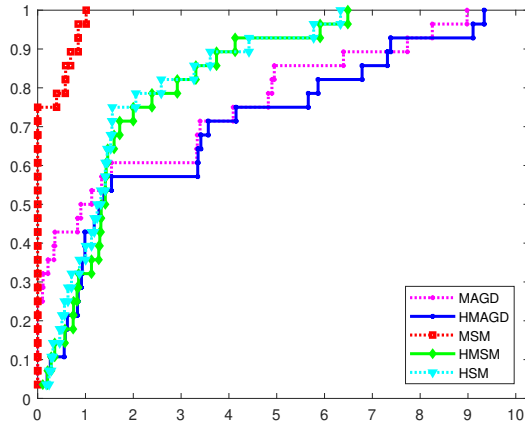
- (i) MSM metod daje bolje rezultate u poređenju sa MAGD, HMAGD, HMSM i HSM metodama u pogledu sva tri razmatrana indikatora performansi: broj iteracija, broj evaluacija funkcije i CPU vreme.

Tabela 4.3: Zbirni numerički rezultati MAGD, HMAGD, MSM, HMSM i HSM metoda za CPU vreme u sekundama.

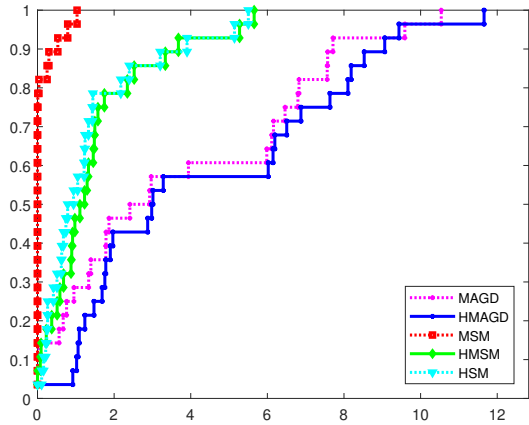
Test funkcija	CPU vreme				
	<i>MAGD</i>	<i>HMAGD</i>	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Extended Penalty	2.69	3.19	1.59	1.38	1.53
Perturbed Quadratic	6049.53	6432.67	116.28	210.17	210.42
Raydan 2	0.17	0.23	0.23	0.20	0.28
Diagonal 3	6401.97	7049.88	52.61	153.34	155.52
Generalized Tridiagonal 1	7.78	7.22	1.47	1.83	1.58
Extended Tridiagonal 1	8853.17	11247.73	29.05	1121.67	1018.38
Extended TET	2.77	2.50	0.52	0.80	0.91
Diagonal 4	16.17	22.34	0.20	1.70	1.86
Diagonal 5	0.31	0.44	0.34	0.55	0.42
Extended Himmelblau	1.03	1.36	0.30	0.66	0.31
Perturbed quadratic diagonal	22820.17	102830.22	139.63	476.63	277.48
Quadratic QF1	6846.45	7960.61	81.53	155.34	128.80
Extended quadratic penalty QP1	1.06	1.25	1.00	0.92	0.84
Extended quadratic penalty QP2	1872.80	532.38	3.52	20.11	18.09
Quadratic QF2	768.56	5263.98	73.44	169.16	158.72
Extended quadratic exponential EP1	0.84	1.05	0.69	1.17	1.17
Extended Tridiagonal 2	2.53	3.34	1.05	0.97	1.22
ARWHEAD (CUTE)	138.00	1627.41	1.97	11.53	13.56
Almost Perturbed Quadratic	7086.56	8258.72	73.05	148.22	131.58
LIARWHD (CUTE)	15372.63	32393.92	9.25	707.83	635.06
ENGVAL1 (CUTE)	2.64	2.47	1.05	1.42	1.27
QUARTC (CUTE)	2.08	1.91	1.84	2.34	2.30
Diagonal 6	0.14	0.38	0.23	0.36	0.33
Generalized Quartic	0.50	0.70	0.28	0.28	0.38
Diagonal 7	0.69	1.02	0.55	0.80	0.84
Diagonal 8	0.66	1.08	0.47	1.13	0.67
Full Hessian FH3	1.19	3.13	0.39	1.72	1.39
Diagonal 9	6662.98	7734.27	43.61	219.52	104.09

Tabela 4.4: Prosečni numerički rezultati za 28 test funkcije testiranih na 12 različitim numeričkih testova.

Prosečne performanse	<i>MAGD</i>	<i>HMAGD</i>	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Broj iteracija	165,982.11	395,281.68	7,251.96	36,918.89	32,783.11
Broj evaluacija funkcije	5,462,603.64	15,118,785.14	46,713.46	179,659.39	157,445.43
CPU vreme (sec)	2,961.29	6,835.19	22.72	121.85	102.46

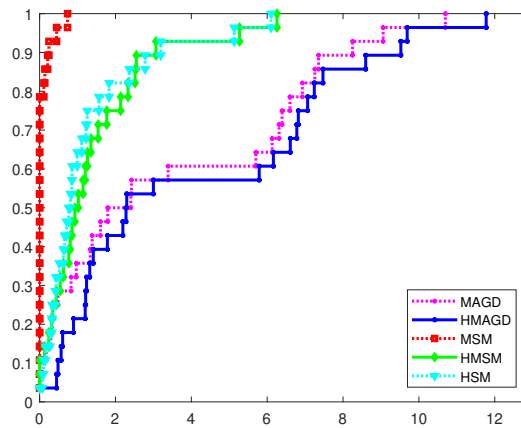


(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 4.1: Profili performansi za MAGD, HMAGD, MSM, HMSM i HSM metod.



Slika 4.2: Profili performansi bazirani na CPU vremenu za MAGD, HMAGD, MSM, HMSM i HSM metod.

- (ii) Ubrzani metod pokazuje bolje performanse od hibridnog metoda. Preciznije, ovo znači da je MAGD metod bolji od HMAGD metode kao i MSM metod u odnosu na HSM metod.
- (iii) Klasa SM metoda (MSM, HSM i HMSM) pokazuje bolje performanse od klase AGD metoda (MAGD i HMAGD).

Kako u Eksperimentu 4.6.1 MSM, HMSM i HSM metod imaju drastično bolje numeričke rezultate od MAGD i HMAGD metoda u sledećem numeričkom eksperimentu vršićemo testiranje MSM, HMSM i HSM metoda na test funkcijama sa većim brojem promenljivih. Cilj ovog numeričkog eksperimenta je da se istraži da li povećanje broja promenljivih u test funkcijama utiče na efikasnost MSM, HMSM i HSM metoda.

Eksperiment 4.6.2. Numerički eksperiment je baziran na 30 test funkcija iz [7]. Za svaku od testiranih funkcija urađeno je 11 numeričkih testova sa različitim izborom broja promenljivih: 1000, 2000, 3000, 5000, 7000, 8000, 10000, 15000, 20000, 30000 i 50000. Zbirni numerički rezultati za MSM, HMSM i HSM metod dobijeni testiranjem 30 test funkcija, predstavljeni su u Tabelama 4.5, 4.6 i 4.7.

Tabela 4.5: Zbirni numerički rezultati MSM, HMSM i HSM metoda za broj iterativnih koraka.

Test funkcija	Broj iteracija		
	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Extended Penalty	747	777	795
Perturbed Quadratic	90056	216249	208908
Raydan 2	99	146	146
Diagonal 2	61480	2761	57062
Diagonal 3	9143	31331	32541
Generalized Tridiagonal 1	310	409	381
Extended Tridiagonal 1	1557	111650	98106
Extended TET	143	231	254
Diagonal 4	88	1606	1958
Diagonal 5	66	110	110
Perturbed quadratic diagonal	66898	325007	270034
Quadratic QF1	94931	236085	219454
Extended quadratic penalty QP1	408	353	400
Extended quadratic penalty QP2	1061	8496	7627
Quadratic QF2	89646	246260	244162
Extended quadratic exponential EP1	93	280	211
Extended Tridiagonal 2	650	654	711
ARWHEAD (CUTE)	431	4396	4431
Almost Perturbed Quadratic	88120	222865	219976
LIARWHD (CUTE)	6201	756758	709335
ENGVAL1 (CUTE)	428	552	502
QUARTC (CUTE)	210	277	244
Diagonal 6	99	151	149
COSINE (CUTE)	220	175767	248
Generalized Quartic	165	193	220
Diagonal 7	165	296	204
Diagonal 8	114	202	197
Full Hessian FH3	55	233	228
HIMMELH (CUTE)	110	99	99
Extended Rosenbrock	55	55	55

Na osnovu podataka datih u Tabelama 4.5, 4.6 i 4.7 nije teško zaključiti da MSM metod postiže bolje rezultate u odnosu na HMSM i HSM metod. Ukoliko posmatramo broj iteracija

Tabela 4.6: Zbirni numerički rezultati MSM, HMSM i HSM metoda za broj evaluacija testirane funkcije.

Test funkcija	Broj evaluacija funkcije		
	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Extended Penalty	4050	4125	4521
Perturbed Quadratic	531633	941045	883112
Raydan 2	209	303	303
Diagonal 2	400443	31069	266617
Diagonal 3	50629	137875	140393
Generalized Tridiagonal 1	1075	1357	1219
Extended Tridiagonal 1	13320	578079	508592
Extended TET	484	657	736
Diagonal 4	583	7775	9080
Diagonal 5	143	232	232
Perturbed quadratic diagonal	626251	1685791	1402532
Quadratic QF1	563538	1027186	927053
Extended quadratic penalty QP1	2396	2083	2454
Extended quadratic penalty QP2	6940	43453	39132
Quadratic QF2	504120	1072757	1032582
Extended quadratic exponential EP1	847	3147	2057
Extended Tridiagonal 2	2936	2599	2708
ARWHEAD (CUTE)	5844	29860	32806
Almost Perturbed Quadratic	525546	969555	929583
LIARWHD (CUTE)	65898	3926707	3685122
ENGVAL1 (CUTE)	2324	3044	2863
QUARTC (CUTE)	475	609	543
Diagonal 6	220	381	372
COSINE (CUTE)	652	775878	692
Generalized Quartic	451	485	579
Diagonal 7	621	1015	819
Diagonal 8	452	836	851
Full Hessian FH3	598	2486	2514
HIMMELH (CUTE)	231	209	209
Extended Rosenbrock	121	121	121

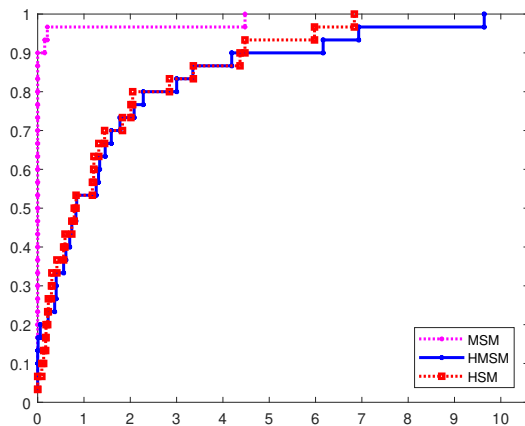
i broj evaluacija test funkcija MSM metod postiže najbolje rezultate na 26 odnosno na 25 test funkcija, dok HMSM metod postiže najbolje rezultate na 2 odnosno na 3 test funkcije. U jednom test slučaju HMSM i HSM imaju najbolji identični rezultat kada je u pitanju broj iteracija i broj evaluacija test funkcija, dok u drugom test slučaju sva tri metoda imaju identičan rezultat. Kada je u pitanju CPU vreme imamo sličnu situaciju da je MSM metod najbolji u 24 test funkcija, dok HMSM i HSM metod postižu najbolje rezultate na po 3 test funkcije.

Tabela 4.7: Zbirni numerički rezultati MSM, HMSM i HSM metoda za CPU vreme u sekundama.

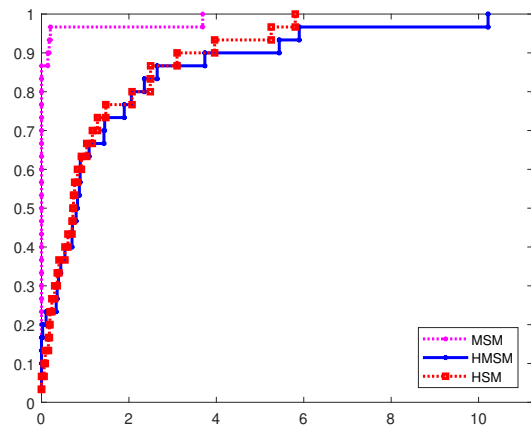
Test funkcija	CPU vreme (sec)		
	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Extended Penalty	3.69	3.91	4.91
Perturbed Quadratic	748.80	1364.23	1312.34
Raydan 2	0.48	0.63	0.66
Diagonal 2	802.17	37.55	613.45
Diagonal 3	128.64	410.58	478.38
Generalized Tridiagonal 1	3.00	4.08	3.39
Extended Tridiagonal 1	68.13	3513.39	2850.02
Extended TET	1.19	1.63	1.72
Diagonal 4	0.42	3.80	4.45
Diagonal 5	0.67	1.45	1.25
Perturbed quadratic diagonal	696.84	1916.58	1609.41
Quadratic QF1	614.67	1056.86	976.09
Extended quadratic penalty QP1	1.94	1.91	2.14
Extended quadratic penalty QP2	11.08	52.48	49.23
Quadratic QF2	548.59	1201.77	1168.66
Extended quadratic exponential EP1	1.13	3.69	2.36
Extended Tridiagonal 2	2.44	2.05	2.20
ARWHEAD (CUTE)	3.98	27.72	26.77
Almost Perturbed Quadratic	548.47	1010.25	984.58
LIARWHD (CUTE)	60.05	5138.77	4750.78
ENGVAL1 (CUTE)	3.02	3.34	2.72
QUARTC (CUTE)	4.20	5.28	5.34
Diagonal 6	0.56	0.89	0.75
COSINE (CUTE)	2.17	1776.70	1.75
Generalized Quartic	0.64	0.77	0.69
Diagonal 7	1.83	2.83	2.11
Diagonal 8	1.31	2.00	2.27
Full Hessian FH3	1.08	4.70	5.11
HIMMELH (CUTE)	1.13	1.08	1.03
Extended Rosenbrock	0.13	0.30	0.20

Grafici na Slikama 4.3 i 4.4 su kreirani na osnovu rezultata datih u Tabelama 4.5, 4.6 i 4.7. Slika 4.3a prikazuje performanse MSM, HMSM i HSM metoda u odnosu na broj iteracija. Na slici se može uočiti da MSM, HMSM i HSM metode uspešno rešavaju sve probleme, dok MSM metod postiže najbolje rezultate u 90.0% testiranih problema u poređenju sa HMSM i HSM. Slika 4.3b prikazuje performanse MSM, HMSM i HSM metoda u odnosu na broj evaluacija funkcije. Takođe, na ovoj slici se može primetiti da MSM, HMSM i HSM metoda uspešno rešavaju sve probleme, pri čemu MSM metod postiže najbolje rezultate u 86.7%

testiranih problema u poređenju sa HMSM i HSM.



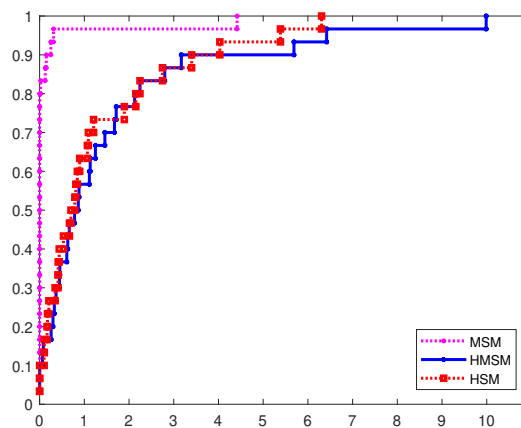
(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 4.3: Profili performansi za MSM, HMSM i HSM metod.

Slika 4.4 prikazuje performanse MSM, HMSM i HSM metoda u odnosu na CPU vreme. Na ovoj slici možemo uočiti da MSM, HMSM i HSM metoda uspešno rešavaju sve probleme, dok u 80.0% od svih testiranih problema MSM metod daje najbolje rezultate u poređenju sa HMSM i HSM.



Slika 4.4: Profili performansi bazirani na CPU vremenu za MSM, HMSM i HSM metod.

Tabela 4.8 sadrži prosečan broj iteracija, broj evaluacija funkcije i CPU vreme za 330 numeričkih testova.

Na osnovu prosečnih numeričkih rezultata prikazanih u Tabeli 4.8, možemo zaključiti da MSM metod u proseku ima tri i više puta bolje rezultate (broj iteracija, broj evaluacija funkcije i CPU vreme) od druga dva metoda.

Tabela 4.8: Prosečni numerički rezultati za 30 test funkcija testiranih na 11 numeričkih testova.

Prosečne performanse	<i>MSM</i>	<i>HMSM</i>	<i>HSM</i>
Broj iteracija	17,124.97	78,141.63	69,291.60
Broj evaluacija funkcije	110,434.33	375,023.97	329,346.57
CPU vreme (sec)	142.08	585.04	495.49

Nakon povećanja broja promenljivih sa 15000 na 50000 u drugom numeričkom eksperimentu, može se primetiti da odnos uspešnosti i brzine rešenih numeričkih test problema između MSM metode i HMSM i HSM metoda ostaje približno isti kao i u Eksperimentu 4.6.1. MSM metod ima tri i više puta bolje numeričke rezultate od druga dva metoda.

U sledećem numeričkom eksperimentu upoređićemo MSM i SM metod da bismo pokazali superiornost MSM metoda u pogledu sva tri indikatora: broja iteracija, broja evaluacija funkcije i CPU vremena.

Eksperiment 4.6.3. Numerički eksperiment je baziran na 30 test funkcija iz Eksperimenta 4.6.2. Pod istim uslovima kao u Eksperimentu 4.6.2 izvršeno je testiranje funkcija za MSM i SM metodu. Zbirni numerički rezultati za MSM i SM metod dobijeni testiranjem 30 test funkcija predstavljani su u Tabeli 4.9.

Grafici na Slikama 4.5 i 4.6 su kreirani na osnovu numeričkih rezultata datih u Tabeli 4.9. Slike 4.5a i 4.5b prikazuju performanse MSM i SM metoda u odnosu na broj iteracija i broj evaluacija funkcije, dok Slika 4.6 prikazuje performanse MSM i SM metoda u odnosu na CPU vreme. Na Slikama 4.5 i 4.6 se može uočiti da MSM i SM metod uspešno rešavaju sve probleme i da grafik MSM metode u sva tri slučaja prvi dostiže vrednost 1, odnosno izilazi na vrh što znači da je MSM metod efikasniji od SM metode.

Na osnovu numeričkih rezultata prezentovanih u Eksperimentima 4.6.1, 4.6.2 i 4.6.3 i njihove analize možemo zaključiti da je MSM metod najefikasniji metod među novim predloženim metodama, kao i od svih posmatranih metoda u ovom odeljku.

4.6.2 Numerički rezultati za TMSM i DMSM metode

U ovom odeljku ispituje se efikasnost višestruke upotrebe linijskog pretraživanja unazad u DMSM i TMSM metodama. Ispitivanje, analiza i prezentovanje numeričkih rezultata je izvršeno na osnovu dva numerička eksperimenta. U Eksperimentu 4.6.4 korišćene su vrednosti parametara koje su veće u drugom i trećem linijskom pretraživanju unazad od vrednosti parametara u primarnom linijskom pretraživanju unazad, dok u Eksperimentu 4.6.5 korišćene su vrednosti parametara koje su manje u drugom i trećem linijskom pretraživanju unazad od vrednosti parametara u primarnom linijskom pretraživanju unazad.

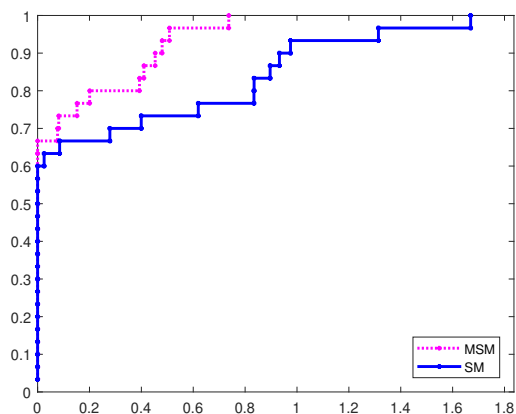
Tabela 4.9: Zbirni numerički rezultati MSM i SM metoda za broj iteracija, broj evaluacija funkcije i CPU vreme.

Test funkcija	Broj iteracija		Broj evaluacija funkcije		CPU vreme (sec)	
	<i>MSM</i>	<i>SM</i>	<i>MSM</i>	<i>SM</i>	<i>MSM</i>	<i>SM</i>
Extended Penalty	747	650	4050	2924	3.69	3.00
Perturbed Quadratic	90056	160542	531633	916964	748.80	1304.34
Raydan 2	99	99	209	209	0.48	0.59
Diagonal 2	61480	58288	400443	326472	802.17	970.42
Diagonal 3	9143	17019	50629	93596	128.64	242.94
Generalized Tridiagonal 1	310	293	1075	1020	3.00	3.86
Extended Tridiagonal 1	1557	3871	13320	32720	68.13	143.08
Extended TET	143	143	484	489	1.19	1.16
Diagonal 4	88	88	583	583	0.42	0.47
Diagonal 5	66	66	143	143	0.67	0.72
Perturbed quadratic diagonal	66898	81148	626251	853416	696.84	915.50
Quadratic QF1	94931	169298	563538	964865	614.67	1046.86
Extended Quad. penalty QP1	408	298	2396	3262	1.94	2.89
Extended Quad. penalty QP2	1061	1630	6940	11055	11.08	13.03
Quadratic QF2	89646	176214	504120	986671	548.59	1143.13
Extended Quad. exponential EP1	93	70	847	724	1.13	0.91
Extended Tridiagonal 2	650	457	2936	2727	2.44	2.19
ARWHEAD (CUTE)	431	309	5844	5308	3.98	6.06
Almost Perturbed Quadratic	88120	168157	525546	952654	548.47	1050.47
LIARWHD (CUTE)	6201	19725	65898	193633	60.05	106.20
ENGVAL1 (CUTE)	428	326	2324	2937	3.02	3.42
QUARTC (CUTE)	210	277	475	609	4.20	5.28
Diagonal 6	99	99	220	220	0.56	0.61
COSINE (CUTE)	220	198	652	636	2.17	1.63
Generalized Quartic	165	175	451	471	0.64	0.53
Diagonal 7	165	99	621	253	1.83	0.97
Diagonal 8	114	116	452	1127	1.31	2.61
Full Hessian FH3	55	55	598	608	1.08	1.14
HIMMELH (CUTE)	110	110	231	231	1.13	1.25
Extended Rosenbrock	55	55	121	121	0.13	0.25

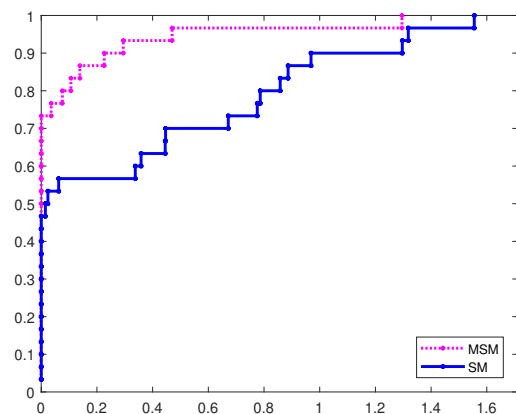
Cilj ovih numeričkih eksperimenata pored ispitivanja efikasnosti DMSM i TMSM metoda je i da se da odgovor na pitanje: Da li izbor većih ili manjih vrednosti parametara u drugom i trećem linijskom pretraživanju unazad u odnosu na primarno linijsko pretraživanje unazad direktno utiče na numeričke rezultate DMSM i TMSM metode?

Eksperiment 4.6.4. Numerički eksperiment je baziran na 24 test funkcija iz [7], gde su mnoge test funkcije preuzete iz CUTER kolekcije [26]. Za svaku od testiranih funkcija urađeno je 12 numeričkih testova sa 100, 200, 300, 500, 1000, 2000, 3000, 5000, 7000, 8000, 10000 i 15000 promenljivih. U Tabelama 4.10, 4.11 i 4.12 dati su zbirni numerički rezultati za 24 test funkcija za AGD, MAGD, MSM, SM, DMSM i TMSM metodu.

Ulazni parametri za primarno linijskog pretraživanje unazad u DMSM i TMSM metodi

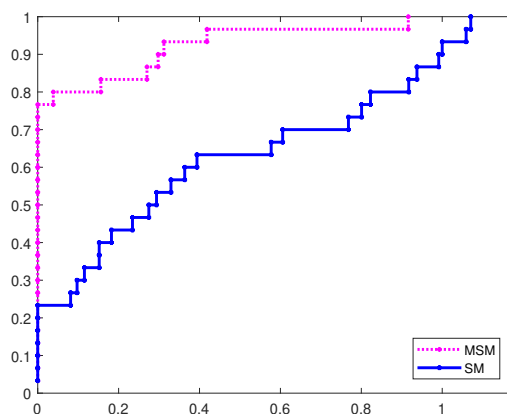


(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 4.5: Profili performansi za MSM i SM metod.



Slika 4.6: Profili performansi bazirani na CPU vremenu za MSM i SM metod.

imaju iste vrednosti kao i u svim ostalim testiranim metodama u numeričkim eksperimentima: $\sigma = 0.0001$ i $\beta = 0.8$ za Algoritam 2.2.3. Kada je u pitanju drugo i treće linijsko pretraživanje unazad ulazni parametri imaju sledeće vrednosti: $\sigma_l = 0.0002$ i $\beta_l = 0.9$ za Algoritam 4.4.1 i $\sigma_j = 0.00015$ i $\beta_j = 0.85$ za Algoritam 4.4.2.

Tabela 4.13 sadrži prosečne vrednosti broja iteracija, broja evaluacija funkcije i CPU vremena za 288 numeričkih testova.

Na osnovu vrednosti za broj iteracija datih u Tabeli 4.13, može se zaključiti da DMSM i TMSM metode postižu bolje numeričke rezultate u odnosu na MAGD, AGD, MSM i SM metode.

Profili performansi iz [58] su korišćeni za upoređivanje AGD, MAGD, MSM, SM, DMSM i TMSM metode. Slika 4.7a prikazuje profile performansi AGD, MAGD, SM, MSM, DMSM i TMSM metode koji se odnose na broj iteracija, dok Slika 4.7b ilustruje profile performansi

Tabela 4.10: Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj iterativnih koraka.

Test funkcija	Broj iteracija					
	<i>MAGD</i>	<i>TMSM</i>	<i>MSM</i>	<i>DMSM</i>	<i>SM</i>	<i>AGD</i>
Perturbed Quadratic	352325	31269	34828	31386	59908	353897
Raydan 1	58504	30148	26046	17238	14918	22620
Diagonal 3	119719	6767	7030	7077	12827	120416
Generalized Tridiagonal 1	647	332	346	350	325	670
Extended Tridiagonal 1	692219	685	1370	728	4206	3564
Extended TET	455	191	156	156	156	443
Diagonal 4	8084	96	96	96	96	120
Diagonal 5	48	72	72	72	72	48
Extended Himmelblau	302	312	260	264	196	396
Perturbed quadratic diagonal	1060824	36640	37454	31662	44903	2542050
Quadratic QF1	362896	32099	36169	33138	62927	366183
Extended quadratic penalty QP1	229	338	369	298	271	210
Extended quadratic penalty QP2	356357	1735	1674	990	3489	395887
Quadratic QF2	71647	31745	32727	30642	64076	100286
Extended Tridiagonal 2	1665	694	659	583	543	1657
ARWHEAD (CUTE)	12834	328	430	302	270	5667
Almost Perturbed Quadratic	354369	30790	33652	32902	60789	356094
LIARWHD (CUTE)	925138	1257	3029	1726	18691	1054019
ENGVAL1 (CUTE)	822	623	461	434	375	743
QUARTC (CUTE)	177	302	217	220	290	171
Generalized Quartic	229	191	181	186	189	187
Diagonal 7	159	144	147	111	108	72
Diagonal 8	154	120	120	109	118	60
Full Hessian FH3	63	63	63	63	63	45

u odnosu na broj evaluacija funkcije. Grafici na Slici 4.8 ilustruju ponašanje AGD, MAGD, SM, MSM, DMSM i TMSM metode u odnosu na CPU vreme.

Na Slici 4.7a se može uočiti da grafik DMSM metode prvi izilazi na vrh, tj. dostiže vrednost 1, što znači da DMSM metod nadmašuje druge razmatrane metode u odnosu na broj iteracija.

Profili performansi na Slici 4.7b potvrđuju da svih šest metoda uspešno rešavaju sve test probleme. Daljom analizom možemo uočiti da MSM metod ima najbolje rezultate u 58.33% svih testiranih problema, dok ostale metode imaju lošije rezultate: MAGD (4.17%), TMSM(0%), DMSM(4.17%), SM(29.17%) i AGD(16.67%).

Grafici na Slici 4.8 potvrđuju da su sve metode u stanju da reše sve testirane probleme, a MSM metod je najbolji u 54.17% testova u odnosu na MAGD (4.17%), TMSM(0%), DMSM(4.17%), SM(37.50%) i AGD(4.17%).

Na osnovu zbirnih numeričkih rezultata prikazanih u Tabelama 4.10, 4.11 i 4.12 i na

Tabela 4.11: Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj evaluacija funkcija.

Test funkcija	Broj evaluacija funkcije					
	<i>MAGD</i>	<i>TMSM</i>	<i>MSM</i>	<i>DMSM</i>	<i>SM</i>	<i>AGD</i>
Perturbed Quadratic	13855459	645704	200106	370595	337910	13916515
Raydan 1	1282162	1305952	311260	326766	81412	431804
Diagonal 3	4244404	131307	38158	80193	69906	4264718
Generalized Tridiagonal 1	9057	2934	1191	2061	1094	9334
Extended Tridiagonal 1	2077341	14797	10989	9147	35621	14292
Extended TET	4130	1689	528	948	528	3794
Diagonal 4	133440	2316	636	1320	636	1332
Diagonal 5	108	300	156	228	156	108
Extended Himmelblau	5192	3636	976	1908	668	6897
Perturbed quadratic diagonal	38728371	1309740	341299	629088	460028	94921578
Quadratic QF1	13192789	661661	208286	392426	352975	13310016
Extended quadratic penalty QP1	2939	6400	2196	5421	2326	2613
Extended quadratic penalty QP2	8846145	44962	11491	14058	25905	9852040
Quadratic QF2	2810965	642829	183142	364257	353935	3989239
Extended Tridiagonal 2	9613	9779	2866	4951	2728	8166
ARWHEAD (CUTE)	468970	15416	5322	8503	3919	214284
Almost Perturbed Quadratic	13936462	639129	194876	393591	338797	14003318
LIARWHD (CUTE)	41619197	39788	27974	33271	180457	47476667
ENGVAL1 (CUTE)	8332	10120	2285	4319	2702	6882
QUARTC (CUTE)	414	1412	494	780	640	402
Generalized Quartic	1244	1311	493	836	507	849
Diagonal 7	745	930	504	696	335	333
Diagonal 8	740	805	383	546	711	304
Full Hessian FH3	1955	2160	566	1263	631	1352

osnovu profila performansi na Slikama 4.7 i 4.8, može se zaključiti sledeće:

- (i) Metode DMSM i TMSM imaju bolje numeričke rezultate u poređenju sa drugim posmatranim metodama kada se upoređuju po broju iteracija.
- (ii) Metode SM, MSM, DMSM i TMSM pokazuju bolje performanse od AGD i MAGD metoda u pogledu sva tri indikatora (broj iteracija, CPU vreme i broj evaluacija funkcije).

Na osnovu podataka u Tabeli 4.10, prosečnih vrednosti u Tabeli 4.13, kao i na osnovu profila performansi na Slici 4.7a, može se izvesti zaključak da DMSM metod ima najbolje numeričke rezultate kada je u pitanju indikator broj iteracija. Takođe, može se izvesti i opštiji zaključak da metode sa višestrukom upotrebom linijskog pretraživanja unazad (DMSM i TMSM metod) su efikasnije od ostalih metoda kada se upoređuju na osnovu broja iteracija, dok imaju lošije numeričke rezultate od MSM i SM metode kada se upoređuju po broju evaluacija funkcija i CPU vremenu, ali bolje numeričke rezultate od AGD i MAGD metode.

Tabela 4.12: Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za CPU vreme u sekundama.

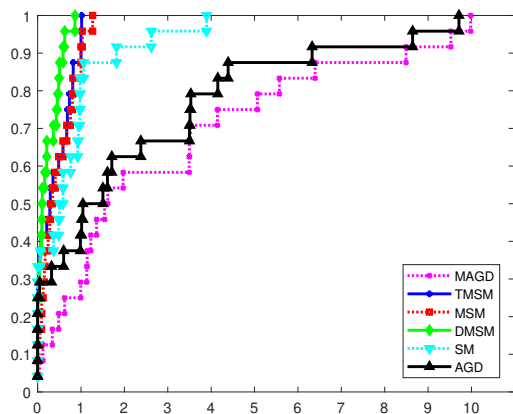
Test funkcija	CPU vreme					
	<i>MAGD</i>	<i>TMSM</i>	<i>MSM</i>	<i>DMSM</i>	<i>SM</i>	<i>AGD</i>
Perturbed Quadratic	6049.531	344.172	116.281	198.328	185.641	6756.047
Raydan 1	334.266	388.156	31.906	67.344	36.078	158.359
Diagonal 3	6401.969	199.547	52.609	120.406	102.875	5527.844
Generalized Tridiagonal 1	7.781	4.641	1.469	3.625	1.203	11.344
Extended Tridiagonal 1	8853.172	26.828	29.047	17.297	90.281	55.891
Extended TET	2.766	1.703	0.516	1.203	0.594	3.219
Diagonal 4	16.172	0.719	0.203	0.359	0.141	0.781
Diagonal 5	0.313	0.750	0.344	0.734	0.328	0.391
Extended Himmelblau	1.031	1.094	0.297	0.703	0.188	1.953
Perturbed quadratic diagonal	22820.172	534.750	139.625	273.188	185.266	44978.750
Quadratic QF1	6846.453	258.938	81.531	168.453	138.172	12602.563
Extended quadratic penalty QP1	1.063	2.234	1.000	3.516	0.797	1.266
Extended quadratic penalty QP2	1872.797	12.578	3.516	8.063	6.547	3558.734
Quadratic QF2	768.563	243.938	73.438	153.109	132.703	1582.766
Extended Tridiagonal 2	2.531	4.938	1.047	2.375	1.031	3.719
ARWHEAD (CUTE)	138.000	6.422	1.969	4.609	1.359	95.641
Almost Perturbed Quadratic	7086.563	285.563	73.047	153.891	133.516	13337.125
LIARWHD (CUTE)	15372.625	10.203	9.250	12.641	82.016	27221.516
ENGVAL1 (CUTE)	2.641	4.328	1.047	2.375	1.188	3.906
QUARTC (CUTE)	2.078	4.531	1.844	3.297	2.313	2.469
Generalized Quartic	0.500	0.734	0.281	0.375	0.188	0.797
Diagonal 7	0.688	0.953	0.547	1.469	0.375	0.625
Diagonal 8	0.656	0.781	0.469	1.078	0.797	0.438
Full Hessian FH3	1.188	1.672	0.391	1.234	0.391	1.438

Tabela 4.13: Prosečni numerički rezultati za 24 test funkcija testiranih na 12 numeričkih testova.

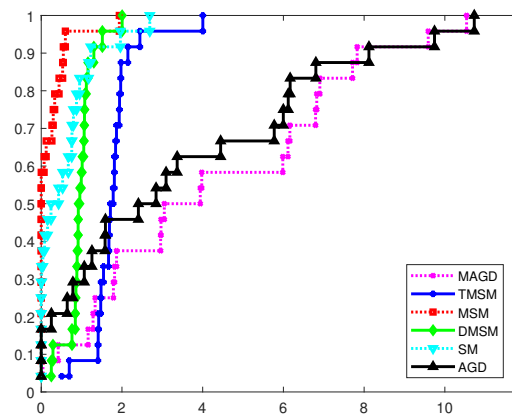
Prosečne performanse	MAGD	TMSM	MSM	DMSM	SM	AGD
Broj iteracija	182494.42	8622.54	9064.83	7947.21	14575.25	221896.04
Broj evaluacija funkcije	5885007.25	228961.54	64424.04	110298.83	93938.63	8434868.21
CPU vreme (sec)	3190.98	97.51	25.90	49.99	46.00	4829.4

Eksperiment 4.6.5. U poređenju sa numeričkim Eksperimentom 4.6.4 u ovom numeričkom eksperimentu koristimo vrednosti parametara u drugom i trećem linijskom pretraživanju unazad koje su manje od vrednosti u primarnom linijskom pretraživanju unazad.

Ulazni parametri za primarno linijskog pretraživanja unazad (Algoritam 2.2.3) imaju iste vrednosti $\sigma = 0.0001$ i $\beta = 0.8$ za sve posmatrane metode kao u prethodnom eksperi-

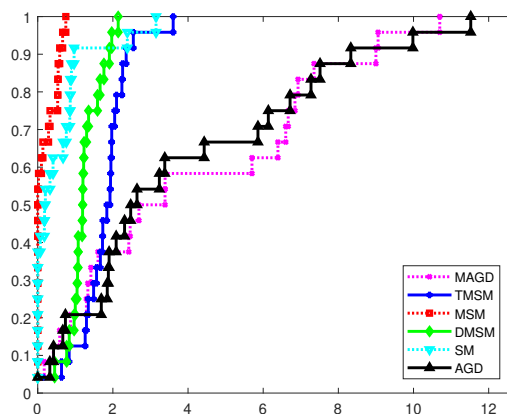


(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 4.7: Profili performansi za AGD, MAGD, MSM, SM, DMSM i TMSM metod.



Slika 4.8: Profili performansi bazirani na CPU vremenu za AGD, MAGD, MSM, SM, DMSM i TMSM metod.

mentu. Kada je u pitanju drugo i treće linijsko pretraživanje unazad ulazni parametri imaju sledeće vrednosti: $\sigma_l = 0.00001$ i $\beta_l = 0.6$ za Algoritam 4.4.1 i $\sigma_j = 0.00005$ i $\beta_j = 0.7$ za Algoritam 4.4.2. Ostali uslovi (kriterijumi zaustavljanja i broj promenljivih) ostaju isti kao u numeričkom Eksperimentu 4.6.4.

Dobijeni zbirni numerički rezultati dati su u Tabelama 4.14, 4.15 i 4.16, dok Tabela 4.17 sadrži prosečne vrednosti broja iteracija, broja evaluacija funkcije i CPU vremena. Na osnovu prosečnih vrednosti broja iteracija datih u Tabeli 4.17, može se primetiti da DMSM i TMSM metod daju bolje numeričke rezultate i u drugom numeričkom eksperimentu u poređenju sa MAGD, AGD, MSM i SM metodama.

Na osnovu numeričkih rezultata datih u Tabelama 4.14, 4.15 i 4.16 kreirani su profili performansi i prikazani na Slikama 4.9 i 4.10. Slika 4.9a prikazuje profile performansi AGD,

Tabela 4.14: Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj iteracija.

Test funkcija	Broj iteracija					
	<i>MAGD</i>	<i>MSM</i>	<i>SM</i>	<i>AGD</i>	<i>TMSM</i>	<i>DMSM</i>
Perturbed Quadratic	352325	34828	59908	353897	35697	28487
Raydan 1	58504	26046	14918	22620	9801	17594
Diagonal 3	119719	7030	12827	120416	8372	6409
Generalized Tridiagonal 1	647	346	325	670	342	348
Extended Tridiagonal 1	692219	1370	4206	3564	907	760
Extended TET	455	156	156	443	156	156
Diagonal 4	8084	96	96	120	96	96
Diagonal 5	48	72	72	48	72	72
Extended Himmelblau	302	260	196	396	288	294
Perturbed quadratic diagonal	1060824	37454	44903	2542050	31031	37331
Quadratic QF1	362896	36169	62927	366183	39619	26585
Extended quadratic penalty QP1	229	369	271	210	303	362
Extended quadratic penalty QP2	356357	1674	3489	395887	2047	1908
Quadratic QF2	71647	32727	64076	100286	39452	28651
Extended quadratic exponential EP1	67	100	73	48	107	107
Extended Tridiagonal 2	1665	659	543	1657	528	615
ARWHEAD (CUTE)	12834	430	270	5667	304	281
Almost Perturbed Quadratic	354369	33652	60789	356094	35755	26274
LIARWHD (CUTE)	925138	3029	18691	1054019	1340	3543
ENGVAL1 (CUTE)	822	461	375	743	418	482
QUARTC (CUTE)	177	217	290	171	289	275
Generalized Quartic	229	181	189	187	197	195
Full Hessian FH3	63	63	63	45	63	63
Diagonal 9	325609	10540	13619	329768	10219	11229

MAGD, MSM, SM, DMSM i TMSM metode koji se odnose na broj iteracija, dok Slika 4.9b prikazuje profile performansi u odnosu na broja evaluacija funkcije. Profili performansi na Slici 4.10 prikazuju ponašanje metoda u odnosu na CPU vreme.

Na osnovu podataka u Tabeli 4.14, prosečnih vrednosti u Tabeli 4.17, kao i na osnovu profila performansi na Slici 4.9a, može se izvesti opšti zaključak da metode sa višestrukom upotrebom linijskog pretraživanja unazad (DMSM i TMSM metod) su efikasnije od ostalih metoda kada se upoređuju na osnovu broja iteracija. Sa druge strane, na osnovu podataka u Tabelama 4.15 i 4.16, prosečnih vrednosti u Tabeli 4.17, kao i na osnovu profila performansi na Slikama 4.9b i 4.10 uočava se da DMSM i TMSM metodi imaju lošije numeričke rezultate od MSM i SM kada se upoređuju po broju evaluacija funkcija i CPU vremenu, ali bolje numeričke rezultate od AGD i MAGD.

Kako je naglašeno prilikom definisanja DMSM i TMSM metoda sa višestrukom upotrebom linijskog pretraživanja unazad u MSM metodi da se očekuje smanjenje broja iteracija

Tabela 4.15: Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj evaluacija funkcija.

Test funkcija	Broj evaluacija funkcije					
	<i>MAGD</i>	<i>MSM</i>	<i>SM</i>	<i>AGD</i>	<i>TMSM</i>	<i>DMSM</i>
Perturbed Quadratic	13855459	200106	337910	13916515	423496	260678
Raydan 1	1282162	311260	81412	431804	124905	280011
Diagonal 3	4244404	38158	69906	4264718	95962	54865
Generalized Tridiagonal 1	9057	1191	1094	9334	2408	2153
Extended Tridiagonal 1	2077341	10989	35621	14292	13562	6800
Extended TET	4130	528	528	3794	1080	828
Diagonal 4	133440	636	636	1332	1284	996
Diagonal 5	108	156	156	108	300	228
Extended Himmelblau	5192	976	668	6897	2136	2418
Perturbed quadratic diagonal	38728371	341299	460028	94921578	619938	529154
Quadratic QF1	13192789	208286	352975	13310016	472273	243573
Extended quadratic penalty QP1	2939	2196	2326	2613	5073	3895
Extended quadratic penalty QP2	8846145	11491	25905	9852040	29847	21345
Quadratic QF2	2810965	183142	353935	3989239	444580	257674
Extended quadratic exponential EP1	1513	894	661	990	2083	1617
Extended Tridiagonal 2	9613	2866	2728	8166	4446	4456
ARWHEAD (CUTE)	468970	5322	3919	214284	9038	6761
Almost Perturbed Quadratic	13936462	194876	338797	14003318	424470	237534
LIARWHD (CUTE)	41619197	27974	180457	47476667	22254	53306
ENGVAL1 (CUTE)	8332	2285	2702	6882	6064	4442
QUARTC (CUTE)	414	494	640	402	1264	909
Generalized Quartic	1244	493	507	849	1043	798
Full Hessian FH3	1955	566	631	1352	1152	957
Diagonal 9	12984028	68189	89287	13144711	131327	125119

u odnosu na MSM metod to se i zaista desilo, što potvrđuju numerički rezultati. Međutim, ostvarilo se i drugo očekivanje, a to je povećanje broja evaluacija funkcije cilja i CPU vremena u DMSM i TMSM metoda u odnosu na MSM metod. Upotreba Algoritama 4.4.1 i 4.4.2 ima i svoju “cenu” koja se direktno izražava preko povećanja broja evaluacija funkcije cilja i CPU vremena.

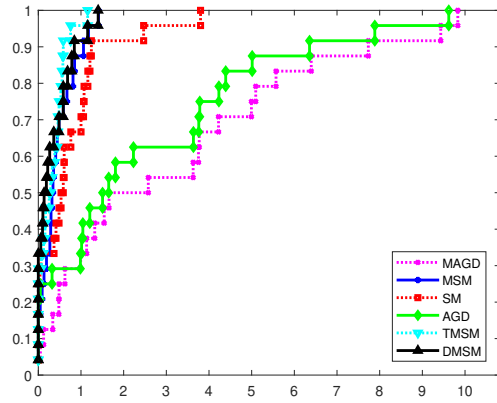
U skladu sa dobijenim numeričkim podacima u prvom i drugom numeričkom eksperimentu, možemo dati odgovor na zadato pitanje na početku ovog odeljka. Nezavisno od izbora vrednosti parametara u drugom i trećem linijskom pretraživanju unazad, DMSM metod postiže najbolje rezultate u odnosu na broj iteracija. Takođe, ako se uporede prosečni numerički rezultati dobijeni u Tabelama 4.13 i 4.17, može se uočiti da postoji blagi procen- tualni pad prosečnih numeričkih rezultata DMSM metode za broj evaluacija funkcije i CPU vreme u poređenju sa MSM metodom u drugom numeričkom eksperimentu.

Tabela 4.16: Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za CPU vreme u sekundama.

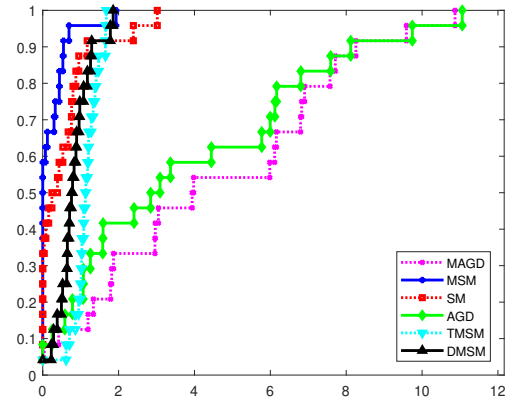
Test funkcija	CPU vreme					
	<i>MAGD</i>	<i>MSM</i>	<i>SM</i>	<i>AGD</i>	<i>TMSM</i>	<i>DMSM</i>
Perturbed Quadratic	6049.531	116.281	185.641	6756.047	219.328	134.781
Raydan 1	334.266	31.906	36.078	158.359	44.828	66.484
Diagonal 3	6401.969	52.609	102.875	5527.844	129.734	96.688
Generalized Tridiagonal 1	7.781	1.469	1.203	11.344	2.969	2.969
Extended Tridiagonal 1	8853.172	29.047	90.281	55.891	25.672	12.609
Extended TET	2.766	0.516	0.594	3.219	1.234	0.938
Diagonal 4	16.172	0.203	0.141	0.781	0.344	0.172
Diagonal 5	0.313	0.344	0.328	0.391	0.594	0.516
Extended Himmelblau	1.031	0.297	0.188	1.953	0.688	0.875
Perturbed quadratic diagonal	22820.172	139.625	185.266	44978.750	263.953	220.719
Quadratic QF1	6846.453	81.531	138.172	12602.563	173.953	91.047
Extended quadratic penalty QP1	1.063	1.000	0.797	1.266	2.781	1.813
Extended quadratic penalty QP2	1872.797	3.516	6.547	3558.734	8.750	5.906
Quadratic QF2	768.563	73.438	132.703	1582.766	169.266	98.141
Extended quadratic exponential EP1	0.844	0.688	0.438	0.750	1.000	0.859
Extended Tridiagonal 2	2.531	1.047	1.031	3.719	1.828	1.922
ARWHEAD (CUTE)	138.000	1.969	1.359	95.641	2.813	2.625
Almost Perturbed Quadratic	7086.563	73.047	133.516	13337.125	158.156	92.578
LIARWHD (CUTE)	15372.625	9.250	82.016	27221.516	5.250	17.406
ENGVAL1 (CUTE)	2.641	1.047	1.188	3.906	2.578	2.391
QUARTC (CUTE)	2.078	1.844	2.313	2.469	4.625	3.203
Generalized Quartic	0.500	0.281	0.188	0.797	0.422	0.500
Full Hessian FH3	1.188	0.391	0.391	1.438	1.063	0.891
Diagonal 9	6662.984	43.609	38.672	6353.172	61.984	114.703

Tabela 4.17: Prosečni numerički rezultati za 24 test funkcija testiranih na 12 nemeričkih testova.

Prosečne performanse	<i>MAGD</i>	<i>MSM</i>	<i>SM</i>	<i>AGD</i>	<i>TMSM</i>	<i>DMSM</i>
Broj iteracija	196051.21	9497.04	15136.33	235632.88	9058.46	8004.88
Broj evaluacija funkcije	6426009.58	67265.54	97642.88	8982579.21	118332.71	87521.54
CPU vreme (sec)	3468.58	27.71	47.58	5094.18	53.49	40.45

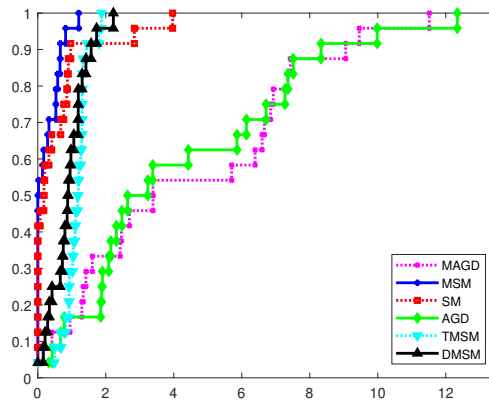


(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 4.9: Profili performansi za AGD, MAGD, MSM, SM, DMSM i TMSM metod.



Slika 4.10: Profili performansi bazirani na CPU vremenu za AGD, MAGD, MSM, SM, DMSM i TMSM metod.

Poglavlje 5

Konjugovani gradijentni metodi

U poređenju sa poznatim metodama za rešavanje nelinearnog problema (2.1) bezuslovne optimizacije velikih dimenzija, kao što je Njutnov metod i mnoge njegove modifikovane varijante, koje zahtevaju matrice velikih dimenzija i ogromnu upotrebu memorije prilikom izračunavanja i skladištenja, u ovom poglavlju razmatraju se metode konjugovanog gradijenta koje predstavljaju jednu od najefikasnijih iterativnih metoda zbog malog zahteva za memorijom u praktičnom računanju.

Metode konjugovanog gradijenta predstavljaju jednu od bitnijih klasa metoda za rešavanje problema bezuslovne optimizacije i sistema nelinearnih jednačina. Konjugovano gradijentne metode su definisane iteracijama oblika (2.2) koje sadrže linijsko pretraživanje i odgo-varajući pravac pretraživanja $\mathbf{d}_k$. Za razliku od gradijentnih metoda, pravac pretraživanja $\mathbf{d}_k$ u konjugovano gradijentnim metodama sadrži parametar ažuriranja konjugovanog gradijenta β_k koji je određen korišćenjem jednog od mnogih dostupnih pravila.

Nelinearne konjugovano gradijentne metode u [40] su podeljene u tri klase: metode ranog konjugovanog gradijenta (*eng. early conjugate gradient*), metode opadajućeg konjugovanog gradijenta (*eng. descent conjugate gradient*) i metode dovoljnog opadajućeg konjugovanog gradijenta (*eng. sufficient descent conjugate gradient*). Takođe, Andrei je klasifikovao konjugovano gradijentne metode u tri klase: klasične konjugovano gradijentne metode (*eng. classical conjugate gradient methods*), skalirane konjugovano gradijentne metode (*eng. scaled conjugate gradient methods*) i hibridne i parametrizovane konjugovano gradijentne metode (*eng. hybrid and parameterized conjugate gradient methods*).

U ovom poglavlju je dat pregled konjugovano gradijentnih metoda za bezuslovnu optimizaciju, a sama klasifikacija predstavljena u ovom poglavlju deli konjugovano gradijentne metode u sledeće kategorije: osnovne metode konjugovanog gradijenta, razmatrane u odeljku 5.1, Dai-Liao klasa metoda, predstavljena u odeljku 5.2, hibridne konjugovano gradijentne metode, opisane u odeljku 5.3 i BFGS konjugovano gradijentni metodi i njihove varijante opisane u odeljcima 5.4 i 5.5.

U odeljku 5.5 dati su naučni rezultati ostvareni urađenim istraživanjima u svrsi izrade ove doktorske disertacije, odnosno predstavljen je tročlani hibridni BFGS–CG metod iz [151]. U podsekcijama ovog odeljaka data je analiza konvergencije kao i numerički rezultati

za H-BFGS-CG1 metod.

5.1 Osnovne metode konjugovanog gradijenta

Proučavanje konjugovano gradijentnih metoda započeli su Hestenes (*Hestenes*) i Stiefel (*Stiefel*) 1952. godine u [77], a sam razvoj konjugovano gradijentnih metoda za rešavanje velikih problema bezuslovne optimizacije još uvek traje. Posle svih ovih godina, i dalje postoji potreba da se pronađe efikasniji konjugovano gradijentni metod koji će rešiti problem bezuslovne optimizacije sa hiljadama promenljivih u najkraćem mogućem vremenskom intervalu, kao i sa minimalnim brojem iteracija i evaluacija funkcija.

Konjugovano gradijentne metode konstruišu niz aproksimacija $\{\mathbf{x}_k, k \geq 0\}$ počevši od početne tačke $\mathbf{x}_0$ i generišući sledeću tačku pomoću opšte iterativne šeme (2.2), gde su pravci pretraživanja $\mathbf{d}_k$ određeni pomoću

$$\mathbf{d}_k = \begin{cases} -\mathbf{g}_0, & k=0, \\ -\mathbf{g}_k + \beta_k \mathbf{d}_{k-1}, & k \geq 1. \end{cases} \quad (5.1)$$

U jednačini (5.1) β_k je realna vrednost koja je poznata kao **parametar ažuriranja konjugovanog gradijenta**. Preciznije, pravac pretraživanja $\mathbf{d}_k$ konjugovano gradijentnih metoda je definisan kao odgovarajuća linearna kombinacija pravca gradijenta opadanja i pozitivnog višestrukog pravca korišćenog u prethodno završenoj iteraciji. Iz (5.1) i (2.2) jasno sledi da su konjugovano gradijentne metode definisane pravcem gradijenta $\mathbf{g}_k$ i parametrom ažuriranja konjugovanog gradijenta β_k . Različite konjugovano gradijentne metode proizilaze iz izbora parametar ažuriranja konjugovanog gradijenta β_k . U skladu sa zajedničkim dogovorom, $\beta_k^{\mathcal{M}}$ označava parametar β_k konjugovano gradijentne metode $\mathcal{M}$. Važno je napomenuti da neki istraživači predlažu korišćenje $\beta_k^{\mathcal{M}+} = \max\{\beta_k^{\mathcal{M}}, 0\}$. Dakle, moguće je koristiti $\beta_k^{\mathcal{M}+}$ umesto $\beta_k^{\mathcal{M}}$ i generisati odgovarajući dualni metod.

Popularnost konjugovano gradijentnih metoda potvrđena je brojnim preglednim radovima i poglavljima u knjigama [60, 74, 114, 116, 119, 152]. Pored ovih osnovnih informacija o hronološkom razvoju konjugovano gradijentnih metoda, važno je napomenuti i njihovu primenu. Generalno, konjugovano gradijentne metode su važne u rešavanju velikih problema optimizacije. Iteracije konjugovano gradijentnih metoda karakterišu mala alokacija memorije i jaka svojstva lokalne i globalne konvergencije. Na osnovu ove činjenice, ove metode postaju korisne u svim oblastima gde su uključeni problemi optimizacije bilo koje vrste. Konjugovano gradijentne metode imaju široku primenu u rešavanju sistema jednačina i obnavljanju (restauraciji) slika [23, 30, 84, 100, 102, 144, 174, 184, 185, 191], kao i u regresionoj analizi [145, 195]. Takođe, konjugovano gradijentne metode utiču na razvoj algoritama učenja veštačkih neuronskih mreža [63, 101]. Primena konjugovano gradijentnih metoda u rešavanju veoma velikih simetričnih pozitivnih poluodređenih linearnih sistema koji se pojavljuju u optimalnim površinskim parametrizacijama opisana je u [88]. Takođe, moguće je konjugovano gradijentne metode primeniti u analizi podataka [147]. Varijanta projekto-

vanog preduslova konjugovano gradijentne metode i njena primena u rešavanju problema linearnog odziva sopstvenih vrednosti je istraživana u [99].

Konjugovano gradijentne metode date u Tabelu 5.1 poznate su kao **rane** ili **klasične konjugovano gradijentne metode**, gde su $\mathbf{G}_{k-1}$, $\mathbf{y}_{k-1}$ i $\mathbf{s}_{k-1}$ definisni na osnovu (1.4) i (1.5), a oznaka $\|\cdot\|$ predstavlja Euklidsku vektorsku normu.

Tabela 5.1: Klasične konjugovano gradijentne metode.

β_k	Naziv	Godina	Referenca
$\beta_k^{\text{HS}} = \frac{\mathbf{y}_{k-1}^T \mathbf{g}_k}{\mathbf{y}_{k-1}^T \mathbf{d}_{k-1}}$	Hestenses–Stiefel	1952	[77]
$\beta_k^{\text{FR}} = \frac{\ \mathbf{g}_k\ ^2}{\ \mathbf{g}_{k-1}\ ^2}$	Fletcher–Reeves	1964	[66]
$\beta_k^{\text{PRP}} = \frac{\mathbf{y}_{k-1}^T \mathbf{g}_k}{\ \mathbf{g}_{k-1}\ ^2}$	Polak–Ribiere–Polyak	1969	[135, 136]
$\beta_k^{\text{CD}} = -\frac{\ \mathbf{g}_k\ ^2}{\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}}$	Conjugate Descent	1987	[64]
$\beta_k^{\text{LS}} = -\frac{\mathbf{y}_{k-1}^T \mathbf{g}_k}{\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}}$	Liu–Storey	1991	[105]
$\beta_k^{\text{DY}} = \frac{\ \mathbf{g}_k\ ^2}{\mathbf{y}_{k-1}^T \mathbf{d}_{k-1}}$	Dai–Yuan	1999	[46]

U navedenim konjugovano gradijentnim metodama u Tabeli 5.1 može se uočiti da brojilac parametra ažuriranja konjugovanog gradijenta β_k je $\|\mathbf{g}_k\|^2$ ili $\mathbf{y}_{k-1}^T \mathbf{g}_k$, a imenilac je $\|\mathbf{g}_{k-1}\|^2$ ili $\mathbf{y}_{k-1}^T \mathbf{d}_{k-1}$ ili $-\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}$. Dakle, dva moguća izbora za brojilac i tri moguća izbora za imenilac vode do šest različitih izbora za β_k .

Tabela 5.2: Klasifikacija konjugovano gradijentnih metoda.

Brojilac	Imenilac		
	$\ \mathbf{g}_{k-1}\ ^2$	$\mathbf{y}_{k-1}^T \mathbf{d}_{k-1}$	$-\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}$
$\ \mathbf{g}_k\ ^2$	FR	DY	CD
$\mathbf{y}_{k-1}^T \mathbf{g}_k$	PRP	HS	LS

Definisati sledeće funkcije:

$$\mathbf{n}_1 := \|\mathbf{g}_k\|^2, \quad \mathbf{n}_2 := \mathbf{y}_{k-1}^T \mathbf{g}_k, \quad \mathbf{d}_1 := \|\mathbf{g}_{k-1}\|^2, \quad \mathbf{d}_2 := \mathbf{y}_{k-1}^T \mathbf{d}_{k-1} \quad \text{i} \quad \mathbf{d}_3 := -\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}.$$

Onda parametri ažuriranja β_k za konjugovano gradijentne metode iz Tabele 5.1 su oblika:

$$\beta_k^{\text{FR}} = \frac{\mathbf{n}_1}{\mathbf{d}_1}, \quad \beta_k^{\text{PRP}} = \frac{\mathbf{n}_2}{\mathbf{d}_1}, \quad \beta_k^{\text{DY}} = \frac{\mathbf{n}_1}{\mathbf{d}_2}, \quad \beta_k^{\text{HS}} = \frac{\mathbf{n}_2}{\mathbf{d}_2}, \quad \beta_k^{\text{CD}} = \frac{\mathbf{n}_1}{\mathbf{d}_3} \quad \text{i} \quad \beta_k^{\text{LS}} = \frac{\mathbf{n}_2}{\mathbf{d}_3}.$$

Međutim, postoje izuzeci od ovih pravila. Jedan drugačiji način određivanja parametra

ažuriranja konjugovanog gradijenta β_k dat je 1967. godine u [53] i sledećeg je oblika

$$\beta_k^D = \frac{\mathbf{g}_k^T \mathbf{G}_{k-1} \mathbf{d}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{G}_{k-1} \mathbf{d}_{k-1}}.$$

Iz prikazanog hronološkog razvoja određivanja parametra ažuriranja konjugovanog gradijenta, može se videti da izbor β_k^D parametra ažuriranja konjugovanog gradijenta se strukturno razlikuje od ostalih izbora.

Za veliki problem bezuslovne nelinearne optimizacije, u praksi, izbori za ažuriranje konjugovano gradijentnih parametra koji ne zahtevaju izračunavanje ili aproksimaciju Hesijana i njegovog inverza su poželjniji u odnosu na metode koje zahtevaju izračunavanje Hesijana ili njegovu aproksimaciju u svakoj iteraciji.

U nastavku ovog odeljka navešćemo neke od osnovnih konjugovano gradijentnih metoda koje su uglavnom nastale modifikacijom klasičnih konjugovano gradijentnih metoda. Wei i ostali u [170] su dali varijantu PRP metode koju nazivamo WYL metodom, sa parametrom β_k oblika

$$\beta_k^{\text{WYL}} = \frac{\mathbf{g}_k^T \left(\mathbf{g}_k - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_{k-1} \right)}{\|\mathbf{g}_{k-1}\|^2} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_k^T \mathbf{g}_{k-1}}{\|\mathbf{g}_{k-1}\|^2}.$$

Izmene koje su uvedene u WYL metodi iskorišćene su od strane Yao i ostalih u [180] za dobijanje novih pravila za izračunavanje parametra β_k , odnosno brojioci u HS i LS metoda su zamenjeni brojiocem iz WYL metode i na taj način su dobijene dve modifikovane formule za β_k oblika:

$$\beta_k^{\text{MHS}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_k^T \mathbf{g}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}$$

i

$$\beta_k^{\text{MLS}} = \frac{\mathbf{g}_k^T \left(\mathbf{g}_k - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_{k-1} \right)}{-\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}}.$$

Zhang je u [187] napravio malu modifikaciju WYL metode i konstruisao NPRP metod na sledeći način

$$\beta_k^{\text{NPRP}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\|\mathbf{g}_{k-1}\|^2}.$$

Štaviše, Zhang je u [187] takođe proširio ovaj rezultat i na MHS metod i predložio NHS metod na sledeći način

$$\beta_k^{\text{NHS}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}.$$

Nedavno su Wei i ostali u [169] predložili varijaciju FR metode koju nazivamo VFR

metod. Parametar β_k u VFR metodi je oblika

$$\beta_k^{\text{VFR}} = \frac{\mu_1 \|\mathbf{g}_k\|^2}{\mu_2 |\mathbf{g}_k^T \mathbf{d}_{k-1}| + \mu_3 \|\mathbf{g}_{k-1}\|^2},$$

gde su $\mu_1 \in (0, +\infty)$, $\mu_2 \in (\mu_1 + \epsilon_1, +\infty)$, $\mu_3 \in (0, +\infty)$ i ϵ_1 proizvoljno date pozitivne konstante. Motivisani ovim modifikacijama u [44] autori su definisali modifikovani PRP metod

$$\beta_k^{\text{DPRP}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| + \|\mathbf{g}_{k-1}\|^2}, \quad \mu > 1.$$

Ako uvedemo sledeće funkcije

$$\mathfrak{N}_1 := \mathbf{g}_k^T \left(\mathbf{g}_k - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_{k-1} \right) = \mathfrak{n}_1 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_k^T \mathbf{g}_{k-1}$$

i

$$\mathfrak{N}_2 := \mathfrak{n}_1 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|,$$

onda imamo

$$\beta_k^{\text{WYL}} = \frac{\mathfrak{N}_1}{\mathfrak{d}_1}, \quad \beta_k^{\text{MHS}} = \frac{\mathfrak{N}_1}{\mathfrak{d}_2}, \quad \beta_k^{\text{MLS}} = \frac{\mathfrak{N}_1}{\mathfrak{d}_3}, \quad \beta_k^{\text{NPRP}} = \frac{\mathfrak{N}_2}{\mathfrak{d}_1} \quad \text{i} \quad \beta_k^{\text{NHS}} = \frac{\mathfrak{N}_2}{\mathfrak{d}_2}.$$

Neke posebne varijante konjugovanih gradijenata su β_k^{DHS} [44] i β_k^{DLS} [190], definisane pomoću

$$\beta_k^{\text{DHS}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| + \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \quad \mu > 1 \quad (5.2)$$

i

$$\beta_k^{\text{DLS}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| - \mathbf{d}_{k-1}^T \mathbf{g}_{k-1}}, \quad \mu > 1. \quad (5.3)$$

Ako su definisane funkcije

$$\mathfrak{D}_1(\mu) := \mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| + \mathbf{d}_{k-1}^T \mathbf{y}_{k-1} \quad \text{i} \quad \mathfrak{D}_2(\mu) := \mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| - \mathbf{d}_{k-1}^T \mathbf{g}_{k-1},$$

onda imamo

$$\beta_k^{\text{DHS}} = \frac{\mathfrak{N}_2}{\mathfrak{D}_1(\mu)} \quad \text{i} \quad \beta_k^{\text{DLS}} = \frac{\mathfrak{N}_2}{\mathfrak{D}_2(\mu)}.$$

Na osnovu uvedenih funkcija i dobijenih parametara ažuriranja konjugovanog gradijenta β_k možemo vrlo lako i jednostavno da uočimo povezanost u procesu modifikacije parametara ažuriranja konjugovanog gradijenta β_k kroz hronološki razvoj konjugovano gradijentnih metoda.

U nastavku je dat opšti algoritam za konjugovano gradijentne metode.

Algoritam 5.1.1 Opšti algoritam konjugovano gradijentnih metoda.

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0$, i realne konstante $0 < \epsilon < 1$, $0 < \delta < 1$.

1: Postaviti $k=0$ i izračunati $\mathbf{d}_0 = -\mathbf{g}_0$.

2: Ako je

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq \delta,$$

onda STOP; inače preći na Korak 3.

3: Izračunati $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.

4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ i odrediti $\mathbf{g}_{k+1}$, $\mathbf{y}_k$ i $\mathbf{s}_k$.

5: Izračunati β_{k+1} parametar ažuriranja konjugovano gradijentnih metoda.

6: Izračunati pravac pretraživanja $\mathbf{d}_{k+1}$.

7: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_{k+1}$ i $f(\mathbf{x}_{k+1})$.

5.2 Dai–Liao konjugovani gradijentni metod i njegove varijante

U ovom odeljku prikazaćemo Dai–Liao konjugovani gradijentni metod i njegove varijante. Mogu se posmatrati dva glavna toka istraživanja u klasi Dai–Liao konjugovano gradijentnih metoda: algoritmi koji poboljšavaju konjugovano gradijentni Dai–Liao parametar ažuriranja β_k i algoritmi koji poboljšavaju skalarni parametar t .

Proširivanje **uslova konjugacije** (*eng. conjugacy condition*)

$$\mathbf{d}_k^T \mathbf{y}_{k-1} = 0 \tag{5.4}$$

je proučavao Perry u [125]. Perry je pokušao da inkorporira informacije drugog reda ciljne funkcije u konjugovano gradijentnim metodama da bi ih ubrzao. Konkretno, korišćenjem kvazi-Njutnove jednačine i pravca pretraživanja kvazi-Njutnovih metoda, koji su definisani sa

$$B_k \mathbf{s}_{k-1} = \mathbf{y}_{k-1} \quad \text{i} \quad B_k \mathbf{d}_k = -\mathbf{g}_k, \tag{5.5}$$

dobija se sledeća relacija

$$\mathbf{d}_k^T \mathbf{y}_{k-1} = \mathbf{d}_k^T (B_k \mathbf{s}_{k-1}) = (B_k \mathbf{d}_k)^T \mathbf{s}_{k-1} = -\mathbf{g}_k^T \mathbf{s}_{k-1}, \tag{5.6}$$

gde je B_k simetrična aproksimacija matrice Hesijana G_k . Tada je Perry shodno tome zamenio uslov konjugacije (5.4) sledećim uslovom

$$\mathbf{d}_k^T \mathbf{y}_{k-1} = -\mathbf{g}_k^T \mathbf{s}_{k-1}. \tag{5.7}$$

Štaviše, Dai i Liao u [42] su uključili nenegativan parametar t u Perry-jev uslov (5.7) i zamenili ga sledećim uslovom

$$\mathbf{d}_k^T \mathbf{y}_{k-1} = -t \mathbf{g}_k^T \mathbf{s}_{k-1}. \quad (5.8)$$

Da bismo pronašli pravac pretraživanja $\mathbf{d}_k$ u (5.1) koji zadovoljava uslov konjugacije (5.8), dovoljno je pomnožiti (5.1) sa $\mathbf{y}_{k-1}$ i primeniti (5.8), a nakon toga odrediti parametar ažuriranja β_k za konjugovano gradijentne metode

$$\beta_k^{\text{DL}} = \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} = \frac{\mathbf{g}_k^T (\mathbf{y}_{k-1} - t \mathbf{s}_{k-1})}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} = \beta_k^{\text{HS}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \quad t > 0. \quad (5.9)$$

Izraz (5.9) kojim je definisan parametar ažuriranja konjugovanog gradijenta β_k^{DL} predstavlja **Dai–Liao (DL) konjugovani gradijentni metod**. Kasnije su, motivisani Dai–Liao konjugovano gradijentnim metodom, istraživači u radovima [33, 106, 133, 175, 178, 190, 193] predložili modifikovane varijante Dai–Liao konjugovano gradijentne metode. Najpoznatiji od njih su β_k^{DHSDL} [190], β_k^{DLSDL} [190] i β_k^{MHSDL} [178] dati u sledećem obliku

$$\begin{aligned} \beta_k^{\text{DHSDL}} &= \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| + \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\ &= \beta_k^{\text{DHS}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \end{aligned} \quad (5.10)$$

$$\begin{aligned} \beta_k^{\text{DLSDL}} &= \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| - \mathbf{d}_{k-1}^T \mathbf{g}_{k-1}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\ &= \beta_k^{\text{DLS}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \end{aligned} \quad (5.11)$$

$$\begin{aligned} \beta_k^{\text{MHSDL}} &= \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\ &= \beta_k^{\text{MHS}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \end{aligned} \quad (5.12)$$

gde je $t > 0$ i $\mu > 1$.

Jasno je da β_k^{DL} sa $t > 0$ definiše klasu nelinearnih konjugovano gradijentnih metoda. Štaviše, u slučaju tačnog linijskog pretraživanja, tj. $\mathbf{g}_k^T \mathbf{s}_{k-1} = 0$, tada je $\beta_k^{\text{DL}} = \beta_k^{\text{HS}}$.

Neke dodatne konjugovano gradijentne metode iz Dai–Liao klase su β_k^{MLSDL} [33] i β_k^{ZZDL} [193], definisane na sledeći način:

$$\beta_k^{\text{MLSDL}} = \frac{\mathbf{g}_k^T \left(\mathbf{g}_k - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_{k-1} \right)}{-\mathbf{d}_{k-1}^T \mathbf{g}_{k-1}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} = \beta_k^{\text{MLS}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \quad (5.13)$$

i

$$\beta_k^{\text{ZZDL}} = \frac{\mathbf{g}_k^T \mathbf{z}_{k-1}}{\mathbf{z}_{k-1}^T \mathbf{d}_{k-1}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{z}_{k-1}}, \quad (5.14)$$

gde je $t > 0$, $\mathbf{z}_{k-1} = \mathbf{y}_{k-1} + C \|\mathbf{g}_{k-1}\|^r \mathbf{s}_{k-1}$, $C > 0$ i $r > 0$.

Da bi uočili i opisali povezanost u klasi Dai–Liao konjugovano gradijentnih metoda, potrebno je definisati funkciju

$$\mathfrak{F}(\beta_k^{\mathcal{M}}, t) := \beta_k^{\mathcal{M}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}},$$

a onda imamo da važi

$$\beta_k^{\text{DL}} = \mathfrak{F}(\beta_k^{\text{HS}}, t), \quad \beta_k^{\text{DHSDL}} = \mathfrak{F}(\beta_k^{\text{DHS}}, t), \quad \beta_k^{\text{DLSDL}} = \mathfrak{F}(\beta_k^{\text{DLS}}, t),$$

$$\beta_k^{\text{MHSDL}} = \mathfrak{F}(\beta_k^{\text{MHS}}, t) \quad \text{i} \quad \beta_k^{\text{MLSDL}} = \mathfrak{F}(\beta_k^{\text{MLS}}, t).$$

U prethodne dve decenije velika pažnja istraživača u okviru klase Dai–Liao konjugovano gradijentnih metoda bila je usmerena i na određivanju najbolje vrednosti parametra t . Sam tok istraživanja najbolje vrednosti parametra t podeljen je u dva pravca. Jedan pravac istraživanja je bio usmeren u pronalaženju najbolje fiksne vrednosti za parametar t , a drugi pravac istraživanja je bio usmeren u pronalaženju najbolje aproksimacije za parametar t u svakom iterativnom koraku.

Analizirajući rezultate iz [33, 42, 178, 190] možemo doneti zaključak da parametar t ima najčešće fiksnu vrednost 0.1 u numeričkim eksperimentima. Takođe, brojna iskustva u vezi sa fiksnom vrednošću $t = 1$ su data u [42].

Opšti zaključak, kada su u pitanju numerička iskustva za fiksne vrednosti parametra t je da različiti izbori vrednosti parametra t iniciraju potpuno različite numeričke rezultate. Upravo opšti zaključak predstavlja i osnovni razlog da se dalja istraživanja fokusiraju na vrednosti parametra t koje se određuju u svakoj iteraciji. Vrednost parametra t u proizvoljnoj k -toj iteraciji biće označena sa t_k .

Hager i Zhang u [73] definišu pravilo za određivanje vrednosti parametra t_k u svakoj iteraciji na sledeći način

$$t_k = 2 \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{y}_{k-1}^T \mathbf{s}_{k-1}}. \quad (5.15)$$

Varijanta Dai–Liao metode (5.9) u kojoj je t_k definisano pomoću (5.15) poznata je kao CG-DESCENT metod.

Dai i Kou u [41] su predložili parametar ažuriranja konjugovanog gradijenta β_k^{DK} oblika

$$\beta_k^{\text{DK}} = \frac{g_k^T y_{k-1}}{y_{k-1}^T d_{k-1}} - \left(\tau_k + \frac{\|y_{k-1}\|^2}{y_{k-1}^T s_{k-1}} - \frac{y_{k-1}^T s_{k-1}}{\|s_{k-1}\|^2} \right) \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}, \quad (5.16)$$

gde je τ_k parametar skaliranja koji proizilazi iz BFGS metode. Očigledno je da Dai i Kou (DK) metod pripada klasi Dai–Liao konjugovano gradijentnih metoda, gde je parametar t_k

definisan na sledeći način

$$t_k = \tau_k + \frac{\|y_{k-1}\|^2}{y_{k-1}^T s_{k-1}} - \frac{y_{k-1}^T s_{k-1}}{\|s_{k-1}\|^2}. \quad (5.17)$$

Rezultati dati u [41] potvrđuju da DK iteracije nadmašuju mnoge postojeće konjugovano gradijentne metode.

Kasnije Babaie-Kafaki i Ghanbari u [19] predstavili su dva pogodna izbora za parametar t u (5.9) oblika:

$$t_k = \frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2} + \frac{\|y_{k-1}\|}{\|s_{k-1}\|} \quad (5.18)$$

i

$$t_k = \frac{\|y_{k-1}\|}{\|s_{k-1}\|}, \quad (5.19)$$

pri čemu su odgovarajuće konjugovane gradijentne metode Dai–Liao tipa označili sa M1 i M2. Andrei u [9] je predložio sledeće pravilo za određivanje vrednosti parametra t u (5.9) i time definisao novu varijantu Dai–Liao metode koju je označio sa DLE:

$$t_k = \frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2}. \quad (5.20)$$

Nedavno Lotfi i Hosseini u [107] su pronašli jednu od najnovijih aproksimacije parametra t :

$$t_k = \max \left\{ t_k^*, v \frac{\|y_{k-1}\|^2}{s_{k-1}^T y_{k-1}} \right\}, \quad (5.21)$$

gde je

$$t_k^* = \frac{(1 - h_k \|g_{k-1}\|^r) s_{k-1}^T g_k + \frac{g_k^T y_{k-1}}{y_{k-1}^T s_{k-1}} h_k \|g_{k-1}\|^r \|s_{k-1}\|^2}{g_k^T s_{k-1} + \frac{g_k^T s_{k-1}}{s_{k-1}^T y_{k-1}} h_k \|g_{k-1}\|^r \|s_{k-1}\|^2}, \quad (5.22)$$

$$h_k = C + \max \left\{ -\frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2}, 0 \right\} \|g_{k-1}\|^{-r}, \quad (5.23)$$

i $v > \frac{1}{4}$, C , r su tri pozitivne konstante.

Deo naučnih rezultata ostvarenih u ovoj doktorskoj disertaciji direktno je povezan sa pronalaženjem najbolje aproksimacije za parametar t u svakom iterativnom koraku u Dai–Liao konjugovano gradijentnoj metodi. Više detalja o dobijenim naučnim rezultatima biće dato u Poglavljima 6 i 8.

5.3 Hibridni konjugovani gradijentni metodi

Klasični konjugovano gradijentni metodi dati u Tabelu 5.1 mogu se podeliti u dve grupe na osnovu mnogobrojnih analiza u preglednim radovima i poglavljima knjiga. Jednu grupu čine FR, CD i DY metode koje imaju jake osobine konvergencije, ali se nisu najbolje pokazali u

numeričkim testovima, zahvaljujući fenomenu zaglavljivanja. Drugu grupu čine HS, PRP i LS metode koje ne moraju da konvergiraju u opštem slučaju, ali se u numeričkim testovima bolje ponašaju od FR, CD i DY metode. Prirodno se dolazi do ideje stvaranja hibridnih metoda koje su nastale kombinacijom metoda jedne i druge grupe, da bi se iskoristile dobre osobine i jedne i druge grupe metoda.

Hibridni konjugovani gradijentni metodi vrše dinamičko podešavanje parametra ažuriranja konjugovanog gradijenta β_k u svakom novom iterativnom koraku. Samim tim mogu sprečiti pojavu malih iterativnih koraka koji karakterišu fenomen zaglavljivanja. Mogu se podeliti u dve klase: *mešoviti konjugovano gradijentni metodi* i *konjugovano gradijentni metodi kombinovani uvođenjem jednog ili više parametara*.

U nastavku odeljka dat je prvo pregled hibridnih metoda iz klase mešovitih konjugovano gradijentnih metoda, a zatim je dat pregled hibridnih metoda iz klase konjugovano gradijentnih metoda kombinovanih uvođenjem jednog ili više parametara.

Jedan od prvih hibridnih konjugovano gradijentnih metoda je predložen u [165] na sledeći način:

$$\beta_k = \begin{cases} \beta_k^{\text{PRP}}, & \text{ako je } 0 \leq \beta_k^{\text{PRP}} \leq \beta_k^{\text{FR}}, \\ \beta_k^{\text{FR}}, & \text{inače,} \end{cases} \quad (5.24)$$

gde kada dođe do zastoja u iteracijama, koristi se parametar PRP za ažuriranje. Hu i Storey u [78] imali su sličnu motivaciju da iskoriste β_k^{PRP} i β_k^{FR} i predložili su sledeće pravilo

$$\beta_k = \max\{0, \min\{\beta_k^{\text{PRP}}, \beta_k^{\text{FR}}\}\}. \quad (5.25)$$

U [67] je ukazano da β_k^{PRP} može imati negativnu vrednost, čak i za jako konveksne funkcije. U pokušaju da prošire dozvoljene izbore za parametar ažuriranja PRP-a, uz zadržavanje globalne konvergencije, Nocedal i Gilbert [67] su predložili izbor

$$\beta_k = \max\{-\beta_k^{\text{FR}}, \min\{\beta_k^{\text{PRP}}, \beta_k^{\text{FR}}\}\}. \quad (5.26)$$

Dai i Yuan u [47] kombinuju DY metod sa HS konjugovano gradijentnim metodom, što dovodi do sledećih β_k parametara ažuriranja konjugovano gradijentnih metoda:

$$\beta_k = \max\{0, \min\{\beta_k^{\text{HS}}, \beta_k^{\text{DY}}\}\} \quad (5.27)$$

i

$$\beta_k = \max\{-c \beta_k^{\text{DY}}, \min\{\beta_k^{\text{HS}}, \beta_k^{\text{DY}}\}\}, \quad c = \frac{1 - \sigma}{1 + \sigma} > 0. \quad (5.28)$$

U [43], testirali su različite konjugovano gradijentne metode za velike probleme bezuslovne optimizacije i zaključili da je hibridni konjugovano gradijentni metod (5.27) dao najbolje rezultate.

Dai je predložio u [39] sledeći hibridni konjugovano gradijentni metod koji koristi DY ili

CD parametar ažuriranja:

$$\beta_k = \frac{\|\mathbf{g}_k\|^2}{\max\{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}, -\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}\}}. \quad (5.29)$$

Modifikovani konjugovano gradijentni metod definisan kao hibridizacija poznatih LS i CD konjugovano gradijentnih metoda je predstavljen i analiziran u [177] u sledećem obliku

$$\beta_k^{\text{LSCD}} = \max\{0, \min\{\beta_k^{\text{LS}}, \beta_k^{\text{CD}}\}\}. \quad (5.30)$$

Konjugovano gradijentne metode se mogu kombinovati uvođenjem jednog ili više parametara. Dai i Yuan u [45, 48] su predložili jednoparametarsku familiju konjugovano gradijentnih metoda oblika

$$\beta_k = \frac{\|\mathbf{g}_k\|^2}{\theta_k \|\mathbf{g}_{k-1}\|^2 + (1 - \theta_k) \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \quad (5.31)$$

gde je parametar $\theta_k \in [0, 1]$. Pri čemu je $\beta_k = \beta_k^{\text{FR}}$ u slučaju $\theta_k = 1$, i $\beta_k = \beta_k^{\text{DY}}$ ako je $\theta_k = 0$.

Hibridni metod koji su predložili Delladji, Belloufi i Sellami u [57] koristi PRP ili HZ parametar ažuriranja i oblika je

$$\begin{aligned} \beta_k^{\text{hPRPHZ}} &= \theta_k \beta_k^{\text{PRP}} + (1 - \theta_k) \beta_k^{\text{HZ}} \\ &= \theta_k \frac{\mathbf{y}_{k-1}^T \mathbf{g}_k}{\|\mathbf{g}_{k-1}\|^2} + (1 - \theta_k) \frac{1}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \left(\mathbf{y}_{k-1} - 2 \mathbf{d}_{k-1} \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right)^T \mathbf{g}_k, \end{aligned} \quad (5.32)$$

gde je $\theta_k \in [0, 1]$ i naziva se **parametar hibridizacije**. U slučaju da je $\theta_k = 1$ onda je $\beta_k^{\text{hPRPHZ}} = \beta_k^{\text{PRP}}$, a ako je $\theta_k = 0$ onda je $\beta_k^{\text{hPRPHZ}} = \beta_k^{\text{HZ}}$.

Kombinacija dva konjugovana gradijentna metoda uz pomoć θ_k parametra hibridizacije se naziva **konveksna kombinacija**.

Nazareth u [120] je predložio dvoparametarsku familiju β_k parametra ažuriranja konjugovano gradijentnih metoda koristeći konveksne kombinacije brojilaca i imenilaca i definisao je na sledeći način

$$\beta_k = \frac{\nu_k \|\mathbf{g}_k\|^2 + (1 - \nu_k) \mathbf{g}_k^T \mathbf{y}_{k-1}}{\theta_k \|\mathbf{g}_{k-1}\|^2 + (1 - \theta_k) \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \quad (5.33)$$

gde su $\nu_k, \theta_k \in [0, 1]$. Ova dvoparametarska familija uključuje FR, DY, PRP i HS metode kao ekstremne slučajeve.

Naučni rezultati ostvareni u ovoj doktorskoj disertaciji u vezi sa hibridnim konjugovano gradijentnim metodama biće detaljno objašnjeni i prezentovani u Poglavlju 6.

5.4 BFGS konjugovani gradijentni metodi

Poznata činjenica je da su konjugovano gradijentne iteracije efikasnije od kvazi-Njutnovih metoda u pogledu CPU vremena. Štaviše, BFGS ažuriranja zahtevaju veću upotrebu memorijskog prostora nego konjugovano gradijentni metodi. S druge strane, kvazi-Njutnove metode zahtevaju manji broj iteracija kao i manji broj evaluacija ciljne funkcije. U tu svrhu, jedan od savremenih trendova u definisanju novih konjugovano gradijentnih metoda je korišćenje BFGS ažuriranja u definisanju novih pravila za definisanje β_k parametra ažuriranja konjugovanog gradijenta. Hibridni metod koji rešava sistem nelinearnih jednačina kombinujući kvazi-Njutnov metod sa optimizacijom haosa (*eng. chaos optimization*) predložen je u [108]. U [76], autori su definisali kombinaciju kvazi-Njutnovog i Košijevog metoda najstrmijeg pada (*eng. steepest descent*) za rešavanje problema bezuslovne optimizacije, koji je poznat kao kvazi-Njutnov metod najstrmijeg pada.

Ibrahim i ostali u [81] su predložili hibridni pravac pretraživanja koji kombinuje kvazi-Njutnove i konjugovano gradijentne metode. Dakle, pravac pretraživanja hibridnog metoda, poznat kao BFGS-CG metod u [81] je definisan na sledeći način

$$\mathbf{d}_k = \begin{cases} -B_k \mathbf{g}_k, & k=0, \\ -B_k \mathbf{g}_k + \eta(-\mathbf{g}_k + \beta_k \mathbf{d}_{k-1}), & k \geq 1, \end{cases} \quad (5.34)$$

gde je $\eta > 0$ i $\beta_k = \frac{\mathbf{g}_k^T \mathbf{g}_{k-1}}{\mathbf{g}_k^T \mathbf{d}_{k-1}}$. U poređenju sa standardnim BFGS metodama i metodama konjugovanog gradijenta, BFGS-CG metod pokazuje značajno poboljšanje u ukupnom broju iteracija i CPU vremenu potrebnom za rešavanje velikih problema bezuslovne optimizacije. Takođe, dokazano je da BFGS-CG metod globalno konvergentan.

Hibridni pravac pretraživanja između BFGS ažuriranja Hesijanove matrice i parametra ažuriranja konjugovanog gradijenta β_k je predložen i ispitan u [20, 91]. Hibridni DFP-CG metod za rešavanje problema bezuslovne optimizacije predstavljen je u [123].

5.5 Tročlani H-BFGS-CG1 metod

Hibridni konjugovano gradijentni algoritmi istraživani su u [151] i zasnovani su na modifikovanom pravcu pretraživanja koji je definisan korišćenjem jedne od sledeće dve hibridizacije:

$$\mathbf{d}_k = - \left(1 + \beta_k \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2} \right) \mathbf{g}_k + \beta_k \mathbf{d}_{k-1}, \quad (5.35)$$

$$\mathbf{d}_k = -B_k \mathbf{g}_k - \left(1 + \beta_k \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2} \right) \mathbf{g}_k + \beta_k \mathbf{d}_{k-1}, \quad (5.36)$$

gde je parametar ažuriranja konjugovanih gradijenata β_k definisan korišćenjem pogodnih kombinacija parametara β_k uključenih u Tabeli 5.1 i prethodno definisanih hibridizacija.

Matrica B_k u (5.36) je definisana kao odgovarajuća Hesijanova aproksimacija BFGS ažuriranja.

Tročlani hibridni BFGS-CG metod, poznat kao H-BFGS-CG1, je predložen od strane Stanimirovića i ostalih u [151] sa pravcem pretraživanja

$$\mathbf{d}_k := \begin{cases} -B_k \mathbf{g}_k, & k=0, \\ -B_k \mathbf{g}_k - \left(1 + \beta_k^{LSCD} \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}\right) \mathbf{g}_k + \beta_k^{LSCD} \mathbf{d}_{k-1}, & k \geq 1. \end{cases} \quad (5.37)$$

Algoritam 5.5.1 definiše odgovarajuću proceduru za izračunavanje H-BFGS-CG1 metode.

Algoritam 5.5.1 Tročlani H-BFGS-CG1 algoritam.

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0$ i realni parametri $0 < \epsilon < 1$ i $0 < \delta < 1$.

1: Postaviti $k=0$ i izračunati $\mathbf{g}_0$, $B_0 = I$, $\mathbf{d}_0 = -B_0 \mathbf{g}_0$.

2: Ako je

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq \delta,$$

onda STOP; inače preći na Korak 3.

3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.

4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ i odrediti $\mathbf{g}_{k+1}$, $\mathbf{y}_k$ i $\mathbf{s}_k$.

5: Izračunati

$$\beta_{k+1}^{LS} = \frac{\mathbf{y}_k^T \mathbf{g}_{k+1}}{-\mathbf{g}_k^T \mathbf{d}_k}, \quad \beta_{k+1}^{CD} = \frac{\|\mathbf{g}_{k+1}\|^2}{-\mathbf{g}_k^T \mathbf{d}_k},$$

$$\beta_{k+1}^{LSCD} = \max\{0, \min\{\beta_{k+1}^{LS}, \beta_{k+1}^{CD}\}\}.$$

6: Izračunati B_{k+1} koristeći (3.19).

7: Odrediti pravac pretraživanja $\mathbf{d}_{k+1}$ pomoću (5.37).

8: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

5.5.1 Analiza konvergencije za H-BFGS-CG1 metod

Da bi smo dokazali globalnu konvergenciju za H-BFGS-CG1 metod potrebne su sledeće pretpostavke:

Pretpostavka 5.5.1. (H1) : Funkcija cilja f je dva puta neprekidno diferencijabilna.

(H2) : Skup S je konveksan. Štaviše, postoje pozitivne konstante c_1 i c_2 takve da je

$$c_1 \|\mathbf{z}\|^2 \leq \mathbf{z}^T F(\mathbf{x}) \mathbf{z} \leq c_2 \|\mathbf{z}\|^2,$$

za svako $\mathbf{z} \in \mathbb{R}^n$ i $\mathbf{x} \in S$, gde je $F(\mathbf{x})$ Hesijanova matrica funkcije f .

(H3) : Gradijent $\mathbf{g}$ je Lipšic neprekidan u tački $\mathbf{x}^*$, tj. postoji pozitivna konstanta c_3 koja

zadovoljava nejednačinu

$$\|\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{x}^*)\| \leq c_3 \|\mathbf{x} - \mathbf{x}^*\|, \text{ za svako } \mathbf{x} \text{ u okolini } \mathbf{x}^*.$$

Teorema 5.5.1. [29] *Neka je $\{B_k\}$ generisan BFGS ažuriranjem (3.19). Pretpostavimo da je matrica B_k simetrična, pozitivno definitna i da zadovoljava jednačinu $B_{k+1}\mathbf{s}_k = \mathbf{y}_k$ i $\mathbf{s}_k^T \mathbf{y}_k > 0$ za svako k . Pored toga, pretpostavimo da $\{\mathbf{s}_k\}$ i $\{\mathbf{y}_k\}$ zadovoljavaju nejednakost*

$$\frac{\|\mathbf{y}_k - G_* \mathbf{s}_k\|}{\|\mathbf{s}_k\|} \leq \epsilon_k,$$

za neku simetričnu i pozitivnu matricu G_* i za neki niz $\{\epsilon_k\}$ koji ima svojstvo

$$\sum_{k=1}^{\infty} \epsilon_k < \infty.$$

Onda

$$\lim_{k \rightarrow \infty} \frac{\|(B_k - G_*)\mathbf{s}_k\|}{\|\mathbf{s}_k\|} = 0$$

i nizovi $\{\|B_k\|\}$, $\{\|B_k^{-1}\|\}$ su ograničeni.

Teorema 5.5.2. *(Dovoljan uslov opadanja i globalna konvergencija.) Razmotrite Algoritam 5.5.1. Pretpostavimo da su uslovi H1, H2 i H3 u Pretpostavci 5.5.1 zadovoljeni kao i uslovi Teoreme 5.5.1. Onda*

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k\|^2 = 0.$$

Dokaz. Množenjem $\mathbf{d}_k$ u (5.37) sa $\mathbf{g}_k$ sa leve strane dobija se

$$\begin{aligned} \mathbf{g}_k^T \mathbf{d}_k &= -\mathbf{g}_k^T B_k \mathbf{g}_k - \mathbf{g}_k^T \mathbf{g}_k - \beta_k^{LSCD} \mathbf{g}_k^T \mathbf{d}_{k-1} + \beta_k^{LSCD} \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &\leq -c_1 \|\mathbf{g}_k\|^2 - \|\mathbf{g}_k\|^2 \\ &= -(c_1 + 1) \|\mathbf{g}_k\|^2 \leq -\|\mathbf{g}_k\|^2, \quad c_1 + 1 > 0, \end{aligned} \tag{5.38}$$

odakle se zaključuje da važi dovoljan uslov opadanja.

Dalje, iz uslova linijskog pretraživanja unazad i (5.38) važi

$$f(\mathbf{x}_k) - f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \leq -\sigma \alpha_k \mathbf{g}_k^T \mathbf{d}_k \leq \sigma \alpha_k \|\mathbf{g}_k\|^2. \tag{5.39}$$

Kako se vrednost funkcija $f(\mathbf{x}_k)$ smanjuje i niz $f(\mathbf{x}_k)$ je ograničen odozdo na osnovu H2, sledi da je

$$\lim_{k \rightarrow \infty} (f(\mathbf{x}_k) - f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)) = 0. \tag{5.40}$$

Dakle iz (5.39) i (5.40) sledi da

$$\lim_{k \rightarrow \infty} \sigma \alpha_k \|\mathbf{g}_k\|^2 = 0.$$

Sada na osnovu $\alpha_k > 0$ i $\sigma > 0$, imamo da je

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k\|^2 = 0,$$

što je i bio cilj da se dokaže. $\square$

5.5.2 Numerički rezultati za H-BFGS-CG1 metod

Da bismo pokazali i dokazali efikasnost tročlanog H-BFGS-CG1 metoda potrebno je uraditi numerička testiranja i uporeti ih sa numeričkim rezultatima poznatih metoda za безусловnu optimizaciju. Tročlani H-BFGS-CG1 metod upoređujemo sa DHSDL i DLSDL metodama iz [190]. Kodovi za sve navedene metode su napisani u Matlab R2017a programskom jeziku i izvršeni na laptop računaru sa Intel Core i3 2.0 GHz CPU-om, 8 GB RAM-a i Windows 10 operativnim sistemom. U numeričkim eksperimentima analizirana su tri indikatora (svojstva) za svaki testirani metod: broj iterativnih koraka (broj iteracija), ukupan broj izračunavanja vrednosti funkcije u različitim tačkama (broj evaluacija funkcije) i vreme izvršavanja algoritma (CPU vreme).

Numerička testiranja su bazirana na 26 test funkcijama iz [7], gde su mnoge test funkcije preuzete iz CUTER kolekcije [26]. Za svaku test funkciju urađeno je 10 numeričkih testova sa različitim izborom broja promenljivih: 100, 200, 300, 500, 700, 800, 1000, 1500, 2000 i 3000.

Kriterijumi zaustavljanja algoritama su isti za sve testirane metode koje se koriste u ovom numeričkom eksperimentu:

$$\|\mathbf{g}_k\| \leq 10^{-6} \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq 10^{-16}.$$

Za ulazne parametre u Algoritmu 2.2.3 linijskog pretraživanja unazad korišćene su vrednosti $\sigma = 0.0001$ i $\beta = 0.8$. Zbirni numerički rezultati za sva tri metoda, testiranih na 26 test funkcija predstavljeni su u Tabelama 5.3, 5.4 i 5.5.

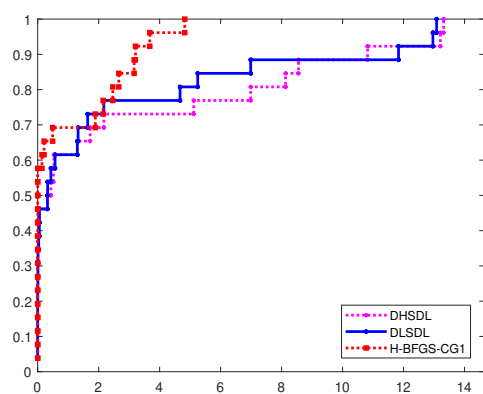
Slike 5.1 i 5.2 prikazuju profile performansi posmatranih metoda u odnosu na broj iteracija, broj evaluacija funkcija i CPU vreme. Dati profili performansi iz [58] predstavljaju jedan od najčešće korišćenih alata za upoređivanje performansi metoda za безусловnu optimizaciju na velikom skupu testova. Gornja kriva odgovara metodi koja pokazuje najbolje performanse u odnosu na izabrani profil performansi.

Na Slici 5.1a vidimo da sva tri metoda uspešno rešavaju sve test probleme, a H-BFGS-CG1 metod je najbolji u 57.69% testiranih problema u poređenju sa DHSDL (34.62%) i DLSDL (15.38%). Takođe, na Slici 5.1b uočavamo da sva tri metoda (DHSDL, DLSDL i H-BFGS-CG1) uspešno rešavaju sve test probleme, dok je H-BFGS-CG1 metod najbolji u 38.46% testiranih problema u poređenju sa DHSDL (50%) i DLSDL (26.92%).

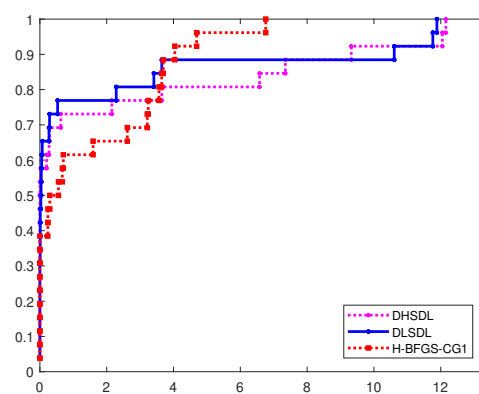
Na Slici 5.2 dati su profili performansi bazirani na CPU vremenu i možemo uočiti da sva tri metoda uspešno rešavaju sve test probleme, pri čemu je H-BFGS-CG1 metod najbolji u

Tabela 5.3: Zbirni numerički rezultati za broj iterativnih koraka.

Test funkcija	Broj iteracija		
	$DHSDL$	$DLSDL$	$H\text{-}BFGS\text{-}CG1$
Extended Penalty	1456	1492	13055
Raydan 1	23050	23217	32385
Raydan 2	390	392	451
Diagonal 2	4758	5209	137
Diagonal 3	22477	19693	15706
Hager	1692	1764	7496
Generalized Tridiagonal 1	623	606	7750
Extended Tridiagonal 1	596328	827	330
Extended TET	3411	3500	2357
Diagonal 5	25070	25070	25043
Extended Himmelblau	1306	1306	7201
Extended quadratic penalty QP1	563	585	2087
Extended quadratic exponential EP1	504	504	3187
Extended Tridiagonal 2	1416	1417	40094
ENGVAL1 (CUTE)	607	594	5486
NONSCOMP (CUTE)	3145116	30559018	8366
Diagonal 6	466	470	514
DIXON3DQ (CUTE)	23873421	20357499	2344
COSINE (CUTE)	31671	2875	113
BIGGSB1 (CUTE)	21483005	18112456	2270
Generalized Quartic	1245	1178	378
Diagonal 7	5454	5454	43
Diagonal 8	17891	17891	13254
Full Hessian FH3	3702	3699	1503
HIMMELH (CUTE)	90	90	20
Extended Rosenbrock	50	50	40



(a) Bazirani na broju iteracija.

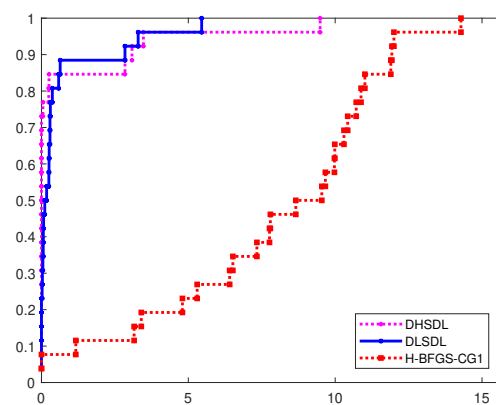


(b) Bazirani na broju evaluacija funkcije.

Slika 5.1: Profili performansi za $DHSDL$, $DLSDL$ i $H\text{-}BFGS\text{-}CG1$ metod.

Tabela 5.4: Zbirni numerički rezultati za broj evaluacija testiranih funkcija.

Test funkcija	Broj evaluacija funkcije		
	<i>DHSDL</i>	<i>DLSDL</i>	<i>H-BFGS-CG1</i>
Extended Penalty	39354	40173	508837
Raydan 1	462262	465590	1394768
Raydan 2	970	975	1192
Diagonal 2	9526	10428	2139
Diagonal 3	679418	589965	939249
Hager	23955	25156	222414
Generalized Tridiagonal 1	10239	10064	260172
Extended Tridiagonal 1	3140661	4912	5800
Extended TET	34150	35020	50201
Diagonal 5	50310	50310	50298
Extended Himmelblau	24774	24774	293347
Extended quadratic penalty QP1	9291	9644	57012
Extended quadratic exponential EP1	13734	13734	171947
Extended Tridiagonal 2	11253	11336	1220619
ENGVAL1 (CUTE)	8520	8769	140092
NONSCOMP (CUTE)	55575511	531637119	340659
Diagonal 6	1164	1171	1372
DIXON3DQ (CUTE)	194329593	161402916	42659
COSINE (CUTE)	950303	105989	9956
BIGGSB1 (CUTE)	174871424	143604806	41301
Generalized Quartic	11696	10944	7574
Diagonal 7	27714	27714	2209
Diagonal 8	89819	89819	146099
Full Hessian FH3	120485	120291	99571
HIMMELH (CUTE)	190	190	1800
Extended Rosenbrock	110	110	90



Slika 5.2: Profili performansi bazirani na CPU vremenu za DHSDL, DLSDL i H-BFGS-CG1 metod.

Tabela 5.5: Zbirni numerički rezultati za CPU vreme u sekundama.

Test funkcija	CPU vreme (sec)		
	<i>DHSDL</i>	<i>DLSDL</i>	<i>H-BFGS-CG1</i>
Extended Penalty	1.891	1.922	7745.109
Raydan 1	17.016	17.906	28628.781
Raydan 2	0.141	0.219	177.156
Diagonal 2	1.688	2.063	66.547
Diagonal 3	99.109	82.953	7598.953
Hager	5.078	5.297	5111.313
Generalized Tridiagonal 1	1.203	1.359	2484.188
Extended Tridiagonal 1	1099.453	1.531	130.000
Extended TET	2.313	2.5	935.063
Diagonal 5	9.656	10.094	9798.609
Extended Himmelblau	0.719	0.719	2737.547
Extended quadratic penalty QP1	0.594	0.5	688.172
Extended quadratic exponential EP1	0.641	0.766	1207.734
Extended Tridiagonal 2	0.641	0.828	12771.406
ENGVAL1 (CUTE)	0.469	0.578	1830.703
NONSCOMP (CUTE)	732.359	32109.703	1647.484
Diagonal 6	0.281	0.344	210.125
DIXON3DQ (CUTE)	10948.516	9275.313	1292.516
COSINE (CUTE)	42.063	5.859	61.828
BIGGSB1 (CUTE)	9884.078	8730.078	891.500
Generalized Quartic	0.719	1.078	159.453
Diagonal 7	2.297	2.375	20.453
Diagonal 8	6.375	6.156	5002.109
Full Hessian FH3	6.922	7.078	1117.875
HIMMELH (CUTE)	0.328	0.391	9.141
Extended Rosenbrock	0.078	0.094	16.984

7.69% testiranih problema u poređenju sa DHSDL (73.08%) i DLSDL (23.08%).

Tabela 5.6: Prosečni numerički rezultati za 26 test funkcija testiranih na 10 numeričkih testova.

Prosečne performanse	<i>DHSDL</i>	<i>DLSDL</i>	<i>H-BFGS-CG1</i>
Broj iteracija	1,894,067.77	2,659,494.46	7,369.62
Broj evaluacija funkcije	16,557,554.85	32,242,381.50	231,206.81
CPU vreme (sec)	879.41	1,933.37	3,551.57

Prosečni numerički rezultati u Tabeli 5.6 pokazuju izuzetan napredak u smanjenju broja iteracija i broja evaluacija funkcija korišćenjem H-BFGS-CG1 metode. U poređenju sa DHSDL i DLSDL metodama, H-BFGS-CG1 metod je postigao drastično manji prosečan

broj iterativnih koraka, a takođe i manji prosečan broj evaluacija funkcija. Međutim, ako posmatramo prosečna CPU vremena za sva tri metoda, možemo zaključiti da je H-BFGS-CG1 metod prilično spor. Isti zaključak sledi i na osnovu profila performansi prikazanih na Slici 5.2. Prema tome, može se zaključiti da H-BFGS-CG1 metod daje dobre prosečne rezultate.

Iz svega navedenog možemo dati konačan zaključak da H-BFGS-CG1 metod daje određena poboljšanja u pogledu efikasnosti kada je u pitanju broj iteracija i broj evaluacija funkcija, dok kada je u pitanju CPU vreme, potrebno je dalje istraživati.

Poglavlje 6

Napredni konjugovani gradijentni metodi

U ovom poglavlju predstavljeni su algoritmi i naučni rezultati iz tri objavljena naučna rada [83, 87, 151], koji predstavljaju glavni deo disertacije. U radu [151] su prezentovana dva hibridna konjugovano gradijentna metoda, od kojih je jedan nastao kombinacijom klasičnih konjugovano gradijentnih metoda, a drugi kombinacijom modifikovanih Dai–Liao konjugovano gradijentnih metoda sa upotrebom odgovarajućih pravaca pretraživanja. Dok u radovima [83, 87] su prezentovane dve nove varijante Dai–Liao konjugovano gradijentnih metoda. U [87] je dato novo pravilo za određivanje parametra t u Dai–Liao konjugovano gradijentnoj metodi. Definisanjem novog načina određivanja vrednosti parametra t došlo se do nove varijante Dai–Liao konjugovano gradijentnog metoda. Za ovaj metod dat je Algoritam, dokazana globalna konvergencija i numerička efikasnost.

U radu [83] urađeno je izjednačavanje pravca pretraživanja iz metode ubrzanog opadajućeg gradijenta sa pravcem pretraživanja u konjugovano gradijentnim metodama Dai–Liao tipa sa ciljem određivanja novog pravila za izračunavanje vrednosti parametra t . Definisanje novog pravila za izračunavanje vrednosti parametra t uticalo je na nastanak nove varijante Dai–Liao konjugovano gradijentnog metoda.

6.1 Mešoviti LS-CD konjugovano gradijentni metod

Hibridni konjugovani gradijentni metodi nastaju kombinacijom različitih konjugovano gradijentnih metoda sa ciljem poboljšanja efikasnosti baznih metoda i sprečavanju pojave malih iterativnih koraka koji karakterišu fenomen zaglavljivanja.

Stanimirović i ostali u radu [151] su predložili hibridni konjugovano gradijentni metod, gde je pravac pretrage $\mathbf{d}_k$, $k \geq 1$ definisan u (5.1), poboljšan korišćenjem sledećeg pravila

$$\mathbf{d}_k = - \left(1 + \beta_k \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2} \right) \mathbf{g}_k + \beta_k \mathbf{d}_{k-1}, \quad (6.1)$$

a β_k se određuje korišćenjem odgovarajućih kombinacija β_k datih u Tabeli 5.1 i/ili prethodno definisanim hibridizacijama.

Koristeći jake konvergencione osobine CD metode i u isto vreme dobro ponašanje u numeričkim testovima LS metode definisan je LSCD metod u [177] na sledeći način

$$\begin{aligned}\beta_k^{\text{LSCD}} &= \max\{0, \min\{\beta_k^{\text{LS}}, \beta_k^{\text{CD}}\}\}, \\ \mathbf{d}_k &= \begin{cases} -\mathbf{g}_0, & k=0, \\ -\mathbf{g}_k + \beta_k^{\text{LSCD}} \mathbf{d}_{k-1}, & k \geq 1. \end{cases}\end{aligned}\quad (6.2)$$

U [151], autori su definisali modifikaciju LSCD metode. Dobijeni metod je poznat kao **mešoviti LS-CD konjugovano gradijentni (MLSCD) metod** sa sledećim pravcem pretraživanja

$$\mathbf{d}_k = \begin{cases} -\mathbf{g}_0, & k=0, \\ -\left(1 + \beta_k^{\text{LSCD}} \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}\right) \mathbf{g}_k + \beta_k^{\text{LSCD}} \mathbf{d}_{k-1}, & k \geq 1. \end{cases}\quad (6.3)$$

Generalno, ideja se bazira na zameni pravca pretraživanja $\mathbf{d}_k$ iz [177] sa pravcem pretraživanja $\mathbf{d}_k$ definisanim u (6.3).

U nastavku je dat Algoritam 6.1.1 za MLSCD metod.

Algoritam 6.1.1 Mešoviti LS-CD konjugovano gradijentni (MLSCD) metod.

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0$ i realni parametri $0 < \epsilon < 1$, $0 < \delta < 1$.

- 1: Postaviti $k=0$ i izračunati $\mathbf{d}_0 = -\mathbf{g}_0$.
- 2: Ako je

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq \delta,$$

onda STOP; inače preći na Korak 3.

- 3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.
- 4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ i odrediti $\mathbf{g}_{k+1}$ i $\mathbf{y}_k$.
- 5: Izračunati

$$\begin{aligned}\beta_{k+1}^{\text{LS}} &= \frac{\mathbf{y}_k^T \mathbf{g}_{k+1}}{-\mathbf{g}_k^T \mathbf{d}_k}, \quad \beta_{k+1}^{\text{CD}} = \frac{\|\mathbf{g}_{k+1}\|^2}{-\mathbf{g}_k^T \mathbf{d}_k}, \\ \beta_{k+1}^{\text{LSCD}} &= \max\{0, \min\{\beta_{k+1}^{\text{LS}}, \beta_{k+1}^{\text{CD}}\}\}.\end{aligned}$$

- 6: Odrediti pravac pretraživanja $\mathbf{d}_{k+1}$ koristeći (6.3).
- 7: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

6.2 Mešoviti DHSDL–DLSDL konjugovano gradijentni metod

U ovom odeljku, predstavimo hibridni konjugovano gradijentni metod baziran na konjugovano gradijentnim metodama Dai–Liao tipa, a koji su dali Stanimirović i ostali u radu [151]. Hibridni metod Dai–Liao tipa je nastao na osnovu ideja iz predhodnog odeljka u vezi MLSCD metode. Za definisanje parametra ažuriranja hibridnog konjugovanog gradijentnog metoda izkorišćene su DHSDL i DLSDL metode iz [190].

Mešoviti DHSDL–DLSDL konjugovano gradijentni metod, odnosno MMDL metod kako je nazvan u [151], je definisan sledećim parametrom ažuriranja konjugovanog gradijenta

$$\beta_k^{MMDL} = \max\{0, \min\{\beta_k^{DHSDL}, \beta_k^{DLSDL}\}\}, \quad (6.4)$$

gde je

$$\beta_k^{DHSDL} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| + \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - \alpha_{k-1} \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \quad (6.5)$$

i

$$\beta_k^{DLSDL} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mu |\mathbf{g}_k^T \mathbf{d}_{k-1}| - \mathbf{d}_{k-1}^T \mathbf{g}_{k-1}} - \alpha_{k-1} \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \quad (6.6)$$

dok je pravcem pretraživanja $\mathbf{d}_k$ određen na sledeći način

$$\mathbf{d}_k = \begin{cases} -\mathbf{g}_0, & k=0, \\ -\left(1 + \beta_k^{MMDL} \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}\right) \mathbf{g}_k + \beta_k^{MMDL} \mathbf{d}_{k-1}, & k \geq 1. \end{cases} \quad (6.7)$$

Međutim, pravila za određivanje parametara ažuriranja konjugovanog gradijenta β_k^{DHSDL} i β_k^{DLSDL} definisana u (6.5) i (6.6) se razlikuju od originalnih pravila datih u (5.10) i (5.11). Razlika je u tome što se umesto fiksne vrenosti pozitivnog realnog parametra $t > 0$ u (5.10) i (5.11) koristi promenljiva vrednost parametra α_k iz linijskog pretrživanja unazad. Ovaj parametar u svakoj iteraciji menja vrednost i samim tim različito utiče na vrednosti parametara β_k^{DHSDL} i β_k^{DLSDL} sa jedne strane, a sa druge strane ispunjava uslov da je $\alpha_k > 0$. Ispunjenost uslova $\alpha_k > 0$ direkno sledi iz Algoritam 2.2.3.

Algoritam za MMDL metod je predstavljen pomoću Algoritma 6.2.1.

6.3 Konjugovano gradijentni metod Dai–Liao tipa sa poboljšanim parametrima

Predstavljena su nova pravila za izračunavanje parametra t uključenog u svakoj iteraciji Dai–Liao konjugovano gradijentnih metoda. Nova pravila za određivanje vrednosti parametra pokreću efikasnije i robusnije varijante Dai–Liao algoritma. Pod odgovarajućim uslovima,

Algoritam 6.2.1 Mešoviti DHSDL–DLSDL konjugovano gradijentni (MMDL) metod.

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0$ i realni parametri $0 < \epsilon < 1$, $0 < \delta < 1$ i $\mu > 1$.

1: Postaviti $k=0$ i izračunati $\mathbf{d}_0 = -\mathbf{g}_0$.

2: Ako je

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq \delta,$$

onda STOP, inače preći na Korak 3.

3: Pronaći dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.

4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ i odrediti $\mathbf{g}_{k+1}$, $\mathbf{y}_k$ i $\mathbf{s}_k$.

5: Izračunati

$$\begin{aligned} \beta_{k+1}^{DHSDL} &= \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| + \mathbf{d}_k^T \mathbf{y}_k} - \alpha_k \frac{\mathbf{g}_{k+1}^T \mathbf{s}_k}{\mathbf{d}_k^T \mathbf{y}_k}, \\ \beta_{k+1}^{DLSDL} &= \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| - \mathbf{d}_k^T \mathbf{g}_k} - \alpha_k \frac{\mathbf{g}_{k+1}^T \mathbf{s}_k}{\mathbf{d}_k^T \mathbf{y}_k}, \\ \beta_{k+1}^{MMDL} &= \max\{0, \min\{\beta_{k+1}^{DHSDL}, \beta_{k+1}^{DLSDL}\}\}. \end{aligned}$$

6: Odrediti pravac pretraživanja $\mathbf{d}_{k+1}$ koristeći (6.7).

7: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

teorijska analiza pokazuje da su predloženi metodi u kombinaciji sa linijskim pretraživanjem unazad globalno konvergentni. Prikazani su i numerički eksperimenti koji potvrđuju uticaj novih vrednosti parametra t na ponašanje Dai-Liao konjugovano gradijentnih metoda.

Kako je parametar t važan za numeričko ponašanje Dai-Liao konjugovano gradijentnih metoda [8], jedan od najvažnijih problema u implementaciji klase Dai-Liao konjugovano gradijentnih metoda je kako odrediti najbolju vrednost parametra $t > 0$ koja će dati željene numeričke rezultate. Mnogi naučnici su u prethodnom periodu uložili mnogo vremena i truda da odrede najbolju definiciju nenegativnog parametra t u Dai-Liao klasi konjugovano gradijentnih metoda. Do sada se istraživanje u pronalaženju odgovarajuće vrednosti parametra t razvijalo u dva pravca. Jedna grupa metoda ima za cilj da pronađe odgovarajuću fiksnu vrednost za t [33, 42, 178, 179, 190], dok metode iz druge grupe promovišu odgovarajuća pravila za izračunavanje vrednosti t u svakoj iteraciji kako bi obezbedili zadovoljavajuće smanjenje funkcije cilja. U našem istraživanju obratićemo pažnju na drugi istraživački tok: pronaći parametar t čije se vrednosti menjaju kroz iteracije tako da se postiže brža konvergencija. Vrednost parametra t definisanog u k -toj iteraciji biće označena sa t_k .

Na osnovu pregleda glavnih konjugovano gradijentnih metoda i motivisani jakim teorijskim svojstvima i računskom efikasnošću modifikovanih Dai-Liao konjugovano gradijent-

nih metoda, koje su predložili mnogi istraživači, predstavljamo u Odeljku 6.3.1 novi način izračunavanja vrednosti parametra t_k iz [87]. Kao posledica toga, predložen je odgovarajući konjugovano gradijentni metod Dai–Liao tipa pod nazivom *Efikasni Dai–Liao* (EDL) metod.

U Odeljku 6.3.2 predstavljen je još jedan način izračunavanja vrednosti parametra t_k iz [83]. Direkna posledica novog način određivanja vrednosti parametra t_k je MSMDL metod, odnosno nova varijanta Dai–Liao konjugovano gradijentnog metoda.

6.3.1 Efikasan Dai–Liao konjugovano gradijentni metod

Veliki broj istraživača u prethodne dve decenije bavio se poboljšanjem efikasnosti Dai–Liao konjugovano gradijentnog metoda i njegovih varijanti. Posebno mesto u istraživanjima ima definisanje i određivanje novih pravila za izračunavanje parametra t_k . Samim tim ovo je i garancija da se ovim pristupom značajno utiče na poboljšanju efikasnosti Dai–Liao konjugovano gradijentnog metoda i njegovih varijanti, odnosno da je još uvek ovaj metod nedovoljno istražen.

Ideja za definisanje novog pravila za izračunavanje parametra t_k^* dolazi iz prethodno opisanih pravila za izračunavanje parametra t_k u Odeljku 5.2, zatim iz rada Li i Ruan [98], i iz ideje koja se može naći u radu Yuan, Li i Hu [185]. Dalje, analizirajući rezultate iz [33, 42, 178, 179, 190], zaključujemo da je parametar t definisan fiksnom vrednošću od 0.1 u pripadajućim numeričkim eksperimentima. Takođe, numerička iskustva u vezi sa fiksnom vrednošću $t = 1$ su data u [42]. Na osnovu ovih iskustva, naš cilj je da definišemo vrednosti promenljivih t_k unutar intervala $(0, 1)$.

Da bismo uspešno definisali parametar t_k sa vrednostima koje pripadaju intervalu $(0, 1)$, počnimo od definicije veličine L_k koja je korišćena pri definisanju pravca pretraživanja $\mathbf{d}_k$ u [98]. Parametar L_k je dat u sledećem obliku

$$L_k = \frac{\mathbf{s}_{k-1}^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}^*} \in (0, 1), \quad k \geq 0,$$

gde je

$$\mathbf{y}_{k-1}^* = \mathbf{y}_{k-1} + \left(\max \left\{ 0, -\frac{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}{\|\mathbf{s}_{k-1}\|^2} \right\} + 1 \right) \mathbf{s}_{k-1}.$$

Zamenom $\mathbf{y}_{k-1}^*$ u L_k , dobija se

$$\begin{aligned} L_k &= \frac{\mathbf{s}_{k-1}^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \left(\mathbf{y}_{k-1} + \left(\max \left\{ 0, -\frac{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}{\|\mathbf{s}_{k-1}\|^2} \right\} + 1 \right) \mathbf{s}_{k-1} \right)} \\ &= \frac{\|\mathbf{s}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + \left(\max \left\{ 0, -\frac{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}{\|\mathbf{s}_{k-1}\|^2} \right\} + 1 \right) \|\mathbf{s}_{k-1}\|^2}. \end{aligned}$$

Dalje, uz određene modifikacije i zamene u jednačini koja definiše L_k , kao i korišćenjem funkcije $\max$ koja određuje maksimum između vrednosti izraza $\mathbf{d}_{k-1}^T \mathbf{g}_k$ i 1, dolazimo do

novog pravila za određivanje vrednosti parametra t_k u svakom iterativnom koraku. Na osnovu unapred zahtevanih ograničenja, novi parametar t_k^* je definisan na sledeći način:

$$t_k^* = \frac{\|\mathbf{g}_k\|^2}{\max\{1, \mathbf{d}_{k-1}^T \mathbf{g}_k\} + \left(\max\left\{0, \frac{\mathbf{d}_{k-1}^T \mathbf{g}_k}{\|\mathbf{g}_k\|^2}\right\} + 1\right) \|\mathbf{g}_k\|^2}. \quad (6.8)$$

Jednostavno se pokazuje da t_k^* definisano u (6.8) zadovoljava sledeću nejednakost

$$\begin{aligned} 0 < t_k^* &\leq \frac{\|\mathbf{g}_k\|^2}{1 + (0 + 1) \|\mathbf{g}_k\|^2} \\ &= \frac{\|\mathbf{g}_k\|^2}{1 + \|\mathbf{g}_k\|^2} \\ &< 1. \end{aligned} \quad (6.9)$$

Prema tome, $t_k^* \in (0, 1)$, što je bio naš početni cilj prilikom definisanja pravila za određivanje vrednosti parametra t_k^* u svakom iterativnom koraku. Jasno je da veće vrednosti $\|\mathbf{g}_k\|$ dovode do vrednosti $t_k^* \nearrow 1$. Dalje, pošto je trend $\|\mathbf{g}_k\| \rightarrow 0$ očekivan, možemo očekivati smanjenje vrednosti $t_k^* \searrow 0$ u kasnijim iteracijama. Zbog toga je t_k^* pogodan za definisanje odgovarajućih parametara ažuriranja konjugovanog gradijenta β_k Dai-Liao tipa.

Uzimajući u obzir $t = t_k^*$ u (5.12), logično je predložiti novu varijantu Dai-Liao konjugovanog gradijentnog parametra ažuriranja β_k^{EDL} koja je sledećeg oblika:

$$\beta_k^{\text{EDL}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^* \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}. \quad (6.10)$$

Pre definisanja glavnog algoritma, potrebno je definisati linijsko pretraživanje unazad, kao jedno od najpopularnijih i praktičnih metoda za izračunavanje dužine koraka α_k u (2.2). Procedura za linijsko pretraživanje unazad predložena u [6] počinje od početne vrednosti $\alpha = 1$ i generiše izlazne vrednosti koje obezbeđuju da se funkcija cilja smanjuje u svakoj iteraciji. Prema tome, pogodno je koristiti Algoritam 2.2.3, preuzet iz [155], da bi se odredila dužina koraka α_k .

Algoritam 6.3.1 opisuje računski postupak za Efikasan Dai-Liao (EDL) metod.

6.3.2 Modifikovani konjugovano gradijentni metod Dai-Liao tipa zasnovan na aproksimaciji Hesijanove matrice pomoću parametra ubrzanja

Uvode se i istražuju odgovarajuća ubrzanja Dai-Liao konjugovano gradijentne porodice iteracija za rešavanje velikih neograničenih optimizacionih problema. Poboljšanja su zasnovana na odgovarajućim modifikacijama parametara ažuriranja konjugovanog gradijenta u Dai-Liao konjugovano gradijentnim metodama. Glavna ideja je da se kombinuju pravci

Algoritam 6.3.1 Efikasan Dai–Liao (EDL) konjugovano gradijentni metod.

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0$ i pozitivni realni parametri $0 < \epsilon < 1$ i $0 < \delta < 1$.

1: Postaviti $k = 0$ i izračunati $\mathbf{d}_0 = -\mathbf{g}_0$.

2: Ako je

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{ i } \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq \delta,$$

STOP, inače preći na Korak 3.

3: Odrediti dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.

4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ i odrediti $\mathbf{g}_{k+1}$, $\mathbf{y}_k$ i $\mathbf{s}_k$.

5: Izračunati t_k^* koristeći (6.8).

6: Odrediti β_{k+1}^{EDL} koristeći (6.10).

7: Izračunati pravac pretraživanja $\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_{k+1}^{\text{EDL}} \mathbf{d}_k$.

8: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

pretraživanja u metodama ubrzanog opadajućeg gradijenta, definisani na osnovu Hessianove aproksimacije odgovarajućom dijagonalnom matricom u kvazi-Njutnovim metodama sa pravcima pretraživanja u konjugovano gradijentnim metodama Dai–Liao tipa.

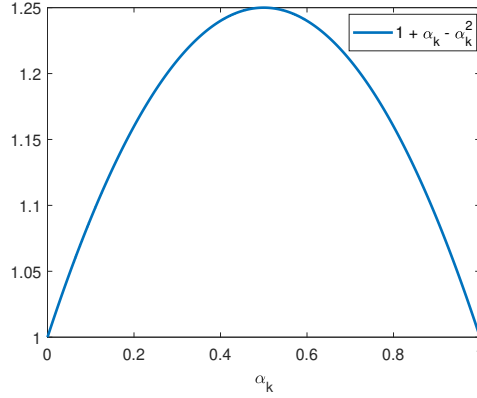
Iterativna šema za rešavanje problema bezuslovne optimizacije (2.1) za MSM metod je oblika (4.4), gde je $\gamma_{k+1}^{\text{MSM}}$ određeno sa (4.8). Kako bi zadovoljili potrebne i dovoljne uslove optimalnosti drugog reda, potrebno je nepogodne vrednosti $\gamma_{k+1}^{\text{MSM}} \leq 0$ koje se pojavljuju u (4.8) zameniti sa $\gamma_{k+1}^{\text{MSM}} = 1$. Da bi izbegli ovakve situacije, u skladu sa [86, 155], koristi se sledeći parametar ubrzanja:

$$\gamma_{k+1}^{\text{MSM}} = \begin{cases} \gamma_{k+1}^{\text{MSM}}, & \gamma_{k+1}^{\text{MSM}} > 0, \\ 1, & \gamma_{k+1}^{\text{MSM}} \leq 0, \end{cases} \quad (6.11)$$

gde je $\gamma_{k+1}^{\text{MSM}}$ definisan pomoću (4.8). Iz (4.4) imamo da se iterativna šema MSM metode može zapisati na sledeći način

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} \mathbf{g}_k = \mathbf{x}_k + \alpha_k \mathbf{d}_k^{\text{MSM}}, \quad (6.12)$$

gde je $\mathfrak{J}_k := 1 + \alpha_k - \alpha_k^2$ i $\mathbf{d}_k^{\text{MSM}} = -\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} \mathbf{g}_k$. Kako je dužina koraka $\alpha_k \in (0, 1]$, glavna ideja koja se koristi u iteracijama MSM metode je da se ubrzaju iteracije SM metode pomoću parametra $\mathfrak{J}_k := 1 + \alpha_k - \alpha_k^2 > 1$. Više detalja o metodama ubrzanog gradijenta možete pronaći u [85, 152, 155]. Kako je $\mathfrak{J}'_k(\alpha_k) = 0$ za $\alpha_k = 1/2$ i $\mathfrak{J}_k(0) = \mathfrak{J}_k(1) = 1$, matematička analiza funkcije $\mathfrak{J}_k(\alpha_k)$ na intervalu $\alpha_k \in (0, 1]$ pokazuje $\max \mathfrak{J}_k(\alpha_k) = \mathfrak{J}_k(1/2) = 5/4$ i $1 \leq \mathfrak{J}_k \leq 5/4$. Slika 6.1 predstavlja grafik funkcije $\mathfrak{J}_k(\alpha_k)$ za $\alpha_k \in (0, 1]$.



Slika 6.1: Grafik funkcije $\mathfrak{J}(\alpha_k) = 1 + \alpha_k - \alpha_k^2$ za $\alpha_k \in (0, 1]$.

Prema tome, pravac pretraživanja koji je sadržan u osnovi MSM metode u (6.12) je određen vektorom

$$\mathbf{d}_k^{\text{MSM}} = -\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} \mathbf{g}_k, \quad (6.13)$$

gde je parametar γ_k^{MSM} definisan pomoću (6.11).

S druge strane, parametar t sadržan u parametru ažuriranja konjugovanog gradijenta β_k^{DL} iz (5.9) može se odrediti zamenom (5.9) u (5.1), što dovodi do

$$\mathbf{d}_k^{\text{DL}} = -\mathbf{g}_k + \beta_k^{\text{DL}} \mathbf{d}_{k-1} = -\mathbf{g}_k + \left(\frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right) \mathbf{d}_{k-1}. \quad (6.14)$$

Nakon izjednačavanja $\mathbf{d}_k^{\text{MSM}}$ iz (6.13) sa $\mathbf{d}_k^{\text{DL}}$ iz (6.14), dobija se jednačina:

$$-\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} \mathbf{g}_k = -\mathbf{g}_k + \frac{g_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{d}_{k-1} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{d}_{k-1}. \quad (6.15)$$

Cilj je rešiti jednačinu (6.15) po t i na taj način definisati novo pravilo za određivanje Dai–Liao parametra t . Ukoliko levu i desnu stranu jednačine (6.15) pomnožite sa $\mathbf{g}_k^T$ dobija se jednačina

$$-\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} \mathbf{g}_k^T \mathbf{g}_k = -\mathbf{g}_k^T \mathbf{g}_k + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1}. \quad (6.16)$$

Prema tome, iz (6.16) dobija se

$$\begin{aligned} t \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} &= -\|\mathbf{g}_k\|^2 + \mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} \|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= (\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1) \|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1}. \end{aligned} \quad (6.17)$$

Sada iz jednačine (6.17) parametar t može se odrediti na sledeći način

$$t = \frac{(\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1) \|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1}}{\frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1}}. \quad (6.18)$$

Kako je $\mathbf{s}_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1} = \mathbf{x}_{k-1} + \alpha_{k-1} \mathbf{d}_{k-1} - \mathbf{x}_{k-1} = \alpha_{k-1} \mathbf{d}_{k-1}$ i $\alpha_{k-1} > 0$, nakon zamene $\mathbf{d}_{k-1} = \mathbf{s}_{k-1}/\alpha_{k-1}$ u (6.18) i sređivanja izraza dobija se

$$\begin{aligned} t &= \frac{(\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1) \|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\frac{1}{\alpha_{k-1}} \mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \frac{1}{\alpha_{k-1}} \mathbf{s}_{k-1}}{\frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\frac{1}{\alpha_{k-1}} \mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \frac{1}{\alpha_{k-1}} \mathbf{s}_{k-1}} \\ &= \frac{(\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1) \|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{s}_{k-1}}{\frac{(\mathbf{g}_k^T \mathbf{s}_{k-1})^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}} \\ &= \frac{(\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1) \|\mathbf{g}_k\|^2 \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + \mathbf{g}_k^T \mathbf{y}_{k-1} \mathbf{g}_k^T \mathbf{s}_{k-1}}{(\mathbf{g}_k^T \mathbf{s}_{k-1})^2} = \tau_k^{\text{MSM}}. \end{aligned} \quad (6.19)$$

Poznato je da je Dai–Liao parametar izračunat da generiše pravac maksimalnog poboljšanja, koristeći matricu pravca pretraživanja koja je ortogonalna na vektor gradijenta [5]. Da bi se osigurali da novi Dai–Liao metod zadovoljava uslov opadanja, definicija za t u (6.19) se menja korišćenjem ideja iz [5, 107] u sledeći oblik

$$t_k^{\text{MSMDL}} = \max \left\{ \tau_k^{\text{MSM}}, \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \right\}, \quad \theta > \frac{1}{4}. \quad (6.20)$$

Uzimajući $t := t_k^{\text{MSMDL}}$ u (5.9), predložena je nova varijanta Dai–Liao konjugovano gradijentnog parametra ažuriranja β_k^{MSMDL} koja je oblika:

$$\beta_k^{\text{MSMDL}} = \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^{\text{MSMDL}} \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}. \quad (6.21)$$

Pre definisanja algoritma za MSMDL metod, koji je baziran na jednačinama (2.2), (5.1), (6.20) i (6.21), potrebno je definisati linijsko pretraživanje unazad za određivanje dužine koraka α_k . Za određivanje dužine koraka α_k koristi se Algoritam 2.2.3.

Algoritam 6.3.2 daje detaljan postupak za primenu MSMDL metode.

Prethodno opisana strategija za kombinovanje MSM i Dai–Liao pristupa može se primeniti na bilo koji kvazi-Njutnov pravac. Ako se izvrši zamena $\gamma_{k+1}^{\text{MSM}}$ sa γ_k^{BB1} iz (3.29), dobija se novi BB1DL metod. Analognim postupkom dobija se

$$t = \frac{((\gamma_k^{\text{BB1}})^{-1} - 1) \|\mathbf{g}_k\|^2 \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + \mathbf{g}_k^T \mathbf{y}_{k-1} \mathbf{g}_k^T \mathbf{s}_{k-1}}{(\mathbf{g}_k^T \mathbf{s}_{k-1})^2} = \tau_k^{\text{BB1}}$$

Algoritam 6.3.2 MSMDL metod

Ulaz: Funkcija cilja $f(\mathbf{x})$, početna aproksimacija $\mathbf{x}_0 \in \text{dom}(f)$ i parametri $0 < \varepsilon \ll 1$, $0 < \delta \ll 1$.

- 1: Postaviti $k = 0$, $\gamma_0 = 1$ i izračunati $f(\mathbf{x}_0)$, $\mathbf{g}_0 = \nabla f(\mathbf{x}_0)$, $\mathbf{d}_0 = -\mathbf{g}_0$.
- 2: Ako su test kriterijumi

$$\|\mathbf{g}_k\| \leq \varepsilon \quad \text{i} \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq \delta,$$

ispunjeni, onda STOP, inače preći na Korak 3.

- 3: Odrediti dužinu koraka α_k koristeći Algoritam 2.2.3.
- 4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ i odrediti $f(\mathbf{x}_{k+1})$ i $\mathbf{g}_{k+1} = \nabla f(\mathbf{x}_{k+1})$.
- 5: Odrediti $\gamma_{k+1}^{\text{MSM}}$ koristeći (6.11).
- 6: Izračunati t_{k+1}^{MSMDL} koristeći (6.20).
- 7: Izračunati $\beta_{k+1}^{\text{MSMDL}}$ koristeći (6.21).
- 8: Odrediti pravac pretraživanja $\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_{k+1}^{\text{MSMDL}} \mathbf{d}_k$.
- 9: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_{k+1}$ i $f(\mathbf{x}_{k+1})$.

i

$$t_k^{\text{BB1DL}} = \max \left\{ \tau_k^{\text{BB1}}, \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \right\}, \quad \theta > \frac{1}{4}. \quad (6.22)$$

Dalje, zamena $t := t_k^{\text{BB1DL}}$ u (5.9) inicira novu varijantu Dai–Liao konjugovano gradijentnog parametra ažuriranja β_k^{BB1DL} :

$$\beta_k^{\text{BB1DL}} = \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^{\text{BB1DL}} \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}. \quad (6.23)$$

Ako se primene date izmene, dolazi se do nove varijante Algoritma 6.3.2, gde se umesto 5., 6., 7. i 8. koraka u MSMDL metodi koriste sledeći koraci:

- Korak 5*:** Odrediti $\gamma_{k+1}^{\text{BB1}}$ koristeći (3.29).
Korak 6*: Izračunati t_{k+1}^{BB1DL} koristeći (6.22).
Korak 7*: Izračunati $\beta_{k+1}^{\text{BB1DL}}$ koristeći (6.23).
Korak 8*: Odrediti $\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_{k+1}^{\text{BB1DL}} \mathbf{d}_k$.

Varijanta Algoritma 6.3.2 zasnovana na koracima 1:–4:;5*:,6*:,7*:,8*: i 9: biće nazvana BB1DL metod. Tačnije, BB1DL metod je zasnovan na jednačinama (2.2), (5.1), (6.22) i (6.23).

Na osnovu svega gore navedenog, može se zaključiti da je opšti okvir predstavljen Algoritmom 6.3.2 primenljiv i na druge kvazi-Njutnove metode.

6.4 Analiza konvergencije

U okviru ovog odeljka, pod standardnim pretpostavkama izvršena je analiza konvergencije za MLSCD (Algoritam 6.1.1), MMDL (Algoritam 6.2.1), EDL (Algoritam 6.3.1) i MSMDL (Algoritam 6.3.2) metode. Osobine konvergencije navedenih metoda su ispitane na skupu uniformno konveksnih funkcija.

U dokazima definisanih tvrđenja u nastavku odeljka podrazumeva se važenje Stava 4.5.1 i Leme 4.5.1 datih u Poglavlju 4.

6.4.1 Analiza konvergencije MLSCD i MMDL metode

Globalna konvergencija predloženih hibridnih varijanti konjugovano gradijentnih metoda je dokazana u nastavku na osnovu standardnih pretpostavki.

Teorema 6.4.1. *Ako je β_k bilo koji parametar ažuriranja konjugovano gradijentnih metoda. Onda pravac pretraživanja $\mathbf{d}_k$ definisan u (6.1) zadovoljava*

$$\mathbf{g}_k^T \mathbf{d}_k = -\|\mathbf{g}_k\|^2. \quad (6.24)$$

Dokaz. Da bi dokazali jednakost (6.24) potrebno je levu i desnu stranu jednačine (6.1) pomnožiti sa $\mathbf{g}_k^T$. Nakon množenja i sređivanja dobija se jednakost

$$\begin{aligned} \mathbf{g}_k^T \mathbf{d}_k &= - \left(1 + \beta_k \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2} \right) \mathbf{g}_k^T \mathbf{g}_k + \beta_k \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= -\|\mathbf{g}_k\|^2 - \beta_k \mathbf{g}_k^T \mathbf{d}_{k-1} + \beta_k \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= -\|\mathbf{g}_k\|^2, \end{aligned} \quad (6.25)$$

čime je dokaz završen. $\square$

Ishod sledeće leme, koji se često naziva Zoutendijk-ov uslov, koristi se za dokazivanje globalne konvergencije nelinearnih konjugovano gradijentnih metoda. Originalno je data u [194].

Lema 6.4.1. [32, 194] *Neka važe pretpostavke (H_1) i (H_2) iz Stava 4.5.1 i neka je niz $\{\mathbf{x}_k\}$ generisan MLSDL metodom sa linijskim pretrživanjem unazad. Onda je*

$$\sum_{k=0}^{\infty} \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} < +\infty. \quad (6.26)$$

Dokaz. Posmatraju se dva slučaja kada se koristi linijsko pretrživanje unazad. Ako je $\alpha_k \neq \zeta$, onda nije teško pokazati da postoji konstanta $c > 0$ takva da

$$\alpha_k \geq c \frac{-\mathbf{g}_k^T \mathbf{d}_k}{\|\mathbf{d}_k\|^2} = c \frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_k\|^2}. \quad (6.27)$$

Ovo zajedno sa linijskim pretrživanjem unazad implicira postojanje konstante $M > 0$ takve da

$$\frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \leq M(f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})). \quad (6.28)$$

S druge strane, ako je $\alpha_k = \zeta$ onda sledi da

$$\frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \leq \|\mathbf{g}_k\|^2 \leq \|\mathbf{d}_k\|^2 \leq \delta^{-1}\zeta^{-2}(f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})).$$

Ukoliko se u poslednjoj nejednačini oznaci sa $M = \delta^{-1}\zeta^{-2}$, onda važi (6.28). $\square$

U Teoremi 6.4.2 dokazana je globalna konvergencija MLSCD metode.

Teorema 6.4.2. *Neka važe pretpostavke (H_1) i (H_2) iz Stava 4.5.1. Onda niz $\{\mathbf{x}_k\}$, generisan MLSDL metodom sa linijskim pretrživanjem unazad zadovoljava*

$$\liminf_{k \rightarrow \infty} \|\mathbf{g}_k\| = 0. \quad (6.29)$$

Dokaz. Da bismo teoremu dokazali kontradikcijom pretpostavimo da (6.29) ne važi. Zatim, pretpostavimo da postoji konstanta $c > 0$ takva da

$$\|\mathbf{g}_k\| \geq c, \text{ za svako } k \geq 0. \quad (6.30)$$

Pravac pretraživanja $\mathbf{d}_k$ iz (6.3) dat je u sledećem obliku

$$\mathbf{d}_k = -l_k \mathbf{g}_k + \beta_k^{LSCD} \mathbf{d}_{k-1}, \text{ gde je } l_k = 1 + \beta_k^{LSCD} \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}. \quad (6.31)$$

Sada, iz (6.31) imamo da važi

$$\mathbf{d}_k + l_k \mathbf{g}_k = \beta_k^{LSCD} \mathbf{d}_{k-1},$$

što dalje implicira

$$\begin{aligned} (\mathbf{d}_k + l_k \mathbf{g}_k)^2 &= (\beta_k^{LSCD} \mathbf{d}_{k-1})^2, \\ \|\mathbf{d}_k\|^2 + 2l_k \mathbf{d}_k^T \mathbf{g}_k + l_k^2 \|\mathbf{g}_k\|^2 &= (\beta_k^{LSCD})^2 \|\mathbf{d}_{k-1}\|^2, \end{aligned}$$

odnosno

$$\|\mathbf{d}_k\|^2 = (\beta_k^{LSCD})^2 \|\mathbf{d}_{k-1}\|^2 - 2l_k \mathbf{d}_k^T \mathbf{g}_k - l_k^2 \|\mathbf{g}_k\|^2. \quad (6.32)$$

Iz definicije parametra ažuriranja konjugovanog gradijenta β_k^{LSCD} imamo da važi

$$\beta_k^{LSCD} = \max\{0, \min\{\beta_k^{LS}, \beta_k^{CD}\}\} \leq |\beta_k^{CD}|. \quad (6.33)$$

Podelimo obe strane u (6.32) sa $(\mathbf{g}_k^T \mathbf{d}_k)^2$, a zatim primenimo (6.24), (6.33) i definiciju za

β_k^{CD} :

$$\begin{aligned}
 \frac{\|\mathbf{d}_k\|^2}{\|\mathbf{g}_k\|^4} &= \frac{\|\mathbf{d}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} = (\beta_k^{LSCD})^2 \frac{\|\mathbf{d}_{k-1}\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - \frac{2l_k \mathbf{d}_k^T \mathbf{g}_k}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} \\
 &\leq (\beta_k^{CD})^2 \frac{\|\mathbf{d}_{k-1}\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - \frac{2l_k}{\mathbf{g}_k^T \mathbf{d}_k} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} \\
 &= \left(\frac{\|\mathbf{g}_k\|^2}{-\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}} \right)^2 \frac{\|\mathbf{d}_{k-1}\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - \frac{2l_k}{\mathbf{g}_k^T \mathbf{d}_k} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2}.
 \end{aligned} \tag{6.34}$$

Sada, primenom (6.24) i (6.30) u (6.34), dobija se

$$\begin{aligned}
 \frac{\|\mathbf{d}_k\|^2}{\|\mathbf{g}_k\|^4} &\leq \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{g}_{k-1}\|^4} \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} + \frac{2l_k}{\|\mathbf{g}_k\|^2} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{\|\mathbf{g}_k\|^4} \\
 &= \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^4} + \frac{2l_k}{\|\mathbf{g}_k\|^2} - l_k^2 \frac{1}{\|\mathbf{g}_k\|^2} \\
 &= \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^4} - \frac{(l_k - 1)^2}{\|\mathbf{g}_k\|^2} + \frac{1}{\|\mathbf{g}_k\|^2} \\
 &\leq \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^4} + \frac{1}{\|\mathbf{g}_k\|^2} \\
 &\leq \sum_{j=0}^k \frac{1}{\|\mathbf{g}_j\|^2} \\
 &\leq \frac{k+1}{c^2}.
 \end{aligned}$$

Poslednja nejednakost implicira

$$\sum_{k \geq 1} \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \geq c^2 \sum_{k \geq 1} \frac{1}{k+1} = \infty, \tag{6.35}$$

što je u suprotnosti sa (6.26), čime je dokaz završen. $\square$

Globalna konvergencija MMDL metode se dokazuje u Teoremi 6.4.3.

Teorema 6.4.3. *Neka važe pretpostavke (H_1) i (H_2) iz Stava 4.5.1. Onda niz $\{\mathbf{x}_k\}$, generisan MMDL metodom sa linijskim pretrživanjem unazad zadovoljava*

$$\liminf_{k \rightarrow \infty} \|\mathbf{g}_k\| = 0. \tag{6.36}$$

Dokaz. Pretpostavimo suprotno da (6.36) ne važi. Tada postoji konstanta $c > 0$ takva da

$$\|\mathbf{g}_k\| \geq c, \text{ za svako } k \geq 0. \tag{6.37}$$

Ako označimo sa

$$l_k = 1 + \beta_k^{MMDL} \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}, \tag{6.38}$$

onda se pravac pretaživanja $\mathbf{d}_k$ iz (6.7) može zapisati u sledećem obliku

$$\mathbf{d}_k + l_k \mathbf{g}_k = \beta_k^{MMDL} \mathbf{d}_{k-1}, \quad (6.39)$$

što dalje implicira

$$\begin{aligned} (\mathbf{d}_k + l_k \mathbf{g}_k)^2 &= (\beta_k^{MMDL} \mathbf{d}_{k-1})^2, \\ \|\mathbf{d}_k\|^2 + 2l_k \mathbf{d}_k^T \mathbf{g}_k + l_k^2 \|\mathbf{g}_k\|^2 &= (\beta_k^{MMDL})^2 \|\mathbf{d}_{k-1}\|^2, \end{aligned}$$

odnosno

$$\|\mathbf{d}_k\|^2 = (\beta_k^{MMDL})^2 \|\mathbf{d}_{k-1}\|^2 - 2l_k \mathbf{d}_k^T \mathbf{g}_k - l_k^2 \|\mathbf{g}_k\|^2. \quad (6.40)$$

Imajući u vidu da je $\mu > 1$, kao i da je $\mathbf{d}_k^T \mathbf{g}_k < 0$ i primenom uslova **proširene konjugacije** (*eng. extended conjugacy*) $\mathbf{d}_k^T \mathbf{y}_{k-1} = t \mathbf{g}_k^T \mathbf{s}_{k-1}$, $t > 0$, koji je eksploatisan u [42, 190], dobija se

$$\begin{aligned} \beta_{k+1}^{DHSDL} &= \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| + \mathbf{d}_k^T \mathbf{y}_k} - \alpha_k \frac{\mathbf{g}_{k+1}^T \mathbf{s}_k}{\mathbf{d}_k^T \mathbf{y}_k} \\ &\leq \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| + \mathbf{d}_k^T \mathbf{y}_k} \\ &= \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| + \mathbf{d}_k^T (\mathbf{g}_{k+1} - \mathbf{g}_k)} \\ &= \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| + \mathbf{d}_k^T \mathbf{g}_{k+1} - \mathbf{d}_k^T \mathbf{g}_k} \\ &\leq \frac{\|\mathbf{g}_{k+1}\|^2}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| + \mathbf{d}_k^T \mathbf{g}_{k+1} - \mathbf{d}_k^T \mathbf{g}_k} \\ &\leq \frac{\|\mathbf{g}_{k+1}\|^2}{-\mathbf{d}_k^T \mathbf{g}_k}. \end{aligned}$$

Dalje za β_k^{DLSDL} koristeći prethodno navedene činjenice sledi da važi

$$\begin{aligned} \beta_{k+1}^{DLSDL} &= \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| - \mathbf{d}_k^T \mathbf{g}_k} - \alpha_k \frac{\mathbf{g}_{k+1}^T \mathbf{s}_k}{\mathbf{d}_k^T \mathbf{y}_k} \\ &\leq \frac{\|\mathbf{g}_{k+1}\|^2 - \frac{\|\mathbf{g}_{k+1}\|}{\|\mathbf{g}_k\|} |\mathbf{g}_{k+1}^T \mathbf{g}_k|}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| - \mathbf{d}_k^T \mathbf{g}_k} \\ &\leq \frac{\|\mathbf{g}_{k+1}\|^2}{\mu |\mathbf{g}_{k+1}^T \mathbf{d}_k| - \mathbf{d}_k^T \mathbf{g}_k} \\ &\leq \frac{\|\mathbf{g}_{k+1}\|^2}{-\mathbf{d}_k^T \mathbf{g}_k}. \end{aligned}$$

Sada, na osnovu dobijenih nejdnakosti za β_k^{DHSDL} i β_k^{DLSDL} može se izvesti sledeći zaključak

$$\beta_k^{MMDL} = \max\{0, \min\{\beta_k^{DHSDL}, \beta_k^{DLSDL}\}\} \leq \frac{\|\mathbf{g}_k\|^2}{-\mathbf{d}_{k-1}^T \mathbf{g}_{k-1}}. \quad (6.41)$$

Dalje, podelimo obe strane u (6.40) sa $(\mathbf{g}_k^T \mathbf{d}_k)^2$, a zatim primenimo (6.24), (6.41) i (6.37)

$$\begin{aligned} \frac{\|\mathbf{d}_k\|^2}{\|\mathbf{g}_k\|^4} &= \frac{\|\mathbf{d}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} = (\beta_k^{MMDL})^2 \frac{\|\mathbf{d}_{k-1}\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - \frac{2l_k \mathbf{d}_k^T \mathbf{g}_k}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} \\ &= (\beta_k^{MMDL})^2 \frac{\|\mathbf{d}_{k-1}\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - \frac{2l_k}{\mathbf{g}_k^T \mathbf{d}_k} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} \\ &\leq \left(\frac{\|\mathbf{g}_k\|^2}{-\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}} \right)^2 \frac{\|\mathbf{d}_{k-1}\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} - \frac{2l_k}{\mathbf{g}_k^T \mathbf{d}_k} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{(\mathbf{g}_k^T \mathbf{d}_k)^2} \\ &= \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{g}_{k-1}\|^4} \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} + \frac{2l_k}{\|\mathbf{g}_k\|^2} - l_k^2 \frac{\|\mathbf{g}_k\|^2}{\|\mathbf{g}_k\|^4} \\ &= \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^4} + \frac{2l_k}{\|\mathbf{g}_k\|^2} - l_k^2 \frac{1}{\|\mathbf{g}_k\|^2} \\ &= \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^4} - \frac{(l_k - 1)^2}{\|\mathbf{g}_k\|^2} + \frac{1}{\|\mathbf{g}_k\|^2} \\ &\leq \frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^4} + \frac{1}{\|\mathbf{g}_k\|^2} \\ &\leq \sum_{j=0}^k \frac{1}{\|\mathbf{g}_j\|^2} \\ &\leq \frac{k+1}{c^2}. \end{aligned} \quad (6.42)$$

Nejdnakosti u (6.42) impliciraju

$$\sum_{k \geq 1} \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \geq c^2 \sum_{k \geq 1} \frac{1}{k+1} = \infty. \quad (6.43)$$

Na osnovu čega sledi da $\|\mathbf{g}_k\| \geq c$ izaziva kontradikciju sa (6.26). Prema tome važi (6.36), čime je dokaz završen. $\square$

6.4.2 Analiza konvergencije EDL i MSMDL metode

Globalna konvergencija predloženih varijanti Dai–Liao konjugovano gradijentnih metoda je izvedena na osnovu standardnih pretpostavki.

Pretpostavke (H_1) i (H_2) iz Stava 4.5.1 iniciraju postojanje pozitivnih konstanti D i γ koje zadovoljavaju

$$(\forall \mathbf{u}, \mathbf{v} \in \mathcal{B}) \quad \|\mathbf{u} - \mathbf{v}\| \leq D \quad (6.44)$$

i

$$(\forall \mathbf{u} \in \mathcal{B}) \quad \|\mathbf{g}(\mathbf{u})\| \leq \gamma. \quad (6.45)$$

Drugi glavni element u dokazivanju konvergencije konjugovano gradijentne metode je svojstvo uniformno konveksnih funkcija

$$\mathbf{s}_{k-1}^T \mathbf{y}_{k-1} \geq \eta \|\mathbf{s}_{k-1}\|^2, \quad (6.46)$$

gde je $\eta > 0$. Verifikacija ove osobine se može naći u [162, Teorema 1.3.16]. Iz (4.32) sledi $\|\mathbf{y}_{k-1}\| \leq L \|\mathbf{s}_{k-1}\|$, gde zajedno sa (6.46) inicira

$$\eta \|\mathbf{s}_{k-1}\|^2 \leq \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} \leq L \|\mathbf{s}_{k-1}\|^2. \quad (6.47)$$

Jasno je da nejednakost (6.47) implicira $\eta \leq L$. Dalje, iz nejednačine (6.47) imamo da važi

$$\mathbf{s}_{k-1}^T \mathbf{y}_{k-1} = \alpha_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1} > 0. \quad (6.48)$$

Uzimajući u obzir da je $\alpha_{k-1} > 0$ u (6.48), može se zaključiti sledeće

$$\mathbf{d}_{k-1}^T \mathbf{y}_{k-1} > 0. \quad (6.49)$$

Tvrđenje iz Leme 6.4.2 koristi se za dokazivanje globalne konvergencije konjugovano gradijentnih metoda. Tačnost ovog tvrđenja može se proveriti na osnovu rezultata datih u [32, Lema 2.2] i [194].

Lema 6.4.2. *Neka važe pretpostavke (H_1) i (H_2) iz Stava 4.5.1 i neka je niz $\{\mathbf{x}_k\}$ generisan pomoću EDL ili MSMDL metode sa linijskim pretraživanjem unazad. Onda je*

$$\sum_{k=0}^{\infty} \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} < +\infty. \quad (6.50)$$

Lema 6.4.3. *Razmotrimo predloženi EDL konjugovani gradijentni metod određen sa (2.2), (5.1) i (6.10). Ako pravac pretraživanja zadovoljava uslov (6.49) za svako $k \geq 0$, onda važi sledeća nejednakost*

$$\mathbf{g}_k^T \mathbf{d}_k \leq -c \|\mathbf{g}_k\|^2, \quad (6.51)$$

za neko $0 \leq c \leq 1$.

Dokaz. Nejednakost (6.51) biće dokazana indukcijom. Za $k = 0$ imamo da važi $\mathbf{g}_0^T \mathbf{d}_0 = -\|\mathbf{g}_0\|^2$. Kako je $c \leq 1$, očigledno je nejednakost (6.51) zadovoljena u ovom slučaju. Pretpostavimo da (6.51) važi za neko $k \geq 1$. Ukoliko pomnožimo sa vektorom $\mathbf{g}_k^T$ levu i desnu

stranu u (5.1), dobija se

$$\begin{aligned}
 \mathbf{g}_k^T \mathbf{d}_k &= -\|\mathbf{g}_k\|^2 + \beta_k^{\text{EDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\
 &= -\|\mathbf{g}_k\|^2 + \left(\frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^* \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right) \mathbf{g}_k^T \mathbf{d}_{k-1} \\
 &= -\|\mathbf{g}_k\|^2 + \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - t_k^* \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\
 &= -\|\mathbf{g}_k\|^2 + \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - t_k^* \frac{\alpha_{k-1} (\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}.
 \end{aligned} \tag{6.52}$$

Koristeći (6.9) zajedno sa (6.49) i $\alpha_{k-1} > 0$, imamo da važi

$$t_k^* \frac{\alpha_{k-1} (\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} > 0. \tag{6.53}$$

Sada iz (6.52), (6.53) i

$$0 \leq \beta_k^{\text{MHS}} = \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \leq \frac{\|\mathbf{g}_k\|^2}{\lambda |\mathbf{g}_k^T \mathbf{d}_{k-1}|}, \quad \lambda \geq 1, \tag{6.54}$$

sledi da

$$\begin{aligned}
 \mathbf{g}_k^T \mathbf{d}_k &\leq -\|\mathbf{g}_k\|^2 + \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\
 &\leq -\|\mathbf{g}_k\|^2 + \frac{\|\mathbf{g}_k\|^2}{\lambda |\mathbf{g}_k^T \mathbf{d}_{k-1}|} |\mathbf{g}_k^T \mathbf{d}_{k-1}| \\
 &= -\left(1 - \frac{1}{\lambda}\right) \|\mathbf{g}_k\|^2.
 \end{aligned} \tag{6.55}$$

Kako je $\lambda \geq 1$, nejednakost (6.51) je zadovoljena za $c = (1 - \frac{1}{\lambda})$ u (6.55) i proizvoljno $k \geq 0$. $\square$

Lema 6.4.4. *Neka važe pretpostavke (H_1) i (H_2) iz Stava 4.5.1, funkcija f je uniformno konveksna i konjugovano gradijentni parametar ažuriranja (6.21) ispunjava uslov $t_k^{\text{MSMDL}} \geq \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}$, za svako $k \geq 0$ i za $\theta > \frac{1}{4}$. Tada MSMDL metod zadovoljava dovoljan uslov opadanja*

$$\mathbf{g}_k^T \mathbf{d}_k \leq -c \|\mathbf{g}_k\|^2 \tag{6.56}$$

za $c = 1 - \frac{1}{4\theta}$.

Dokaz. Pretpostavke iz Stava 4.5.1 garantuju (6.49) za pravce pretraživanja (5.1) u predloženom MSMDL metodu. Nejednakost (6.56) biće dokazana indukcijom. Za $k = 0$ imamo da važi $\mathbf{g}_0^T \mathbf{d}_0 = -\|\mathbf{g}_0\|^2 \leq -c \|\mathbf{g}_0\|^2$. Prema tome, nejednakost (6.56) je zadovoljena za $k = 0$. Pretpostavimo da nejednakost (6.56) važi za neko $k \geq 1$. Ukoliko za slučaj $k + 1$ jednačinu

(5.1) koja odgovara MSMDL metodi pomnožimo sa vektorom $\mathbf{g}_{k+1}^T$ sa leve i sa desne strane, dobija se

$$\begin{aligned}
 \mathbf{g}_{k+1}^T \mathbf{d}_{k+1} &= -\|\mathbf{g}_{k+1}\|^2 + \beta_{k+1}^{\text{MSMDL}} \mathbf{g}_{k+1}^T \mathbf{d}_k \\
 &= -\|\mathbf{g}_{k+1}\|^2 + \left(\frac{\mathbf{g}_{k+1}^T \mathbf{y}_k}{\mathbf{d}_k^T \mathbf{y}_k} - t_{k+1}^{\text{MSMDL}} \frac{\mathbf{g}_{k+1}^T \mathbf{s}_k}{\mathbf{d}_k^T \mathbf{y}_k} \right) \mathbf{g}_{k+1}^T \mathbf{d}_k \\
 &= -\|\mathbf{g}_{k+1}\|^2 + \frac{\mathbf{g}_{k+1}^T \mathbf{y}_k}{\mathbf{d}_k^T \mathbf{y}_k} \mathbf{g}_{k+1}^T \mathbf{d}_k - t_{k+1}^{\text{MSMDL}} \frac{\mathbf{g}_{k+1}^T \mathbf{s}_k}{\mathbf{d}_k^T \mathbf{y}_k} \mathbf{g}_{k+1}^T \mathbf{d}_k \\
 &= -\|\mathbf{g}_{k+1}\|^2 + \frac{\mathbf{g}_{k+1}^T \mathbf{y}_k}{\mathbf{d}_k^T \mathbf{y}_k} \mathbf{g}_{k+1}^T \mathbf{d}_k - t_{k+1}^{\text{MSMDL}} \frac{\alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2}{\mathbf{d}_k^T \mathbf{y}_k}. \tag{6.57}
 \end{aligned}$$

Sada iz jednakosti (6.57) sledi

$$\begin{aligned}
 \mathbf{g}_{k+1}^T \mathbf{d}_{k+1} &= -\|\mathbf{g}_{k+1}\|^2 + \frac{\mathbf{g}_{k+1}^T \mathbf{y}_k}{\mathbf{d}_k^T \mathbf{y}_k} \mathbf{g}_{k+1}^T \mathbf{d}_k - \frac{t_{k+1}^{\text{MSMDL}} \alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2}{\mathbf{d}_k^T \mathbf{y}_k} \\
 &= \frac{-\|\mathbf{g}_{k+1}\|^2 (\mathbf{d}_k^T \mathbf{y}_k)^2 + (\mathbf{g}_{k+1}^T \mathbf{y}_k) (\mathbf{d}_k^T \mathbf{y}_k) (\mathbf{g}_{k+1}^T \mathbf{d}_k)}{(\mathbf{d}_k^T \mathbf{y}_k)^2} - \frac{t_{k+1}^{\text{MSMDL}} \alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2 (\mathbf{d}_k^T \mathbf{y}_k)}{(\mathbf{d}_k^T \mathbf{y}_k)^2}. \tag{6.58}
 \end{aligned}$$

Koristeći nejednakost $\mathbf{u}^T \mathbf{v} \leq \frac{1}{2}(\|\mathbf{u}\|^2 + \|\mathbf{v}\|^2)$, gde je $\mathbf{u} = \frac{1}{\sqrt{2\theta}}(\mathbf{d}_k^T \mathbf{y}_k) \mathbf{g}_{k+1}$ i $\mathbf{v} = \sqrt{2\theta}(\mathbf{g}_{k+1}^T \mathbf{d}_k) \mathbf{y}_k$, dobija se

$$\begin{aligned}
 \mathbf{g}_{k+1}^T \mathbf{d}_{k+1} &\leq \frac{\|\mathbf{g}_{k+1}\|^2 (\mathbf{d}_k^T \mathbf{y}_k)^2 + \frac{1}{2} \left(\left\| \frac{1}{\sqrt{2\theta}} (\mathbf{d}_k^T \mathbf{y}_k) \mathbf{g}_{k+1} \right\|^2 + \left\| \sqrt{2\theta} (\mathbf{g}_{k+1}^T \mathbf{d}_k) \mathbf{y}_k \right\|^2 \right)}{(\mathbf{d}_k^T \mathbf{y}_k)^2} \\
 &\quad - \frac{t_{k+1}^{\text{MSMDL}} \alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2 (\mathbf{d}_k^T \mathbf{y}_k)}{(\mathbf{d}_k^T \mathbf{y}_k)^2} \\
 &= \frac{-\|\mathbf{g}_{k+1}\|^2 (\mathbf{d}_k^T \mathbf{y}_k)^2 + \frac{1}{2} \left(\frac{1}{2\theta} (\mathbf{d}_k^T \mathbf{y}_k)^2 \|\mathbf{g}_{k+1}\|^2 + 2\theta (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2 \|\mathbf{y}_k\|^2 \right)}{(\mathbf{d}_k^T \mathbf{y}_k)^2} \\
 &\quad - \frac{t_{k+1}^{\text{MSMDL}} \alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2}{\mathbf{d}_k^T \mathbf{y}_k} \\
 &= \frac{-\|\mathbf{g}_{k+1}\|^2 (\mathbf{d}_k^T \mathbf{y}_k)^2 + \frac{1}{4\theta} (\mathbf{d}_k^T \mathbf{y}_k)^2 \|\mathbf{g}_{k+1}\|^2 + \theta (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2 \|\mathbf{y}_k\|^2}{(\mathbf{d}_k^T \mathbf{y}_k)^2} - t_{k+1}^{\text{MSMDL}} \frac{\alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2}{\mathbf{d}_k^T \mathbf{y}_k} \\
 &\leq -\|\mathbf{g}_{k+1}\|^2 + \frac{1}{4\theta} \|\mathbf{g}_{k+1}\|^2 + \frac{\theta (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2 \|\mathbf{y}_k\|^2}{(\mathbf{d}_k^T \mathbf{y}_k)^2} - \theta \frac{\|\mathbf{y}_k\|^2}{\mathbf{s}_k^T \mathbf{y}_k} \frac{\alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2}{\mathbf{d}_k^T \mathbf{y}_k} \\
 &= -\|\mathbf{g}_{k+1}\|^2 + \frac{1}{4\theta} \|\mathbf{g}_{k+1}\|^2 + \frac{\theta (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2 \|\mathbf{y}_k\|^2}{(\mathbf{d}_k^T \mathbf{y}_k)^2} - \theta \frac{\|\mathbf{y}_k\|^2}{\alpha_k \mathbf{d}_k^T \mathbf{y}_k} \frac{\alpha_k (\mathbf{g}_{k+1}^T \mathbf{d}_k)^2}{\mathbf{d}_k^T \mathbf{y}_k} \\
 &= -\left(1 - \frac{1}{4\theta}\right) \|\mathbf{g}_{k+1}\|^2. \tag{6.59}
 \end{aligned}$$

Kako je $\theta > \frac{1}{4}$ nejednakost (6.56) je zadovoljena za $c = 1 - \frac{1}{4\theta}$ u (6.59) i proizvoljno $k \geq 0$. $\square$

Globalnu konvergenciju predložene EDL metode potvrđuje Teorema 6.4.4.

Teorema 6.4.4. *Neka važe pretpostavke (H_1) i (H_2) iz Stava 4.5.1 i neka je funkcija f uniformno konveksna. Onda za niz $\{\mathbf{x}_k\}$ generisan pomoću EDL metode određene sa (2.2), (5.1) i (6.10) važi*

$$\liminf_{k \rightarrow \infty} \|\mathbf{g}_k\| = 0. \quad (6.60)$$

Dokaz. Pretpostavimo suprotno, tj. da (6.60) ne važi. Onda ovo inicira postojanje konstante $c_1 > 0$ takve da

$$\|\mathbf{g}_k\| \geq c_1, \quad \text{za svako } k \geq 0. \quad (6.61)$$

Kvadriranjem obe strane izraza (5.1) dobija se

$$\|\mathbf{d}_k\|^2 = \|\mathbf{g}_k\|^2 - 2\beta_k^{\text{EDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} + (\beta_k^{\text{EDL}})^2 \|\mathbf{d}_{k-1}\|^2. \quad (6.62)$$

Uzimajući u obzir (6.10), sledi

$$\begin{aligned} -2\beta_k^{\text{EDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} &= -2 \left(\frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^* \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right) \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= -2 \left(\frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - t_k^* \frac{\alpha_{k-1} (\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right). \end{aligned} \quad (6.63)$$

Sada iz (6.53) i (6.54), sledi da je

$$\begin{aligned} -2\beta_k^{\text{EDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} &\leq 2 \left| \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}|}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right| |\mathbf{g}_k^T \mathbf{d}_{k-1}| \\ &\leq 2 \frac{\|\mathbf{g}_k\|^2}{\lambda |\mathbf{g}_k^T \mathbf{d}_{k-1}|} |\mathbf{g}_k^T \mathbf{d}_{k-1}| \\ &= 2 \frac{\|\mathbf{g}_k\|^2}{\lambda}. \end{aligned} \quad (6.64)$$

Dalje iz (6.10) imamo da važi

$$\begin{aligned}
 \beta_k^{\text{EDL}} &= \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}| - t_k^* \mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\
 &\leq \left| \frac{\mathbf{g}_k^T \mathbf{g}_k - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} |\mathbf{g}_k^T \mathbf{g}_{k-1}| - t_k^* \mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right| \\
 &\leq \frac{|\mathbf{g}_k^T (\mathbf{g}_k - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_{k-1} - t_k^* \mathbf{s}_{k-1})|}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{|\mathbf{g}_k^T (\mathbf{g}_k - \mathbf{g}_{k-1} + \mathbf{g}_{k-1} - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \mathbf{g}_{k-1} - t_k^* \mathbf{s}_{k-1})|}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &\leq \frac{\|\mathbf{g}_k\| \left(\|\mathbf{g}_k - \mathbf{g}_{k-1}\| + \left\| \mathbf{g}_{k-1} \left(1 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \right) \right\| + t_k^* \|\mathbf{s}_{k-1}\| \right)}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{\|\mathbf{g}_k\| \left(\|\mathbf{g}_k - \mathbf{g}_{k-1}\| + \left| 1 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} \right| \|\mathbf{g}_{k-1}\| + t_k^* \|\mathbf{s}_{k-1}\| \right)}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{\|\mathbf{g}_k\| (\|\mathbf{g}_k - \mathbf{g}_{k-1}\| + \|\mathbf{g}_{k-1}\| - \|\mathbf{g}_k\| + t_k^* \|\mathbf{s}_{k-1}\|)}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &\leq \frac{\|\mathbf{g}_k\| (\|\mathbf{g}_k - \mathbf{g}_{k-1}\| + \|\mathbf{g}_{k-1} - \mathbf{g}_k\| + t_k^* \|\mathbf{s}_{k-1}\|)}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{\|\mathbf{g}_k\| (2 \|\mathbf{g}_k - \mathbf{g}_{k-1}\| + t_k^* \|\mathbf{s}_{k-1}\|)}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &\leq \frac{\|\mathbf{g}_k\| (2L \|\mathbf{s}_{k-1}\| + t_k^* \|\mathbf{s}_{k-1}\|)}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{(2L + t_k^*) \|\mathbf{g}_k\| \|\mathbf{s}_{k-1}\|}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{(2L + t_k^*) \|\mathbf{g}_k\| \alpha_{k-1} \|\mathbf{d}_{k-1}\|}{\eta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{(2L + t_k^*) \|\mathbf{g}_k\|}{\eta \|\mathbf{d}_{k-1}\|}.
 \end{aligned} \tag{6.65}$$

Koristeći $t_k^* \in (0, 1)$, (6.64) i (6.65) u (6.62), imamo da je

$$\begin{aligned}
 \|\mathbf{d}_k\|^2 &\leq \|\mathbf{g}_k\|^2 + 2 \frac{\|\mathbf{g}_k\|^2}{\lambda} + \frac{(2L + t_k^*)^2 \|\mathbf{g}_k\|^2}{\eta^2 \|\mathbf{d}_{k-1}\|^2} \|\mathbf{d}_{k-1}\|^2 \\
 &= \left(1 + \frac{2}{\lambda} + \frac{(2L + t_k^*)^2}{\eta^2} \right) \|\mathbf{g}_k\|^2 \\
 &= \frac{(\lambda + 2)\eta^2 + \lambda(2L + t_k^*)^2}{\lambda\eta^2} \|\mathbf{g}_k\|^2.
 \end{aligned} \tag{6.66}$$

Zatim, podelivši obe strane nejednačine (6.66) sa $\|\mathbf{g}_k\|^4$ i koristeći (6.61), imamo da važi

$$\begin{aligned} \frac{\|\mathbf{d}_k\|^2}{\|\mathbf{g}_k\|^4} &\leq \frac{(\lambda + 2)\eta^2 + \lambda(2L + t_k^*)^2}{\lambda\eta^2} \cdot \frac{1}{c_1^2}, \text{ odnosno} \\ \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} &\geq \frac{\lambda\eta^2 \cdot c_1^2}{(\lambda + 2)\eta^2 + \lambda(2L + t_k^*)^2}. \end{aligned} \quad (6.67)$$

Nejednačine u (6.67) impliciraju

$$\sum_{k=0}^{\infty} \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \geq \sum_{k=0}^{\infty} \frac{\lambda\eta^2 \cdot c_1^2}{(\lambda + 2)\eta^2 + \lambda(2L + t_k^*)^2} = \infty. \quad (6.68)$$

Prema tome, $\|\mathbf{g}_k\| \geq c_1$ izaziva kontradikciju sa Lemom 6.4.2. Odavde sledi da (6.60) važi, čime je dokaz završen. $\square$

Globalnu konvergenciju predložene MSMDL metode potvrđuje Teorema 6.4.5.

Teorema 6.4.5. *Neka važe ograničenja iz pretpostavki (H_1) i (H_2) Stava 4.5.1. Ako je funkcija f uniformno konveksna, onda za niz $\{\mathbf{x}_k\}$ generisan pomoću MSMDL iteracija važi*

$$\liminf_{k \rightarrow \infty} \|\mathbf{g}_k\| = 0. \quad (6.69)$$

Dokaz. Pretpostavimo suprotno, tj. da (6.69) ne važi. Onda ovo inicira postojanje pozitivne konstante $c_1 > 0$ takve da

$$\|\mathbf{g}_k\| \geq c_1, \text{ za svako } k \geq 0. \quad (6.70)$$

Dalje, na osnovu pravca pretraživanja (5.1) za predloženi MSMDL metod važi

$$\begin{aligned} \|\mathbf{d}_k\| &= \left\| -\mathbf{g}_k + \beta_k^{\text{MSMDL}} \mathbf{d}_{k-1} \right\| \\ &\leq \|\mathbf{g}_k\| + |\beta_k^{\text{MSMDL}}| \|\mathbf{d}_{k-1}\| \\ &= \|\mathbf{g}_k\| + \left| \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^{\text{MSMDL}} \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right| \|\mathbf{d}_{k-1}\| \\ &= \|\mathbf{g}_k\| + \left| \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\alpha_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^{\text{MSMDL}} \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\alpha_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right| \|\alpha_{k-1} \mathbf{d}_{k-1}\| \\ &= \|\mathbf{g}_k\| + \left| \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} - t_k^{\text{MSMDL}} \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \right| \|\mathbf{s}_{k-1}\| \\ &\leq \|\mathbf{g}_k\| + \left(\frac{\|\mathbf{g}_k\| \|\mathbf{y}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} + |t_k^{\text{MSMDL}}| \frac{\|\mathbf{g}_k\| \|\mathbf{s}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \right) \|\mathbf{s}_{k-1}\| \\ &= \|\mathbf{g}_k\| + \frac{\|\mathbf{g}_k\| (\|\mathbf{y}_{k-1}\| + |t_k^{\text{MSMDL}}| \|\mathbf{s}_{k-1}\|)}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \|\mathbf{s}_{k-1}\|. \end{aligned}$$

Primenom (4.32) i (6.46) u prethodnoj nejednačini dobija se

$$\begin{aligned}
 \|\mathbf{d}_k\| &\leq \|\mathbf{g}_k\| + \frac{\|\mathbf{g}_k\| (L\|\mathbf{s}_{k-1}\| + |t_k^{\text{MSMDL}}| \|\mathbf{s}_{k-1}\|)}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \|\mathbf{s}_{k-1}\| \\
 &= \|\mathbf{g}_k\| + \frac{\|\mathbf{g}_k\| (L + |t_k^{\text{MSMDL}}|) \|\mathbf{s}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \|\mathbf{s}_{k-1}\| \\
 &\leq \|\mathbf{g}_k\| + \frac{\|\mathbf{g}_k\| (L + |t_k^{\text{MSMDL}}|) \|\mathbf{s}_{k-1}\|}{\eta \|\mathbf{s}_{k-1}\|^2} \|\mathbf{s}_{k-1}\| \\
 &= \left(1 + \frac{L + |t_k^{\text{MSMDL}}|}{\eta}\right) \|\mathbf{g}_k\|.
 \end{aligned} \tag{6.71}$$

Da bi kompletirali dokaz teoreme potrebno je dokazati da je $|t_k^{\text{MSMDL}}|$ ograničeno. U tom slučaju razlikujemo dva slučaja na osnovu definicije t_k^{MSMDL} u (6.20):

(i): Ako je $t_k^{\text{MSMDL}} = \theta \|\mathbf{y}_{k-1}\|^2 / \mathbf{s}_{k-1}^T \mathbf{y}_{k-1}$, onda imamo da je

$$|t_k^{\text{MSMDL}}| = \left| \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \right| \leq \theta \frac{L^2 \|\mathbf{s}_{k-1}\|^2}{\eta \|\mathbf{s}_{k-1}\|^2} = \frac{\theta L^2}{\eta}. \tag{6.72}$$

(ii): Ako je $t_k^{\text{MSMDL}} = \tau_k$, onda imamo da je

$$\begin{aligned}
 |t_k^{\text{MSMDL}}| &= \frac{|(\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1) \|\mathbf{g}_k\|^2 \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + \mathbf{g}_k^T \mathbf{y}_{k-1} \mathbf{g}_k^T \mathbf{s}_{k-1}|}{(\mathbf{g}_k^T \mathbf{s}_{k-1})^2} \\
 &\leq \frac{|\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1| \|\mathbf{g}_k\|^2 L \|\mathbf{s}_{k-1}\|^2 + \|\mathbf{g}_k\| \|\mathbf{y}_{k-1}\| \|\mathbf{g}_k\| \|\mathbf{s}_{k-1}\|}{\|\mathbf{g}_k\|^2 \|\mathbf{s}_{k-1}\|^2} \\
 &\leq \frac{|\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1| \|\mathbf{g}_k\|^2 L \|\mathbf{s}_{k-1}\|^2 + \|\mathbf{g}_k\|^2 L \|\mathbf{s}_{k-1}\|^2}{\|\mathbf{g}_k\|^2 \|\mathbf{s}_{k-1}\|^2} \\
 &= L |\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1| + L.
 \end{aligned} \tag{6.73}$$

Na osnovu Algoritma 2.2.3 i početne vrednosti za $\alpha_k = 1$, sledi da je $\alpha_k \in (0, 1]$. Činjenica $\alpha_k \in (0, 1]$ implicira da je $1 \leq \mathfrak{J}_k \leq 5/4$. Za više informacija pogledati analizu funkcije $\mathfrak{J}_k(\alpha_k)$ i njen grafik Slika 6.1. Dalje, zajedno sa (6.73) imamo da važi

$$\begin{aligned}
 |t_k^{\text{MSMDL}}| &\leq L |\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} - 1| + L \\
 &\leq L (\mathfrak{J}_k(\gamma_k^{\text{MSM}})^{-1} + 1) + L \\
 &\leq L \left(2 + \frac{5}{4} (\gamma_k^{\text{MSM}})^{-1}\right).
 \end{aligned} \tag{6.74}$$

Kako je γ_k aproksimacija Hesijana $\nabla^2 f(\mathbf{x}_k)$, nejednakost (4.34) implicira $m \leq \gamma_k \leq M$.

Sada iz (6.74) sledi

$$|t_k^{\text{MSMDL}}| \leq L \left(2 + \frac{5}{4m} \right). \quad (6.75)$$

Dalje, iz (6.72) i (6.75) imamo da važi

$$|t_k^{\text{MSMDL}}| \leq \max \left\{ \frac{\theta L^2}{\eta}, L \left(2 + \frac{5}{4m} \right) \right\} = T. \quad (6.76)$$

Sada, na osnovu (6.71) i (6.76) sledi

$$\|\mathbf{d}_k\| \leq \left(1 + \frac{L+T}{\eta} \right) \|\mathbf{g}_k\| = \frac{\eta + L + T}{\eta} \|\mathbf{g}_k\|. \quad (6.77)$$

Kvadriranjem obe strane u nejednačini (6.77) dobija se

$$\|\mathbf{d}_k\|^2 \leq \frac{(\eta + L + T)^2}{\eta^2} \|\mathbf{g}_k\|^2. \quad (6.78)$$

Zatim, deljenjem obe strane nejednakosti (6.78) sa $\|\mathbf{g}_k\|^4$ i koristeći (6.70), imamo da važi

$$\frac{\|\mathbf{d}_k\|^2}{\|\mathbf{g}_k\|^4} \leq \frac{(\eta + L + T)^2}{\eta^2} \cdot \frac{1}{c_1^2} \iff \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \geq \frac{\eta^2 \cdot c_1^2}{(\eta + L + T)^2}. \quad (6.79)$$

Nejednačine u (6.79) impliciraju

$$\sum_{k=0}^{\infty} \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \geq \sum_{k=0}^{\infty} \frac{\eta^2 \cdot c_1^2}{(\eta + L + T)^2} = \infty. \quad (6.80)$$

Prema tome, $\|\mathbf{g}_k\| \geq c_1$ izaziva kontradikciju sa Lemom 6.4.2. Odavde sledi da (6.69) važi, čime je dokaz završen. $\square$

Teorema 6.4.6. *Neka važe ograničenja iz pretpostavki (H_1) i (H_2) Stava 4.5.1. Ako je funkcija cilja f uniformno konveksna, onda za niz $\{\mathbf{x}_k\}$ generisan pomoću BB1DL iteracija važi (6.69).*

Dokaz. Dokaz Teoreme 6.4.6 je sličan dokazu Teoreme 6.4.5. Razlikuje se jedino u delu gde je potrebno dokazati da je parametar $|t_k^{\text{BB1DL}}|$ ograničen. U nastavku dokazujemo da je parametar $|t_k^{\text{BB1DL}}|$ ograničen. U tom slučaju razlikujemo dva slučaja na osnovu definicije t_k^{BB1DL} u (6.22):

(i): Ako je $t_k^{\text{BB1DL}} = \theta \|\mathbf{y}_{k-1}\|^2 / \mathbf{s}_{k-1}^T \mathbf{y}_{k-1}$, onda na osnovu (6.72) imamo da je

$$|t_k^{\text{BB1DL}}| \leq \frac{\theta L^2}{\eta}. \quad (6.81)$$

(ii): Ako je $t_k^{\text{BB1DL}} = \tau_k^{\text{BB1}}$, onda imamo da je

$$\begin{aligned}
 |t_k^{\text{BB1DL}}| &= \frac{|((\gamma_k^{\text{BB1}})^{-1} - 1) \|\mathbf{g}_k\|^2 \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + \mathbf{g}_k^T \mathbf{y}_{k-1} \mathbf{g}_k^T \mathbf{s}_{k-1}|}{(\mathbf{g}_k^T \mathbf{s}_{k-1})^2} \\
 &\leq \frac{|(\gamma_k^{\text{BB1}})^{-1} - 1| \|\mathbf{g}_k\|^2 L \|\mathbf{s}_{k-1}\|^2 + \|\mathbf{g}_k\| \|\mathbf{y}_{k-1}\| \|\mathbf{g}_k\| \|\mathbf{s}_{k-1}\|}{\|\mathbf{g}_k\|^2 \|\mathbf{s}_{k-1}\|^2} \\
 &\leq \frac{|(\gamma_k^{\text{BB1}})^{-1} - 1| \|\mathbf{g}_k\|^2 L \|\mathbf{s}_{k-1}\|^2 + \|\mathbf{g}_k\|^2 L \|\mathbf{s}_{k-1}\|^2}{\|\mathbf{g}_k\|^2 \|\mathbf{s}_{k-1}\|^2} \\
 &= L |(\gamma_k^{\text{BB1}})^{-1} - 1| + L.
 \end{aligned} \tag{6.82}$$

Sada iz (3.29) i (6.48), sledi da je $\gamma_k^{\text{BB1}} > 0$, tj. imamo da važi

$$\begin{aligned}
 |t_k^{\text{BB1DL}}| &\leq L |(\gamma_k^{\text{BB1}})^{-1} - 1| + L \\
 &\leq L((\gamma_k^{\text{BB1}})^{-1} + 1) + L \\
 &= L \left(\frac{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}{\mathbf{y}_{k-1}^T \mathbf{y}_{k-1}} \right)^{-1} + 2L \\
 &= L \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} + 2L \\
 &\leq L \frac{L^2 \|\mathbf{s}_{k-1}\|^2}{\eta \|\mathbf{s}_{k-1}\|^2} + 2L \\
 &\leq L \left(\frac{L^2}{\eta} + 2 \right).
 \end{aligned} \tag{6.83}$$

Pretposlednja nejednakost u (6.83) direktno sledi iz (4.32) i (6.46). Dalje, na osnovu (6.81) i (6.83) imamo da je

$$|t_k^{\text{BB1DL}}| \leq \max \left\{ \frac{\theta L^2}{\eta}, L \left(\frac{L^2}{\eta} + 2 \right) \right\} = T. \tag{6.84}$$

Time je dokazano da je parametar $|t_k^{\text{BB1DL}}|$ ograničen, čime je dokaz završen. $\square$

U nastavku odeljka dokazujemo linearnu konvergenciju MSMDL metode. Teorema 6.4.7 pokazuje linearnu konvergenciju MSMDL metode pod pretpostavkom da je funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i uniformno konveksna na $\mathbb{R}^n$. Da bismo dokazali linearnu konvergenciju MSMDL metode potrebna nam je Lema 6.4.5 koja daje donju granicu dužine koraka α_k . Dokaz Leme 6.4.5 je sličan dokazu u [54, Lema 4] ili dokazu u [96, Lema 4].

Lema 6.4.5. *Pretpostavimo da je funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i uniformno konveksna na $\mathbb{R}^n$ i da je niz $\{\mathbf{x}_k\}$ generisana MSMDL metodom sa linijskim pretraživanjem unazad. Tada postoji konstanta $\lambda > 0$ takva da*

$$\alpha_k \geq \lambda, \quad \forall k \geq 0. \quad (6.85)$$

Dokaz. Uslov linijskog pretraživanja unazad je zadat sledećom nejednačinom

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \leq f(\mathbf{x}_k) + \sigma \alpha_k \mathbf{g}_k^T \mathbf{d}_k. \quad (6.86)$$

Ako je $\alpha_k \neq 1$, onda $\rho^{-1} \alpha_k$ ne zadovoljava (6.86), tj.

$$f(\mathbf{x}_k + \rho^{-1} \alpha_k \mathbf{d}_k) - f(\mathbf{x}_k) > \sigma \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k. \quad (6.87)$$

Teorema o srednjoj vrednosti i (4.32) obezbeđuju postojanje $\xi_k \in [0, 1]$, tako da

$$\begin{aligned} f(\mathbf{x}_k + \rho^{-1} \alpha_k \mathbf{d}_k) - f(\mathbf{x}_k) &= \rho^{-1} \alpha_k (\mathbf{g}(\mathbf{x}_k + \xi_k \rho^{-1} \alpha_k \mathbf{d}_k))^T \mathbf{d}_k \\ &= \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k + \rho^{-1} \alpha_k (\mathbf{g}(\mathbf{x}_k + \xi_k \rho^{-1} \alpha_k \mathbf{d}_k) - \mathbf{g}_k)^T \mathbf{d}_k \\ &\leq \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k + L \rho^{-2} \alpha_k^2 \|\mathbf{d}_k\|^2. \end{aligned} \quad (6.88)$$

Iz (6.78) imamo da je

$$\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_k\|^2} \geq \frac{\eta^2}{(\eta + L + T)^2} = \lambda_1. \quad (6.89)$$

Dalje, sledeće nejednakosti važe na osnovu (6.88) i (6.87):

$$\begin{aligned} \sigma \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k &< f(\mathbf{x}_k + \rho^{-1} \alpha_k \mathbf{d}_k) - f(\mathbf{x}_k) \leq \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k + L \rho^{-2} \alpha_k^2 \|\mathbf{d}_k\|^2 \\ \sigma \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k - \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k &\leq L \rho^{-2} \alpha_k^2 \|\mathbf{d}_k\|^2 \\ (\sigma - 1) \rho^{-1} \alpha_k \mathbf{g}_k^T \mathbf{d}_k &\leq L \rho^{-2} \alpha_k^2 \|\mathbf{d}_k\|^2 \\ (\sigma - 1) \mathbf{g}_k^T \mathbf{d}_k &\leq L \rho^{-1} \alpha_k \|\mathbf{d}_k\|^2 \\ -(1 - \sigma) \mathbf{g}_k^T \mathbf{d}_k &\leq L \rho^{-1} \alpha_k \|\mathbf{d}_k\|^2. \end{aligned} \quad (6.90)$$

Iz nejednakosti (6.90), (6.56) i (6.89) sledi da važi

$$(1 - \sigma) c \|\mathbf{g}_k\|^2 \leq L \rho^{-1} \alpha_k \|\mathbf{d}_k\|^2,$$

odnosno oдавde imamo da je

$$\alpha_k \geq \frac{\rho(1 - \sigma)c}{L} \frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_k\|^2} \geq \frac{\rho(1 - \sigma)c}{L} \lambda_1. \quad (6.91)$$

Iz (6.91) sledi da važi tražena nejednakost (6.85) za $\lambda = \frac{\rho(1-\sigma)c}{L}\lambda_1$, čime je dokaz završen. $\square$

Teorema 6.4.7. *Neka je funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i uniformno konveksna na $\mathbb{R}^n$ i neka je $\mathbf{x}^*$ jedinstven minimum u (2.1). Tada postoje konstante $p > 0$ i $r \in (0, 1)$ takve da niz $\{\mathbf{x}_k\}$ generisana MSMDL metodom zadovoljava nejednakost*

$$\|\mathbf{x}_k - \mathbf{x}^*\| \leq p r^k. \quad (6.92)$$

Dokaz. Iz linijskog pretraživanja unazad, Leme 6.4.4 i Leme 6.4.5, sledi

$$\begin{aligned} f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*) &\leq f(\mathbf{x}_k) - f(\mathbf{x}^*) + \sigma \alpha_k \mathbf{g}_k^T \mathbf{d}_k \\ &\leq f(\mathbf{x}_k) - f(\mathbf{x}^*) - \sigma \alpha_k c \|\mathbf{g}_k\|^2 \\ &\leq f(\mathbf{x}_k) - f(\mathbf{x}^*) - \sigma \lambda c \|\mathbf{g}_k\|^2. \end{aligned} \quad (6.93)$$

Dalje, ako se u (4.36) uvedena smena $\mathbf{y} = \mathbf{x}^*$ i primeni teorema o srednjoj vrednosti kao i Cauchy–Schwartzova nejednakost, dobija se

$$m \|\mathbf{x}_k - \mathbf{x}^*\| \leq \|\mathbf{g}_k\| \leq M \|\mathbf{x}_k - \mathbf{x}^*\|, \quad \forall \mathbf{x}_k \in \mathbb{R}^n. \quad (6.94)$$

Koristeći levu nejednakost u (6.94), a zatim desnu nejednakost u (4.35), nejednakost (6.93) se transformiše u nejednakost

$$\begin{aligned} f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*) &\leq f(\mathbf{x}_k) - f(\mathbf{x}^*) - \sigma \lambda c m^2 \|\mathbf{x}_k - \mathbf{x}^*\|^2 \\ &\leq f(\mathbf{x}_k) - f(\mathbf{x}^*) - 2\sigma \lambda c \frac{m^2}{M} (f(\mathbf{x}_k) - f(\mathbf{x}^*)) \\ &= \left(1 - 2\sigma \lambda c \frac{m^2}{M}\right) (f(\mathbf{x}_k) - f(\mathbf{x}^*)). \end{aligned} \quad (6.95)$$

U nejednačini (6.95) uvedimo i analizirajmo zamenu $r = 1 - 2\sigma \lambda c \frac{m^2}{M}$. Jasno, na osnovu $0 < \sigma < 0.5$, $\lambda > 0$, $\frac{m^2}{M} > 0$ i $c = 1 - \frac{1}{4\theta} > 0$ za $\theta > \frac{1}{4}$ sledi da je $r < 1$. S druge strane kako je $\sigma < 0.5$, $\lambda \leq \alpha_k \leq 1$, $\frac{m^2}{M} < 1$ i $c < 1$ sledi da je $r > 1 - \lambda c > 0$, odnosno da za ovako definisano r važi $r \in (0, 1)$.

Dalje, iz (6.95) sledi da važi

$$f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*) \leq r (f(\mathbf{x}_k) - f(\mathbf{x}^*)) \leq \dots \leq r^{k+1} (f(\mathbf{x}_0) - f(\mathbf{x}^*)). \quad (6.96)$$

Kombinujući levu nejednakost u (4.35) sa (6.96), dobija se

$$\|\mathbf{x}_k - \mathbf{x}^*\| \leq \frac{2}{m} (f(\mathbf{x}_k) - f(\mathbf{x}^*)) \leq \frac{2}{m} (f(\mathbf{x}_0) - f(\mathbf{x}^*)) r^k \leq p r^k, \quad (6.97)$$

čime dokazujemo da nejednakost (6.92) važi za $p = \frac{2}{m} (f(\mathbf{x}_0) - f(\mathbf{x}^*)) > 0$, a samim tim

dokaz je završen. $\square$

Napomena 6.4.1. *Rezultat dobijen u Teoremi 6.4.7 može se direktno primeniti na BB1DL metod.*

6.5 Numerički rezultati i primena

U ovom odeljku ispitujemo efikasnost predloženih MLSCD, MMDL, EDL i MSMDL metoda. Odeljak se sastoji od četiri pododeljka, gde se u prva tri ispituje efikasnost predloženih metoda, a u četvrtom pododeljku je data primena MSMDL metode u rešavanju problema kontrole pokreta robota.

Svi kodovi korišćeni u numeričkim eksperimentima za testirane metode su napisani u Matlab R2017a programskom jeziku i izvršeni na laptop računaru sa Intel Core i3 2.0 GHz CPU-om, 8 GB RAM-a i Windows 10 operativnim sistemom. Tri važna kriterijuma (indikatora) se analiziraju u svakom pojedinačnom test slučaju: broj iterativnih koraka (broj iteracija), broj izračunavanja vrednosti funkcije u različitim tačkama (broj evaluacija funkcije) i procesorsko vreme, odnosno vreme izvršavanja algoritma (CPU vreme). Numerička testiranja su bazirana na test funkcijama iz [7], gde su mnoge test funkcije preuzete iz CUTER kolekcije [26].

Za svaki testirani algoritam korišćeni su isti kriterijumi zaustavljanja programa:

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{ i } \quad \frac{|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)|}{1 + |f(\mathbf{x}_k)|} \leq \delta,$$

gde je $\epsilon = 10^{-6}$ i $\delta = 10^{-16}$. Za ulazne parametre u Algoritmu 2.2.3 linijskog pretraživanja unazad za sve testirane algoritme korišćene su vrednosti $\sigma = 0.0001$ i $\beta = 0.8$. Što znači da prihvatamo malo smanjenje ciljne funkcije f predviđeno linearnom aproksimacijom u trenutnoj tački.

Profili performansi predloženi u [58] predstavljaju jedan od najčešće korišćenih alata za upoređivanje performansi algoritama za bezuslovnu optimizaciju na velikom skupu testova. Za upoređivanje koriste se numerički rezultati dobijene praćenjem broja iteracija, broja evaluacija funkcije i CPU vreme za svaki algoritam za svaku testiranu funkciju.

Metod Dolan-a i Moré-a u [58] funkcioniše na osnovu odnosa rezultata posmatrane performanse određenog algoritma u odnosu na najbolji rezultat posmatrane performanse svih algoritama. Da bismo približno procenili numeričke performanse različitih algoritama, usvajamo profile performansi koji su uveli Dolan i Moré [58] kao alat za procenu efikasnosti algoritama. U dijagramu profila performansi, gornja kriva odgovara algoritmu koji je rešio najviše problema u vremenu koje je unutar datog faktora najboljeg vremena. Leva strana profila daje procenat test zadataka za koje je algoritam najbrži, dok desna strana profila daje procenat test zadataka koji su uspešno rešeni ovim algoritmima. U suštini, desna strana dijagrama je mera za robusnost algoritma, dok je leva mera efikasnosti algoritma.

Profili performansi rangiraju algoritme uključene u skup $\mathcal{A}$ na skupu test problema $\mathcal{P}$. Količnik profila performansi $r_{p,a}$ je definisan za svaki problem $p \in \mathcal{P}$ i svaki algoritam $a \in \mathcal{A}$ formulom

$$r_{p,a} = \frac{m_{p,a}}{\min \{m_{p,a} : p \in \mathcal{P} \wedge a \in \mathcal{A}\}},$$

gde $m_{p,a}$ označava meru performansi (broj iterativnih koraka ili broj evaluacija finkcija ili CPU vreme) potrebnu za rešavanje problema p od strane algoritma a . Ukoliko posmatrani algoritam a ima najbolje performanse na problemu p posmatrajući sve algoritme iz skupa $\mathcal{A}$, onda je količnik profila performansi $r_{p,a} = 1$. Zatim se profil performansi (broj iterativnih koraka, broj evaluacija finkcija ili CPU vreme) za algoritam a definiše na sledeći način

$$\rho_a(\tau) = \frac{1}{n_p} \text{size} \{p \in \mathcal{P} : \log_2 r_{p,a} \leq \tau\},$$

gde je n_p ukupan broj svih testiranih problema. Algoritmi sa većom verovatnoćom $\rho_a(\tau)$ su poželjniji. Ako algoritam a_1 postigne bolje rezultate u poređenju sa algoritmom a_2 , onda se kriva $\rho_{a_1}(\tau)$ profila performansi koji generiše algoritam a_1 nalazi iznad odgovarajuće krive $\rho_{a_2}(\tau)$ profila performansi koji generiše algoritam a_2 . Funkcija $\rho_a : \mathbb{R} \rightarrow [0, 1]$ naziva se kumulativna funkcija raspodele količnika performansi $r_{p,a}$ i ona prikazuje performanse algoritma a , tj. predstavlja profil performansi algoritma a .

6.5.1 Numerički rezultati za MLSCD i MMDL metodu

U ovom odeljku predstavljamo numeričke rezultate dobijene testiranjem DHSDL, DLSDL, MMDL i MLSCD metoda. Numerička testiranja su bazirana na 33 test funkcija iz [7], gde su mnoge test funkcije preuzete iz CUTEr kolekcije [26]. Za svaku od testiranih funkcija urađeno je 10 numeričkih testova sa različitim izborom broja promenljivih: 100, 500, 1000, 2000, 3000, 5000, 7000, 8000, 10000 i 15000. Zbirni numerički rezultati za DHSDL, DLSDL, MMDL i MLSCD metod, testirani na 33 test funkcija velikih dimenzija, predstavljeni su u Tabelama 6.1, 6.2 i 6.3.

Slika 6.2a prikazuje profile performanse upoređenih metoda u odnosu na broj iteracija, dok Slika 6.2b prikazuje profile performanse posmatranih metoda u odnosu na broj evaluacija funkcije. Slika 6.3 prikazuje profile performansi posmatranih metoda u odnosu na CPU vreme. Kako gornja kriva odgovara metodi koja pokazuje najbolje performanse u odnosu na izabrani profil performansi, ovde je očigledno da MLSCD metod postiže najbolje rezultate.

Na Slikama 6.2 i 6.3 se može primetiti da DHSDL, DLSDL, MMDL i MLSCD metode uspešno rešavaju sve test probleme. Na Slici 6.2a MLSCD metod je najbolji u 82% testiranih problema u poređenju sa drugim metodama, dok na Slici 6.2b MLSCD metod je najbolji u 78% testiranih problema u poređenju sa drugim metodama. Kada su u pitanju profili performansi bazirani na CPU vremenu na Slici 6.3 može se uočiti da je MLSCD metod najbolji u 73% testiranih problema u poređenju sa ostalim metodama. Prema tome, Slike 6.2 i 6.3 pokazuju da MLSCD metod postiže najbolje rezultate u odnosu na sva tri posmatrana

Tabela 6.1: Zbirni numerički rezultati DHSDL, DLSDL, MMDL i MLSCD metoda za broj iterativnih koraka.

Test funkcija	Broj iteracija			
	<i>DHSDL</i>	<i>DLSDL</i>	<i>MMDL</i>	<i>MLSCD</i>
Extended Penalty	1215	1234	1205	1189
Perturbed Quadratic	270275	221969	259193	6776
Raydan 1	53186	49828	49015	3321
Raydan 2	390	390	56	56
Diagonal 1	29328	27623	32947	3548
Diagonal 3	57572	46091	52521	4477
Hager	3185	3205	2962	1614
Generalized Tridiagonal 1	622	625	622	1385
Extended Tridiagonal 1	986304	857	1250116	3939
Extended TET	3320	3416	2005	1042
Diagonal 4	8184	8046	7778	1439
Diagonal 5	25070	25070	40	40
Extended Himmelblau	1348	1348	1345	1614
Perturbed quadratic diagonal	1314838	832200	1162770	12650
Quadratic QF1	269831	231134	262582	8654
Extended quadratic penalty QP1	560	556	585	846
Extended quadratic penalty QP2	120793	115663	112874	2863
Quadratic QF2 function	290961	259641	280104	7875
Extended quadratic exponential EP1	506	506	501	501
Extended Tridiagonal 2	1334	1339	1095	748
ARWHEAD (CUTE)	37844	34583	37508	3159
Almost Perturbed Quadratic	273128	225250	262536	6792
LIARWHD (CUTE)	1491307	1382551	1441296	3498
ENGVAL1 (CUTE)	558	570	561	1045
Diagonal 6	482	486	56	56
COSINE (CUTE)	6420	13212	5802	2154
Generalized Quartic	1540	1593	1226	1163
Diagonal 7	5163	5163	539	539
Diagonal 8	16366	16366	577	577
Full Hessian FH3	2458	2455	2454	2456
Diagonal 9	221364	158244	200892	4654
HIMMELH (CUTE)	90	90	90	67
Extended Rosenbrock	50	50	50	40

indikatora.

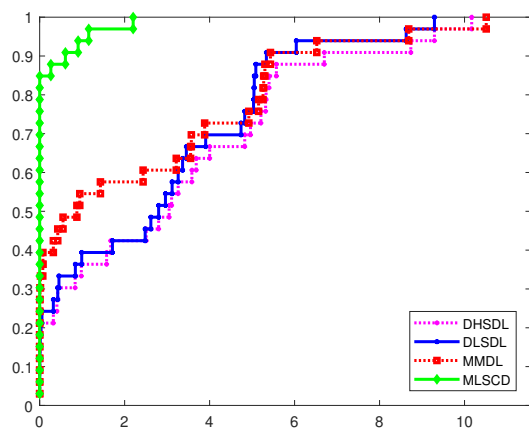
Tabela 6.4 sadrži prosečan broj iteracija i broj evaluacija funkcija, kao i prosečno CPU vreme za 330 numeričkih testova.

Na osnovu rezultata datih u Tabeli 6.4, očigledno je da MLSCD metod daje bolje nu-

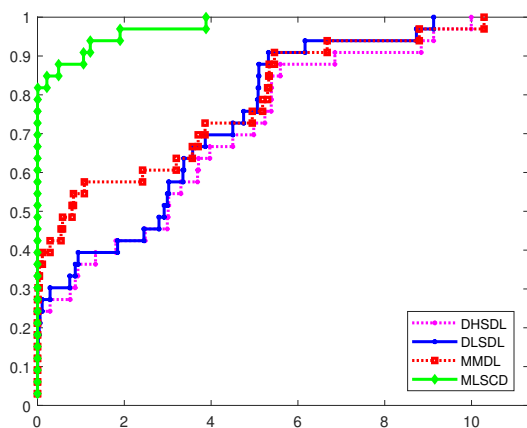
Tabela 6.2: Zbirni numerički rezultati DHSDL, DLSDL, MMDL i MLSCD metoda za broj evaluacija funkcija.

Test funkcija	Broj evaluacija funkcije			
	<i>DHSDL</i>	<i>DLSDL</i>	<i>MMDL</i>	<i>MLSCD</i>
Extended Penalty	44466	44600	44179	42846
Perturbed Quadratic	11355510	9214352	10909537	271818
Raydan 1	1435352	1334819	1329884	91573
Raydan 2	970	970	122	122
Diagonal 1	1100988	1032762	1252937	136604
Diagonal 3	2158912	1709338	1986158	167563
Hager	56222	56459	52392	29503
Generalized Tridiagonal 1	10290	10568	10387	23914
Extended Tridiagonal 1	5194192	5092	6369195	19043
Extended TET	33240	34180	20100	9510
Diagonal 4	158336	155132	151761	28344
Diagonal 5	50310	50310	90	90
Extended Himmelblau	25572	25572	25515	29640
Perturbed quadratic diagonal	54423738	33793696	48110702	471146
Quadratic QF1	10439311	8879307	10198882	330341
Extended quadratic penalty QP1	10551	10221	11047	14320
Extended quadratic penalty QP2	3443064	3283726	3243403	82215
Quadratic QF2	13106020	11629683	12644902	347236
Extended quadratic exponential EP1	13968	13968	13776	13776
Extended Tridiagonal 2	11261	11255	9185	6157
ARWHEAD (CUTE)	1670783	1521208	1661430	127659
Almost Perturbed Quadratic	11471846	9348191	11052096	273094
LIARWHD (CUTE)	71820981	66491445	69605196	156571
ENGVAL1 (CUTE)	8182	8815	8406	17065
Diagonal 6	1205	1262	122	122
COSINE (CUTE)	176032	485018	120982	69682
Generalized Quartic	18131	18004	15666	10756
Diagonal 7	26630	26630	3275	3275
Diagonal 8	82669	82669	3654	3654
Full Hessian FH3	86912	86866	86267	86714
Diagonal 9	9542988	6768095	8683251	197364
HIMMELH (CUTE)	190	190	190	2803
Extended Rosenbrock	110	110	110	90

meričke rezultate u poređenju sa DHSDL, DLSDL i MMDL metodama. Ovaj zaključak potvrđuju i profili performansi za broj iteracija, broj evaluacija funkcija i CPU vreme.

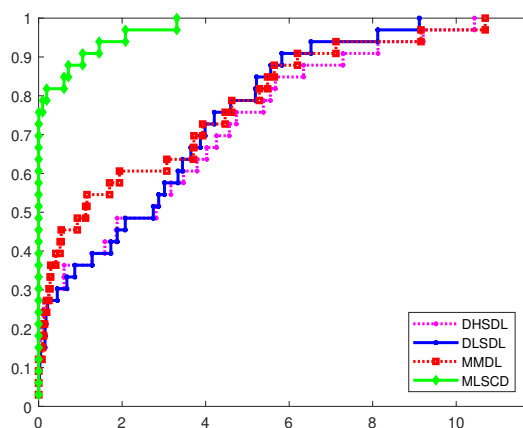


(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 6.2: Profili performansi za DHSDL, DLSDL, MMDL i MLSCD metod.



Slika 6.3: Profili performansi bazirani na CPU vremenu za DHSDL, DLSDL, MMDL i MLSCD metod.

6.5.2 Numerički rezultati za EDL metod

U ovom odeljku vršimo analizu i upoređivanje numeričkih rezultata dobijenih EDL metodom čija je implementacija zasnovana na Algoritmu 6.3.1, sa numeričkim rezultatima dobijenih iz četiri varijante MHSDL metode (5.12). Ove varijante su definisane sa $t \equiv t_{k3}$ iz (5.18), $t \equiv t_{k4}$ iz (5.19), $t \equiv t_{k5}$ iz (5.20), i $t \equiv t_{k6}$ iz (5.21) i označene kao MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metod. Dobijeni rezultati se ne upoređuju sa rezultatima za t_{k1} iz (5.15) i t_{k2} iz (5.17), jer su u [19] autori već pokazali da t_{k3} iz (5.18) i t_{k4} iz (5.19) postižu bolje numeričke performanse u poređenju sa t_{k1} iz (5.15) i t_{k2} iz (5.17).

Numerička testiranja se izvode korišćenjem 28 test funkcija iz [7], gde je veći deo test funkcija preuzet iz CUTER kolekcije u [26]. Sve metode koje se koriste u testiranju polaze

Tabela 6.3: Zbirni numerički rezultati DHSDL, DLSDL, MMDL i MLSCD metoda za CPU vreme u sekundama.

Test funkcija	CPU vreme (sec)			
	<i>DHSDL</i>	<i>DLSDL</i>	<i>MMDL</i>	<i>MLSCD</i>
Extended Penalty	11.469	11.844	12.141	12.266
Perturbed Quadratic	4703.016	4348.422	4617.125	92.375
Raydan 1	430.313	386.734	402.156	47.875
Raydan 2	0.656	0.922	0.313	0.219
Diagonal 1	1006.297	920.359	1186.688	90.953
Diagonal 3	2926.469	2296.703	2776.672	210.828
Hager	87.500	88.078	80.781	36.203
Generalized Tridiagonal 1	9.094	9.359	8.422	17.500
Extended Tridiagonal 1	14915.734	10.750	17806.109	45.438
Extended TET	20.766	20.797	12.313	5.641
Diagonal 4	19.422	17.781	17.406	5.359
Diagonal 5	83.000	82.625	0.359	0.297
Extended Himmelblau	3.672	3.750	4.000	6.016
Perturbed quadratic diagonal	21479.750	12615.844	19096.688	137.219
Quadratic QF1	4994.438	3891.266	4762.094	106.500
Extended quadratic penalty QP1	2.922	2.828	3.359	4.313
Extended quadratic penalty QP2	513.609	499.391	479.750	31.516
Quadratic QF2	4023.453	3586.750	3756.359	96.453
Extended quadratic exponential EP1	4.281	4.844	4.172	4.188
Extended Tridiagonal 2	3.031	3.172	2.641	1.984
ARWHEAD (CUTE)	791.922	720.609	733.859	29.734
Almost Perturbed Quadratic	5771.016	4021.641	5214.688	71.000
LIARWHD (CUTE)	24076.406	22565.406	23170.625	40.625
ENGVAL1 (CUTE)	2.438	2.703	2.391	6.531
Diagonal 6	1.031	1.031	0.172	0.141
COSINE (CUTE)	29.609	370.688	113.250	33.828
Generalized Quartic	4.813	5.750	6.000	3.156
Diagonal 7	17.922	17.094	2.813	2.547
Diagonal 8	50.344	48.484	3.000	2.625
Full Hessian FH3	20.938	20.391	21.500	20.094
Diagonal 9	2539.172	1570.594	2368.234	107.125
HIMMELH (CUTE)	0.594	0.609	0.547	5.422
Extended Rosenbrock	0.203	0.234	0.250	0.172

od iste početne tačke $\mathbf{x}_0$ za posmatranu test funkciju. Svaka test funkcija se testira 10 puta sa postepenim povećanjem broja promenljivih: 100, 500, 1000, 3000, 5000, 7000, 8000, 10000, 15000 i 20000.

Dodatni parametri koji su neophodni samo za MHSDL6 metodu definisani su na sledeći

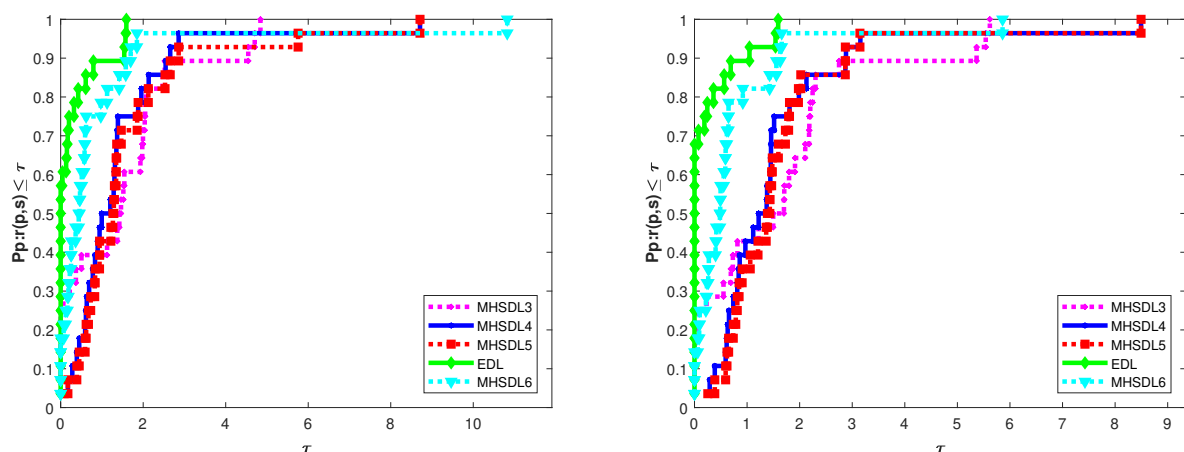
Tabela 6.4: Prosečni numerički rezultati za 33 test funkcija testiranih na 10 numeričkih testova.

Prosečne performanse	<i>DHSDL</i>	<i>DLSDL</i>	<i>MMDL</i>	<i>MLSCD</i>
Broj iteracija	166,533.091	111,253.152	164,663.727	2,750.818
Broj evaluacija funkcije	5,999,361.576	4,731,348.879	5,685,599.970	92,866.970
CPU vreme (sec)	2,683.191	1,762.044	2,626.269	38.671

način: $C = 1$, $v = 0.26$ i $r = r_k = v\|\mathbf{g}_{k-1}\|$.

Zbirni numerički rezultati za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metode, testirani na 28 test funkcija predstavljeni su u Tabelama 6.5–6.7. Tabele 6.5, 6.6 i 6.7 prikazuju numeričke rezultate koji odgovaraju za sva tri kriterijuma (broj iteracija, broj evaluacija funkcije i CPU vreme) za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metodu.

Slike 6.4 i 6.5 prikazuju profile performansi za numeričke rezultate date u Tabelama 6.5, 6.6 i 6.7. Slika 6.4a prikazuje profile performansi za broj iteracija generisani za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metodu. Na slici se može uočiti da su EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metode uspešno rešili sve test funkcije, pri čemu EDL metod postiže najbolje rezultate u 57.14% testiranih funkcija u poređenju sa: MHSDL3 (25.00%), MHSDL4 (0.00%), MHSDL5 (0.00%) i MHSDL6 (17.86%) metodoma. Na Slici 6.4a takođe može se uočiti da grafik EDL metode izilazi prvi na vrh, odnosno dostiže vrednost 1, što znači da EDL metod nadmašuje druge posmatrane metode u odnosu na broj iteracija.



(a) Bazirani na broju iteracija.

(b) Bazirani na broju evaluacija funkcije.

Slika 6.4: Profili performansi za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metod.

Slika 6.4b prikazuje profile performansi za broj evaluacija funkcija za EDL, MHSDL3,

Tabela 6.5: Zbirni numerički rezultati EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metoda za broj iterativnih koraka.

Test funkcija	Broj iteracija				
	<i>MHSDL3</i>	<i>MHSDL4</i>	<i>MHSDL5</i>	<i>EDL</i>	<i>MHSDL6</i>
Extended Penalty	1466	2243	2231	1259	1371
Perturbed Quadratic	1203710	754291	746557	305622	423037
Raydan 1	159055	110587	106586	55477	75154
Raydan 2	1636	441	441	70	209
Diagonal 1	116788	78844	73512	30978	20332
Diagonal 2	176983	270434	271595	515000	271295
Diagonal 3	150328	98647	104417	47155	37711
Hager	8666	5219	5157	3234	3625
Generalized Tridiagonal 1	1862	1471	1485	639	877
Extended TET	1357	5954	5915	4030	2664
Diagonal 4	30693	19589	19332	8040	12012
Diagonal 5	1721	25120	25120	60	216
Extended Himmelblau	1777	8023	7946	1376	3682
Perturbed quadratic diagonal	2940970	2115659	2027128	1136414	1352704
Quadratic QF1	1270802	799192	786032	309509	325415
Extended quadratic penalty QP1	770	594	575	560	543
Extended quadratic penalty QP2	399671	240530	245254	96620	137799
Extended quadratic exponential EP1	462	606	606	513	526
Extended Tridiagonal 2	3119	2176	2177	1132	1455
ARWHEAD (CUTE)	88824	69868	67413	40713	48669
ENGVAL1 (CUTE)	2323	1407	1415	552	820
INDEF (CUTE)	20	31	1080	23	36240
QUARTC (CUTE)	173913	262291	262291	524299	262181
Diagonal 6	1824	508	508	70	227
Generalized Quartic	1208	1403	2846	1265	1154
Diagonal 7	3217	655	655	653	580
Diagonal 8	511	698	698	686	596
Full Hessian FH3	1456	5353	5350	2523	3176

MHSDL4, MHSDL5 i MHSDL6 metodu. Na slici se može uočiti da su EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metode uspešno rešile sve test probleme, dok EDL metod postiže najbolje rezultate u 67.86% test funkcija u poređenju sa: MHSDL3 (17.86%), MHSDL4 (0.00%), MHSDL5 (0.00%) i MHSDL6 (14.28%) metodama. Na Slici 6.4b primećuje se da grafik EDL metode prvi izilazi na vrh, tj. dostiže vrednost 1, što potvrđuje da je EDL metod najefikasniji u pogledu broja evaluacija funkcija.

Slika 6.5 sadrži grafike profila performansi koji odgovaraju CPU vremenu za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metodu. Očigledno je da su EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metode uspešno rešile sve test probleme. Dalja analiza

Tabela 6.6: Zbirni numerički rezultati EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metoda za broj evaluacija funkcija.

Test funkcija	Broj evaluacija funkcije				
	<i>MHSDL3</i>	<i>MHSDL4</i>	<i>MHSDL5</i>	<i>EDL</i>	<i>MHSDL6</i>
Extended Penalty	54876	73764	73429	46820	49791
Perturbed Quadratic	56691737	34287604	33885701	13168688	18486375
Raydan 1	5066739	3364983	3236335	1551846	2170553
Raydan 2	6554	1162	1162	159	428
Diagonal 1	5004640	3256274	3022015	1200086	744278
Diagonal 2	353976	540878	543200	1030010	542600
Diagonal 3	6339146	3998904	4229565	1798032	1400076
Hager	192474	107413	106534	59187	69735
Generalized Tridiagonal 1	37429	27860	28138	10760	15177
Extended TET	19546	77422	76925	40340	29334
Diagonal 4	713120	425023	418666	155027	242443
Diagonal 5	6874	50460	50460	140	442
Extended Himmelblau	45972	192362	190524	26104	80854
Perturbed quadratic diagonal	135901222	94177165	90238441	48147512	57702654
Quadratic QF1	55972697	33836473	33243711	12316721	12853424
Extended quadratic penalty QP1	17016	12882	12565	11116	10544
Extended quadratic penalty QP2	13015888	7454686	7584960	2743358	4030601
Extended quadratic exponential EP1	14914	18463	18463	14132	15133
Extended Tridiagonal 2	36450	22564	22379	9687	12920
ARWHEAD (CUTE)	4296028	3305257	3182138	1846606	2230650
ENGVAL1 (CUTE)	40462	22432	22898	8209	12858
INDEF (CUTE)	1808	2182	5995	2060	104962
QUARTC (CUTE)	347926	524662	524662	1048648	524422
Diagonal 6	7394	1416	1408	159	468
Generalized Quartic	14364	21842	48770	16695	14103
Diagonal 7	6454	6838	6838	3891	4521
Diagonal 8	6098	6938	6938	4161	5494
Full Hessian FH3	60792	212799	212701	89890	114962

pokazuje da je EDL metod najbolji u 67.86% test slučajeva u poređenju sa: MHSDL3 (17.86%), MHSDL4 (0.00%), MHSDL5 (0.00%) i MHSDL6 (14.28%) metodama. Na Slici 6.5 se vidi da grafik EDL metode prvi dostiže vrednost 1, što ukazuje na njegovu superiornost u poređenju sa drugim metodama u odnosu na CPU vreme.

Na osnovu prethodne analize numeričkih rezultata prikazanih u Tabelama 6.5–6.7 i na Slikama 6.4 i 6.5 može se zaključiti da EDL metod postiže izuzetne rezultate u pogledu sva tri kriterijuma (broj iteracija, broj evaluacija funkcije i CPU vreme).

Numerička poređenja na opsežnim neograničenim optimizacijskim test funkcijama različitih struktura i složenosti potvrđuju efikasnost EDL algoritma i njegovu superiornost u

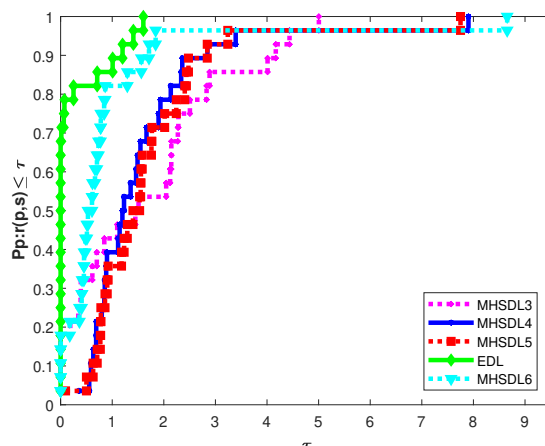
Tabela 6.7: Zbirni numerički rezultati EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metoda za CPU vreme u sekundama.

Test funkcija	CPU vreme (sec)				
	<i>MHSDL3</i>	<i>MHSDL4</i>	<i>MHSDL5</i>	<i>EDL</i>	<i>MHSDL6</i>
Extended Penalty	29.75	34.11	31.42	18.30	24.27
Perturbed Quadratic	40532.66	24358.20	24947.84	8335.80	13225.80
Raydan 1	3054.67	1904.48	1692.06	690.91	1184.86
Raydan 2	6.77	1.58	1.66	0.31	0.77
Diagonal 1	7834.03	5106.41	4592.28	1476.89	486.09
Diagonal 2	885.13	1428.05	1447.02	2352.11	1513.50
Diagonal 3	13614.27	8416.77	9064.30	3132.02	1916.30
Hager	586.63	325.75	333.41	142.06	198.13
Generalized Tridiagonal 1	66.14	35.59	34.42	15.19	21.63
Extended TET	20.50	78.34	82.94	41.23	31.45
Diagonal 4	134.53	77.86	87.88	30.41	55.34
Diagonal 5	18.06	134.73	121.09	0.56	1.84
Extended Himmelblau	11.13	44.47	44.36	6.19	18.30
Perturbed quadratic diagonal	91655.55	58226.16	60920.06	32179.38	36383.83
Quadratic QF1	62610.50	31552.48	28679.91	8832.11	8465.34
Extended quadratic penalty QP1	7.56	7.25	6.98	4.98	4.94
Extended quadratic penalty QP2	3814.16	2128.86	2288.55	671.52	1204.72
Extended quadratic exponential EP1	9.11	10.23	8.55	8.00	8.02
Extended Tridiagonal 2	11.13	8.83	6.95	4.08	5.25
ARWHEAD (CUTE)	2709.42	2336.92	2369.28	1266.80	1689.80
ENGVAL1 (CUTE)	19.47	11.33	11.81	4.03	6.70
INDEF (CUTE)	2.44	2.89	10.70	1.92	774.34
QUARTC (CUTE)	3106.56	4818.58	4808.70	7138.72	4735.39
Diagonal 6	6.75	1.92	2.03	0.38	1.34
Generalized Quartic	7.16	11.53	21.05	7.53	9.78
Diagonal 7	5.98	8.20	8.28	4.56	6.25
Diagonal 8	6.17	8.20	8.08	4.72	7.69
Full Hessian FH3	30.08	66.45	79.48	35.77	43.42

odnosu na ranije poznate Dai–Liao varijante konjugovano gradijentnih metoda, kao što su MHSDL3, MHSDL4, MHSDL5 i MHSDL6. Tokom testiranja pratili smo broj iteracija, broj evaluacija funkcija i performanse raspona procesorskog vremena za svaku funkciju i svaku metodu. Analiza dobijenih profila performansi pokazala je da je EDL metod najefikasniji.

6.5.3 Numerički rezultati za MSMDL metod

U ovom odeljku ćemo dokazati numeričku efikasnost MSMDL i BB1DL metoda čija je implementacija zasnovana na Algoritmu 6.3.2. U tom cilju uradićemo dva upoređivanja sa



Slika 6.5: Profili performansi bazirani na CPU vremenu za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metod.

standardnim test funkcijama i datim početnim tačkama iz [7, 26], pri čemu se svaka test funkcija testira 10 puta sa postepenim povećanjem broja promenljivih: 100, 500, 1000, 3000, 5000, 7000, 8000, 10000, 15000 i 20000.

Prvo upoređivanje je između CG-DESCENT [73], M1 [19], DK [41] i MSMDL metoda, a drugo između BB1DL, MSMDL i dve nedavno razvijene Dai–Liao konjugovano gradijentne metode (EDL [87] i MDL [107]). Metode koje učestvuju u prvom i u drugom upoređivanju su predstavljene u Odeljcima 5.2 i 6.3.1.

Specifični parametri koji se koriste samo u MDL i MSMDL metodama su definisani na sledeći način:

- u MDL metodi: $\theta = 0.26$, $C = 1$ i $r = r_k = \theta \|\mathbf{g}_{k-1}\|$;
- u MSMDL metodi: $\theta = 0.26$.

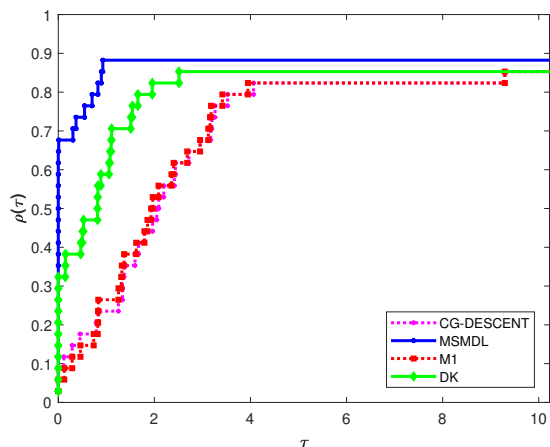
Pored standardnih kriterijuma zaustavljanja algoritama koji su dati na početku ovog odeljka, ovde je dodat još jedan kriterijum, a to je da se algoritam zaustavlja nakon 50000 iteracija. Simbol “*” u datim numeričkim rezultatima u narednim tabelama označava da metod nije uspeo da dostigne traženu tačnost nakon 50000 iteracija za jednu ili više testiranih dimenzija posmatrane test funkcije.

Zbirni numerički rezultati za prvo testiranje između MSMDL, CG-DESCENT, M1 i DK metoda dobijeni su testiranjem 34 test funkcija i predstavljani u Tabeli 6.8. Ova tabela uključuje numeričke rezultate dobijene praćenjem broj iterativnih koraka, broj evaluacija funkcije i CPU vremena u MSMDL, CG-DESCENT, M1 i DK metodama.

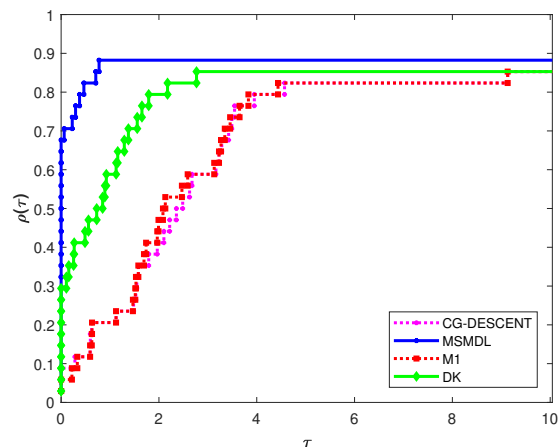
Profili performansi predloženi u [58] se primenjuju za upoređivanje dobijenih numeričkih podataka za broj iterativnih koraka, broj evaluacija funkcije i CPU vreme generisani testiranim algoritmima. Leva strana svakog profila performansi na Slikama 6.6-6.9 označava procenat test problema u kojima je posmatrana metoda najbolja među testiranim metodama, dok desna strana daje procenat test problema koji je uspešno rešen od strane posmatrane metode.

Tabela 6.8: Zbirni numerički rezultati MSMDL, CG-DESCENT, M1 i DK metoda za broj iterativnih koraka (NI), broj evaluacija testirane funkcije (NFE) i CPU vreme u sekundama (CPUts).

Test funkcija	CG-DESCENT		MSMDL		M1		DK	
	NI/NFE/CPUts		NI/NFE/CPUts		NI/NFE/CPUts		NI/NFE/CPUts	
Extended Penalty	2205/72128/21.656		1575/49766/18.766		2264/73854/23.219		1275/47664/18.375	
Perturbed Quadratic	*/ */ *		*/ */ *		*/ */ *		*/ */ *	
Raydan 1	126771/3900575/1572.109		30614/837631/408.953		119648/3667586/1457.172		53905/1508669/649.625	
Raydan 2	441/1162/1.219		57/124/0.313		441/1162/1.313		57/124/0.25	
Diagonal 1	85038/3530672/3958.094		21812/822865/1005		79024/3271384/3613.234		31406/1212581/1472.969	
Diagonal 2	*/ */ *		*/ */ *		*/ */ *		*/ */ *	
Diagonal 3	109803/4480102/6578.172		34503/1294525/2266.453		106160/4317169/6554.422		47699/1816310/2718.922	
Hager	6093/128794/299.625		1430/22994/52.063		6074/127372/284.891		3087/56298/117.359	
Generalized Tridiagonal 1	1525/29126/30.094		1124/17065/19.25		1499/28511/29.813		604/9965/11.141	
Extended Tridiagonal 1	*/ */ *		1053/3292/12.797		*/ */ *		*/ */ *	
Extended TET	5818/75694/55.813		1090/7769/7.25		5759/74907/55.203		2014/20160/17.406	
Diagonal 4	23674/525926/79.516		2058/34012/5.906		21963/483613/73.734		7986/153701/25.313	
Diagonal 5	25121/50463/105.547		40/90/0.438		25120/50460/104.656		40/90/0.297	
Extended Himmelblau	8028/192482/33.438		1220/21461/4.578		7837/187928/31.531		1350/25610/5.453	
Extended quadratic penalty QP1	595/13170/5.063		894/13296/5.406		598/13631/4.969		549/10835/4.313	
Extended quadratic exponential EPI	606/18463/7.188		499/12243/4.938		606/18463/7.063		496/13643/5.203	
ARWHEAD (CUTE)	82994/3970405/2213.172		18117/774822/474.234		69143/3280771/1753.766		38236/1728522/1018.219	
Almost Perturbed Quadratic	*/ */ *		*/ */ *		*/ */ *		*/ */ *	
ENGVAL1 (CUTE)	1436/23198/8.781		1008/13576/6.484		1423/22977/8.75		569/8306/3.656	
QUARTC (CUTE)	262291/524662/4001.109		*/ */ *		262291/524662/3917.75		*/ */ *	
Diagonal 6	513/1393/1.578		57/124/0.281		509/1555/1.563		57/124/0.297	
Generalized Quartic	2517/41813/17.469		832/6769/2.984		1470/21907/10.641		1473/19908/8.891	
Diagonal 7	655/6838/5.922		189/1750/2.594		655/6838/5.844		538/3321/3.766	
Diagonal 8	698/6926/6.063		270/2318/2.844		698/6926/6.453		579/3837/4.016	
Full Hessian FH3	5350/212795/54.609		2249/73823/25.156		5361/213236/55.313		2494/88852/25.203	
Diagonal 9	*/ */ *		66739/2712392/1231.156		*/ */ *		138771/5914443/1726.516	
Extended Rosenbrock	50/110/0.313		50/110/0.328		50/110/0.344		50/110/0.328	
Extended BDI (Block Diagonal)	1101/15879/7.703		1175/10059/5.188		1101/15870/6.641		618/7286/3.406	
Extended Maratos	93270/2927642/440.75		9716/250596/40.563		88036/2749653/422.266		30626/866995/135.375	
NONDQUAR (CUTE)	24/2324/9.516		31/2721/10.156		40/2708/10.594		34/2503/9.984	
DQDRITC (CUTE)	38363/1087562/306.25		7082/170118/49.594		36248/1022842/291.016		12486/316078/96.109	
Extended Freudenstein and Roth	291520/11172996/2176.25		31821/1040192/213.016		277174/10582041/2018.797		92706/3270158/656.016	
Extended Beale	31457/590812/1474.203		1880/24895/65.016		29002/538469/1354.422		10725/169640/427.641	
EDENSCH (CUTE)	984/15904/54.594		1050/13383/55.078		990/15918/56.281		719/10307/31.75	

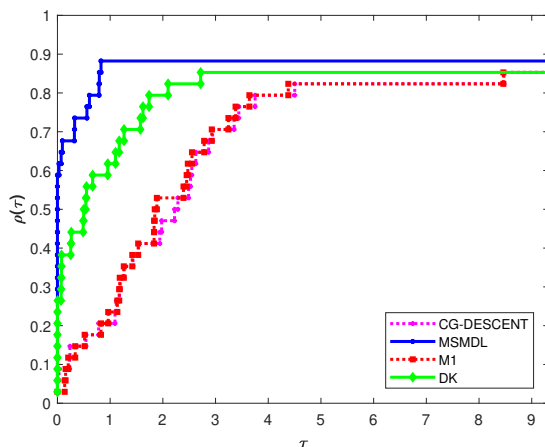


(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 6.6: Profili performansi za MSMDL, M1, CG-DESCENT i DK metod.



Slika 6.7: Profili performansi bazirani na CPU vremenu za MSMDL, M1, CG-DESCENT i DK metod.

Na Slici 6.6 upoređujemo profile performansi za broj iterativnih koraka i broj evaluacija funkcija za CG-DESCENT, MSMDL, M1 i DK metod na osnovu numeričkih vrednosti datih u Tabeli 6.8. Detaljnom analizom numeričkih podataka dolazimo do sledećeg zaključka da MSMDL metod rešava 64.71% testiranih problema sa najmanjim brojem iterativnih koraka u poređenju sa CG-DESCENT (8.82%), M1 (5.88%) i DK (32.35%) metodom. Na Slici 6.6a, očigledno je da grafik MSMDL metode prvi dostiže najviši nivo, što ukazuje da MSMDL nadmašuje druge posmatrane metode u odnosu na kriterijum broj iterativnih koraka. Slika 6.6b pokazuje da je MSMDL metod efikasniji od CG-DESCENT, M1 i DK metode, u odnosu na broj evaluacija funkcija, jer rešava 67.65% test problema sa najmanjim brojem evaluacija funkcija u poređenju sa CG-DESCENT (8.82%), M1 (5.88%) i DK (29.41%) metodom. Na

Slici 6.6b vidi se da grafik MSMDL metode prvi dostiže najviši nivo, tako da to ukazuje na činjenicu da je MSMDL metod efikasniji u odnosu na druge posmatrane metode u pogledu broja evaluacija funkcije.

Slika 6.7 prikazuje profile performansi CG-DESCENT, MSMDL, M1 i DK metode zasnovane na CPU vremena data u Tabeli 6.8. MSMDL metod rešava 58.82% testiranih problema za najmanje CPU vreme u poređenju sa CG-DESCENT (5.88%), M1 (2.94%) i DK (23.53%) metodama. Na Slici 6.7 vidi se da grafik MSMDL metode dostiže najviši nivo, što potvrđuje njegovu dominaciju nad ostalim metodama u pogledu CPU vremena.

MSMDL metod nije uspešno rešio 4 (11.76%) test funkcije od svih test funkcija u Tabeli 6.8, dok svaki od CG-DESCENT, M1 i DK metoda nije uspešno rešio po 5 (14.71%) test funkcija. Detaljna analiza numeričkih rezultata za svaku metodu je data u Tabeli 6.9.

Tabela 6.9: Statistika neuspešno rešenih test funkcija (test problema) po svakom metodu.

Metod	Test funkcija $\sum = 34$	Procenat (%)	Test problem $\sum = 340$	Procenat (%)
CG-DESCENT	5	14.71	26	7.65
MSMDL	4	11.76	10	2.94
M1	5	14.71	26	7.65
DK	5	14.71	22	6.47

Od ukupno 340 test problema, MSMDL je u stanju da reši najveći broj test problema (97,06% svih testiranih problema), dok su M1 i CG-DESCENT rešili po 92,35%, a DK 93,53% svih testiranih problema.

Na osnovu numeričkih podataka datih u Tabelama 6.8 i 6.9 i profila performansi datih na Slikama 6.6 i 6.7, može se zaključiti da MSMDL metod postiže najbolje rezultate u poređenju sa CG-DESCENT, M1 i DK metodama po tri osnovna kriterijuma: broj iterativnih koraka, broj evaluacija funkcije i CPU vreme.

Pored standardnih analiza numeričkih rezultata, izvršili smo dodatne analize za MSMDL metodu. Cilj dodatnih analiza je praćenje korišćenja nametnute vrednosti $\gamma_k = 1$ u iteracijama za svaku od testiranih funkcija. Takođe, pratimo i broj dodela $t_k^{\text{MSMDL}} = \tau_k$ u (6.20) za svaku pojedinačnu test funkciju. Treći analizirani parametar je maksimalna vrednost za $\frac{g_k^T d_k}{\|g_k\|^2}$ dobijena tokom testiranja. Rezultati dodatnih testiranja su dati u Tabeli 6.10.

Ukupan broj dodeljenih vrednosti $\gamma_k = 1$ (odnosno $t_k^{\text{MSMDL}} = \tau_k$) u Tabeli 6.10 biće označen sa $\angle_{\gamma_k=1}$ (odnosno $\angle_{t_k=\tau_k}$). Dalje, sa μ biće označena vrednost $\mu = \max \left\{ \frac{g_k^T d_k}{\|g_k\|^2} \right\}$ u Tabeli 6.10 za svaku testiranu funkciju. Ukupan zbir pojedinačnih vrednosti broja iterativnih koraka (NI), $\angle_{\gamma_k=1}$ i $\angle_{t_k=\tau_k}$ u testiranim funkcijama biće označen sa $\sum \text{NI}$, $\sum \angle_{\gamma_k=1}$ i $\sum \angle_{t_k=\tau_k}$, respektivno. Ukupan zbir svih iterativnih koraka u svim test funkcijama je $\sum \text{NI} = 240235$, što dalje implicira $\sum \angle_{\gamma_k=1} = 117966 = 0.491044 \sum \text{NI}$ i $\sum \angle_{t_k=\tau_k} = 118352 = 0.492651 \sum \text{NI}$. Iz ovoga sledi da je $\sum \angle_{\gamma_k=1} \approx \sum \angle_{t_k=\tau_k} \approx 0.5 \sum \text{NI}$, odnosno da je zastupljenost $\gamma_k = 1$ (odnosno $t_k^{\text{MSMDL}} = \tau_k$) u iterativnim koracima oko 50%.

U narednim numeričkim eksperimentima upoređujemo MSMDL i BB1DL metode sa

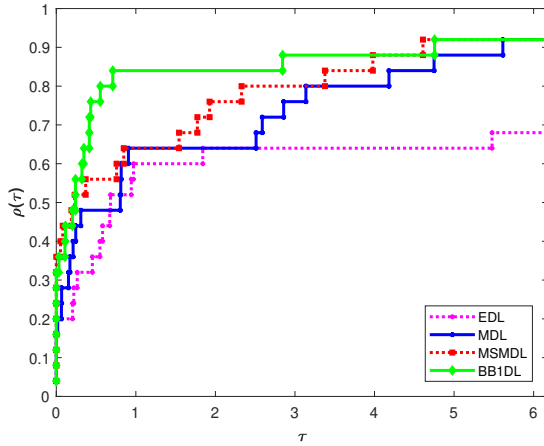
EDL i MDL metodama.

Tabela 6.10: Zbirni rezultati analize ponašanja parametara u MSMDL metodi.

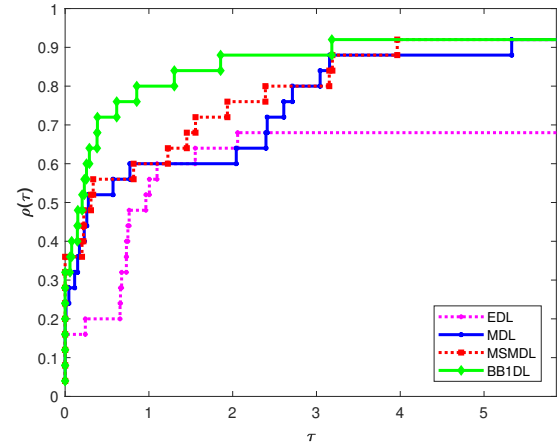
Test funkcija	Broj iterativnih koraka (NI)	$\angle_{\gamma_k=1}$	$\angle_{t_k=\tau_k}$	μ
Extended Penalty	1575	758	757	-0.2533
Raydan 1	30614	14981	14983	-0.2565
Raydan 2	57	57	57	-1
Diagonal 1	21812	10876	10876	-0.2608
Diagonal 3	34503	17203	17203	-0.2711
Hager	1430	606	607	-0.2600
Generalized Tridiagonal 1	1124	477	477	-0.2600
Extended Tridiagonal 1	1053	463	444	-0.2600
Extended TET	1090	330	415	-0.2201
Diagonal 4	2058	925	925	-0.2557
Diagonal 5	40	40	34	-1
Extended Himmelblau	1220	555	555	-0.2619
Extended quadratic penalty QP1	894	378	383	-0.2465
Extended quadratic exponential EP1	499	247	247	-0.26
ARWHEAD (CUTE)	18117	9074	9059	-0.2494
ENGVAL1 (CUTE)	1008	391	391	-0.2600
Diagonal 6	57	57	57	-1
Generalized Quartic	832	246	351	-0.2600
Diagonal 7	189	73	179	-0.26
Diagonal 8	270	109	263	-0.26
Full Hessian FH3	2249	1139	1143	-0.26
Diagonal 9	66739	33345	33345	-0.2617
Extended Rosenbrock	50	0	10	-0.26
Extended BD1 (Block Diagonal)	1175	411	437	-0.2403
Extended Maratos	9716	4760	4751	-0.1634
NONDQUAR (CUTE)	31	15	15	-0.2719
DQDRTIC (CUTE)	7082	3452	3452	-0.2600
Extended Freudenstein and Roth	31821	15851	15851	-0.2612
Extended Beale	1880	758	698	-0.1753
EDENSCH (CUTE)	1050	389	387	-0.2600

Upoređenja performansi MSMDL i BB1DL metode sa EDL i MDL metodama prikazana su na Slikama 6.8 i 6.9. Slika 6.8a upoređuje posmatrane metode u odnosu na broj iterativnih koraka, a Slika 6.8b u odnosu na broj evaluacija funkcija. Grafici profila performansi koji se odnose na CPU vreme za EDL, MDL, MSMDL i BB1DL metode su dati na Slici 6.9.

Slike 6.8a i 6.8b pokazuju da su MSMDL i BB1DL metode postigle nešto bolje rezultate u odnosu na EDL i MDL metode u pogledu broja iterativnih koraka i broja evaluacija funkcija, što potvrđuju grafici njihovih profila performansi.

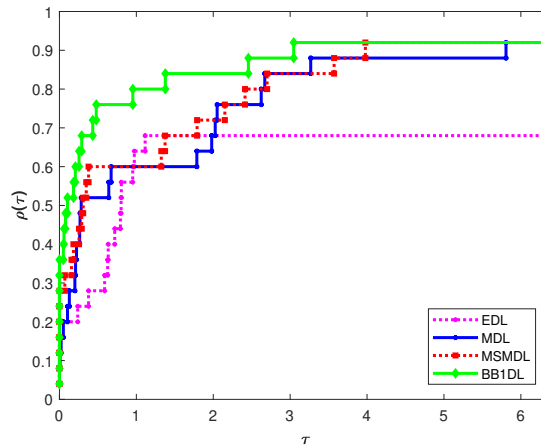


(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije.

Slika 6.8: Profili performansi za EDL, MDL, MSMDL i BB1DL metod.



Slika 6.9: Profili performansi bazirani na CPU vremenu za EDL, MDL, MSMDL i BB1DL metod.

Zbirni numerički rezultati za drugi eksperiment između EDL, MDL, MSMDL i BB1DL metoda su dobijeni testiranjem 25 test funkcija i dati u Tabeli 6.11. Tabela 6.11 sadrži numeričke podatke dobijene praćenjem tri standardna kriterijuma: broj iterativnih koraka, broj evaluacija testirane funkcije i CPU vreme u sekundama za EDL, MDL, MSMDL i BB1DL metodu.

Numerički rezultati u Tabeli 6.11 pokazuju da MSMDL metod rešava oko 36%, dok BB1DL metod uspešno rešava 32% test problema sa najmanjim brojem iterativnih koraka i najmanjim brojem evaluacija testiranih funkcija.

Profili performansi bazirani na CPU vremenu za MSMDL i BB1DL metod na Slici 6.9 pokazuju bolje performanse u poređenju sa profilima performansi za EDL i MDL metod. U numeričkim rezultatima datim u Tabeli 6.11 uočili smo da MSMDL metod rešava oko 28%, dok BB1DL metod uspešno rešava 36% test problema sa minimalnim CPU vremenom.

Tabela 6.11: Zbirni numerički rezultati EDL, MDL, MSMDL i BB1DL metoda za broj iterativnih koraka (NI), broj evaluacija testirane funkcije (NFE) i CPU vreme u sekundama (CPUts).

Test funkcija	EDL		MDL		MSMDL		BB1DL	
	NI/NFE/CPUts		NI/NFE/CPUts		NI/NFE/CPUts		NI/NFE/CPUts	
Extended Penalty	2304/82602/32.719		1866/55194/20.516		1575/49766/18.766		1862/55274/19.813	
Raydan 2	*/ */		325/660/1.078		57/124/0.313		76/162/0.422	
Hager	1940/33206/85.703		1274/20621/45.203		1430/22994/52.063		1218/19995/43.641	
Generalized Tridiagonal 1	2161/33285/37.266		1250/18463/22.453		1124/17065/19.25		1329/19648/22.969	
Extended Tridiagonal 1	308/4129/11.063		5590/11602/32.516		1053/3292/12.797		452/1407/5.109	
Diagonal 5	*/ */		290/590/1.813		40/90/0.438		50/110/0.453	
Extended Himmelblau	50/2413/0.859		1344/21510/5.297		1220/21461/4.578		1350/21921/4.719	
Extended quadratic penalty QP1	1157/18043/7.359		845/10736/5.125		894/13296/5.406		865/11335/4.766	
Extended quadratic exponential EP1	21431/43829/6.25		481/10518/4.391		499/12243/4.938		481/10518/3.797	
ENGVAL1 (CUTE)	1975/27260/10.063		1167/16242/7.891		1008/13576/6.484		1160/16202/7.359	
Diagonal 6	*/ */		343/757/1.109		57/124/0.281		76/190/0.344	
DIXON3DQ (CUTE)	*/ */		*/ */		*/ */		*/ */	
BIGGSB1 (CUTE)	*/ */		*/ */		*/ */		*/ */	
Generalized Quartic	959/10662/2.344		1459/10061/3.656		832/6769/2.984		1361/7935/3.266	
Diagonal 7	*/ */		110/457/0.75		189/1750/2.594		105/1128/1.453	
Diagonal 8	*/ */		125/1338/1.781		270/2318/2.844		71/254/0.438	
Full Hessian FH3	*/ */		1330/41888/20.891		2249/73823/25.156		1795/54550/20.578	
Extended Rosenbrock	60/130/0.406		50/110/0.375		50/110/0.328		50/110/0.313	
Extended BD1 (Block Diagonal)	1200/12605/4.938		1160/8976/4.813		1175/10059/5.188		1030/7974/4.188	
Extended Cliff	950/13187/4.891		46500/530421/273.781		4773/36089/21.719		1208/23850/12.703	
NONDQUAR (CUTE)	86/4989/15.781		45/2732/9.156		31/2721/10.156		24/2332/9.109	
DQDRTIC (CUTE)	3637/92315/28.766		2460/58286/19.141		7082/170118/49.594		2430/57850/19.938	
Extended Freudenstein and Roth	2018/66654/13.516		3524/113935/21.531		31821/1040192/213.016		2186/69397/14.531	
Extended Beale	181/4748/5.469		1593/19562/52.656		1880/24895/65.016		1300/17186/45.172	
EDENSCH (CUTE)	1684/22731/96.219		1300/16218/66.328		1050/13383/55.078		1132/14810/63.563	

Na kraju je važno napomenuti da EDL i MDL metode, koje su trenutno među najboljim Dai-Liao konjugovano gradijentnim metodama datim u literaturi, daju lošije numeričke rezultate od MSMDL i BB1DL metoda u smislu broja iterativnih koraka, broj evaluacija testiranih funkcija i CPU vremena.

6.5.4 Primena MSMDL metode u 2D robotskoj kontroli pokreta

Problemi koji proističu iz koncepta robotskih sistema privukli su pažnju istraživača, što je uticalo na to da se kasnije razviju neki algoritmi za njihovo rukovanje [111, 181]. Na primer, Zhang i ostali u [189] su diskutovali o osnovama robota sa n veza (*eng. n-link-robots*) poznatih kao robotski sistem sa 1 vezom (*eng. 1-link robot system*). Qiang i ostali u [138] su ukazali na važnost uzimanja u obzir karakteristika motoričke dinamike da bi se postigli zahtevi za tačnost i stabilnost pokreta robota. Štaviše, među kriterijumima koje motorička dinamika treba da zadovolji jeste da stvarni izlaz sistema prati željeni izlaz u okviru prihvatljive minimalne greške [163]. Motivisani radom Zhang i ostalih u [188], Sun i ostali u [161] su primenili algoritam za rešavanje problema bezuslovne optimizacije na rešavanje problema kontrole pokreta robotskog manipulatora (2D robotska kontrola pokreta).

Razmatramo problem kontrole pokreta koji uključuje dvozglobne robotske manipulatore u ravni (*eng. two-joint planar robotic manipulator*) kao što je opisano u [188]. Neka $\zeta_k \in \mathbb{R}^2$ i $\varphi_k \in \mathbb{R}^2$ označavaju vektor zglobnog ugla (*eng. joint angle*) i vektor položaja krajnjeg efektora (*eng. end effector position*), respektivno. U robotici, krajnji efektor je uređaj na kraju robotske ruke, dizajniran za interakciju sa okolinom. Tačna priroda ovog uređaja zavisi od primene robota.

Jednačina kinematike diskretnog vremena dvozglobnog robotskog manipulatora u ravni na nivou pozicije data je sledećim modelom

$$h(\zeta_k) = \varphi_k. \quad (6.98)$$

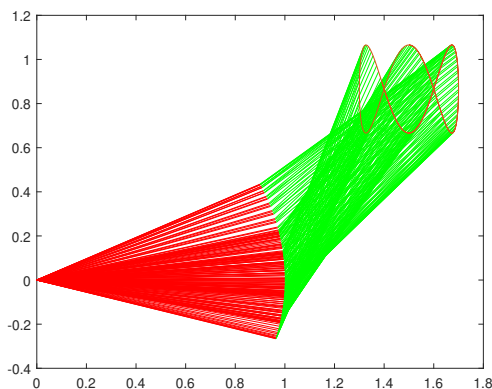
Funkcija sa vektorskom vrednošću $h(\cdot)$ se odnosi na kinematičko preslikavanje koje ima sledeću strukturu

$$h(\zeta) = \begin{bmatrix} \ell_1 \cos(\zeta_1) + \ell_2 \cos(\zeta_1 + \zeta_2), & \ell_1 \sin(\zeta_1) + \ell_2 \sin(\zeta_1 + \zeta_2) \end{bmatrix}^T,$$

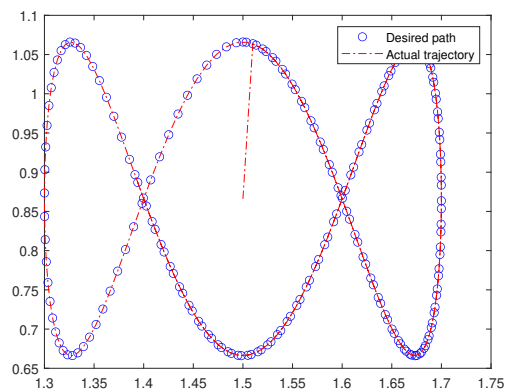
gde parametri ℓ_1 i ℓ_2 predstavljaju dužine prvog i drugog štapa, respektivno. Sada, što se tiče robotske kontrole pokreta, sledeći problem bezuslovne optimizacije

$$\min_{\varphi_k \in \mathbb{R}^2} \frac{1}{2} \|\varphi_k - \hat{\varphi}_k\|^2, \quad (6.99)$$

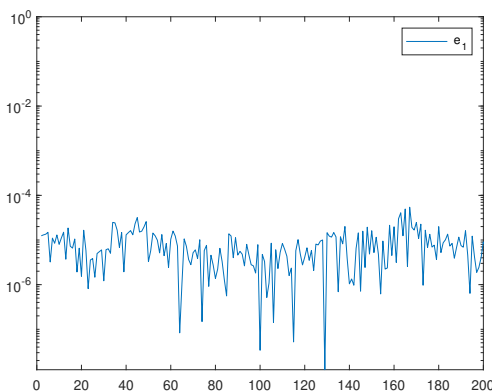
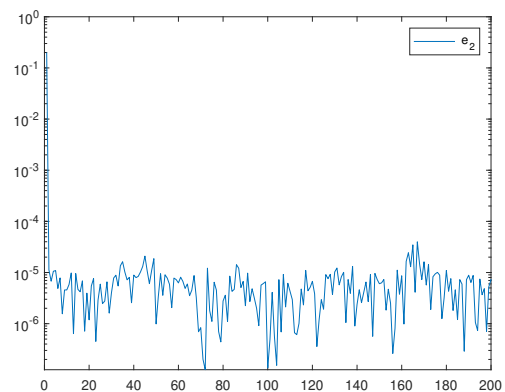
se rešava u svakom trenutku t_k unutar intervala $[0, t_{\text{end}}]$, pri čemu je t_{end} trajanje zadatka. Krajnji efektor φ_k se obično kontroliše tako da prati Lisažuvu (*eng. Lissajous*) krivu označenu sa $\hat{\varphi}_k$. Primećujemo da uzimajući $f(\mathbf{x}) \equiv \frac{1}{2} \|\varphi_k - \hat{\varphi}_k\|^2$, onda problem (6.99)



(a) Sintetizovane trajektorije robota.



(b) Trajektorija krajnjeg efektora i željena putanja.


 (c) Praćenje rezidualne greške na x -osi.

 (d) Praćenje rezidualne greške na y -osi.

 Slika 6.10: Numerički rezultati generisani MSMDL metodom za $\hat{\varphi}_k = \hat{\varphi}_{1k}$.

ima oblik problema (2.1) i stoga se predloženi MSMDL metod može koristiti za njegovo rešavanje.

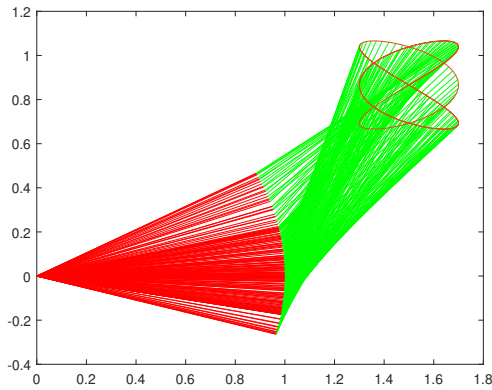
U ovom eksperimentu, za razliku od Lisažuove krive koja se koristi u [14, 161, 176], zahtevamo da krajnji efektor φ_k prati sledeće dve Lisažuove krive,

$$\hat{\varphi}_{1k} = \left[\frac{3}{2} + \frac{1}{5} \sin(t_k), \quad \frac{\sqrt{3}}{2} + \frac{1}{5} \sin\left(3t_k + \frac{\pi}{2}\right) \right]^T, \quad (6.100)$$

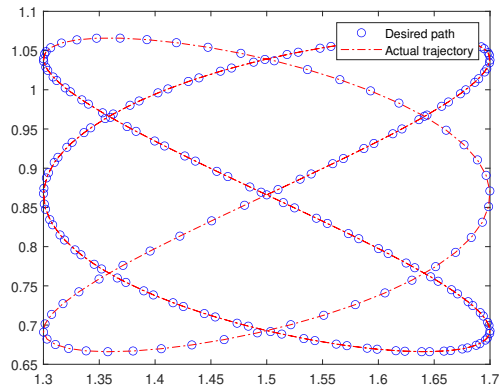
i

$$\hat{\varphi}_{2k} = \left[\frac{3}{2} + \frac{1}{5} \sin(3t_k), \quad \frac{\sqrt{3}}{2} + \frac{1}{5} \sin(2t_k) \right]^T. \quad (6.101)$$

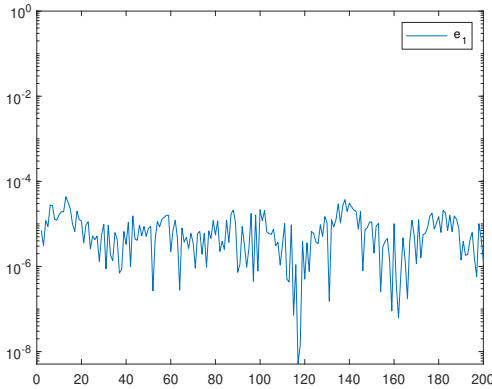
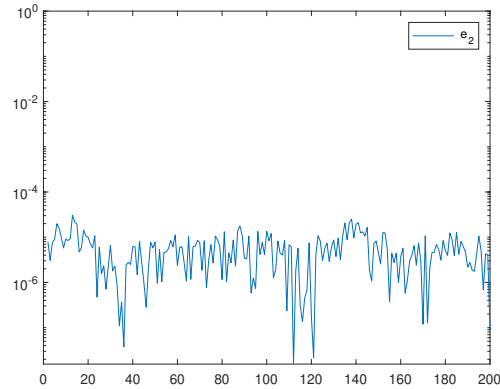
Implementacija predloženog MSMDL metoda u vezi sa eksperimentom kontrole pokreta je kodirana u MATLAB R2019b i pokrenuta na računaru sa intel Core(TM) i5-8250u procesorom sa 4 GB RAM-a i CPU 1.60 GHZ. U ovom eksperimentu, dužine prvog i drugog



(a) Sintetizovane trajektorije robota.



(b) Trajektorija krajnjeg efektora i željena putanja.


 (c) Praćenje rezidualne greške na x -osi.

 (d) Praćenje rezidualne greške na y -osi.

 Slika 6.11: Numerički rezultati generisani MSMDL metodom za $\hat{\varphi}_k = \hat{\varphi}_{2k}$.

štapa se uzimaju kao $\ell_1 = \ell_2 = 1$, gde je početna tačka $\zeta_0 = [\zeta_1, \zeta_2] = [0, \frac{\pi}{3}]^T$ sa trajanjem zadatka $[0, t_{\text{end}}]$ podeljenim na 200 jednakih delova, gde je $t_{\text{end}} = 10$ sekundi. Eksperimentalni rezultati za $\hat{\varphi}_k = \hat{\varphi}_{1k}$ su predstavljeni na Slici 6.10, dok su rezultati za $\hat{\varphi}_k = \hat{\varphi}_{2k}$ dati na Slici 6.11. Slike 6.10(a) i 6.11(a) prikazuju trajektorije (putanje) robota koje je sintetizovao MSMDL metod za $\hat{\varphi}_{1k}$ i $\hat{\varphi}_{2k}$, respektivno. S druge strane, Slike 6.10(b) i 6.11(b), respektivno, prikazuju trajektoriju krajnjeg efektora i željenu putanju za $\hat{\varphi}_{1k}$ i $\hat{\varphi}_{2k}$.

Greške koje je MSMDL metod zabeležio tokom eksperimenta u odnosu na horizontalnu x -osu prikazane su na Slikama 6.10(c) i 6.11(c) za $\hat{\varphi}_{1k}$ i $\hat{\varphi}_{2k}$, respektivno. Nasuprot tome, greške na vertikalnoj y -osi su predstavljene na Slikama 6.10(d) i 6.11(d) za $\hat{\varphi}_{1k}$ i $\hat{\varphi}_{2k}$, respektivno.

Slike 6.10(a), 6.10(b), 6.11(a) i 6.11(b) potvrđuju da je MSMDL metod uspešno i efikasno izvršio zadatak koji je dat sa prihvatljivim greškama. Pored toga, Slike 6.10(c), 6.10(d), 6.11(c) i 6.11(d) pokazuju da predloženi MSMDL metod ne samo da rešava probleme testiranja, već ih upotpunjuje prihvatljivom greškom. Ovo dalje pokazuje efikasnost i primenljivost MSMDL metode.

Poglavlje 7

Konjugovani gradijentni metodi Dai-Liao tipa sa poboljšanim parametrima za rešavanje sistema nelinearnih monotonih jednačina velikih dimenzija

U ovom poglavlju biće predstavljen postupak za rešavanje sistema nelinearnih monotonih jednačina velikih dimenzija koji kombinuje projektni metod i pravac pretraživanja Dai-Liao tipa konjugovanog gradijenta. Pored predstavljanja bazičnog DLPM algoritma, biće predstavljeni naučni rezultati i MSMDLPM algoritam iz [84] koji predstavlja deo istraživanja ove doktorske disertacije.

Ivanov i ostali autori u radu [84] predložili su novi pravac pretraživanja koji proizilazi iz odgovarajućih parametara ubrzanja dobijenih nakon nadogradnje ubrzanog gradijentnog MSM metoda sa DLPM metodom. Glavni cilj predložene MSMDLPM metode je da poveže MSM i DLPM metod i stvori efikasan metod za rešavanje sistema nelinearnih monotonih jednačina velikih dimenzija. Globalna konvergencija i brzina konvergencije MSMDLPM metode su teorijski istraženi i dokazani, dok numerički rezultati potvrđuju efikasnost predloženog metoda. Kao posledica rešavanja sistema nelinearnih monotonih jednačina velikih dimenzija data je efikasna primena MSMDLPM metode u restauraciji slika, a samim tim data je veza između dve naučne oblasti: nelinearna optimizacija i obrada (restauracija) slika.

7.1 Dai-Liao projektni metod

Sistem nelinearnih monotonih jednačina ima opšti oblik

$$F(\mathbf{x}) = 0, \quad \mathbf{x} \in \mathbb{R}^n, \quad (7.1)$$

gde je $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ neprekidno preslikavanje koje zadovoljava uslov monotonosti

$$(F(\mathbf{x}) - F(\mathbf{y}))^T(\mathbf{x} - \mathbf{y}) \geq 0, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n. \quad (7.2)$$

Generalno, problem (7.1) može proizaći iz problema bezuslovne optimizacije. Neka je f norma funkcije definisana na sledeći način

$$f(\mathbf{x}) = \frac{1}{2} \|F(\mathbf{x})\|^2, \quad \mathbf{x} \in \mathbb{R}^n, \quad (7.3)$$

gde je $\|\cdot\|$ Euklidova norma. Tada je problem nelinearnih jednačina (7.1) ekvivalentan sledećem globalnom problemu bezuslovne optimizacije

$$\min f(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n, \quad (7.4)$$

gde je ciljna funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i čiji minimum želimo da odredimo.

Za svako rešenje $\mathbf{x}^*$ sistema jednačina (7.1) važi $f(\mathbf{x}^*) = 0$, a kako je $f(\mathbf{x}) \geq 0$ za svako $\mathbf{x} \in \mathbb{R}^n$ na osnovu definicije (7.3) funkcije $f(\mathbf{x})$, onda svako $\mathbf{x}^*$ minimizira funkciju f . Međutim, ako je funkcija f strogo pozitivna u tačkama u kojima dostiže lokalni minimum, onda te tačke nisu rešenja sistema jednačina (7.1). Uprkos ovom nedostatku, ovaj pristup, u kojem se za rešavanje sistema jednačina (7.1) koriste metode za rešavanje nelinearnog problema bezuslovne optimizacije (7.4), je implementiran u mnogim softverskim paketima i u praksi se uspešno primenjuje.

Sistemi nelinearni monotoni jednačina se javljaju kao problemi u mnogim naučnim oblastima, pa je rešavanje ovakvih sistema vrlo popularno među naučnim istraživačima. Monotonost sistema jednačina nameće primenu projektivog metoda pogodnog za rešavanje ovakvih sistema jednačina, jer omogućava jednostavnu globalizaciju. Kod projektivnog metoda, linijsko pretraživanje se koristi u cilju konstrukcije hiperravni koja trenutnu iteraciju razdvaja od skupa rešenja sistema jednačina. Međutim, linijskim pretraživanjem u datom pravcu određuje se dužina koraka i nakon toga generiše pomoćna iteracija koja služi za konstrukciju hiperravni koja razdvaja rešenje sistema jednačina od trenutne iteracije. Naredna iteracija nastaje projektovanjem trenutne iteracije na datu hiperravan. Na ovaj način se postiže globalna konvergencija.

U radu [150] prvi put je predstavljen projektivni metod za sisteme monotoni jednačina, gde je predložen hibridni projektivni netačni Njutnov postupak koji kombinuje projektivni metod sa Njutnovim pravcem pretraživanja. Glavna karakteristika hibridnog projektivnog netačnog Njutnovog postupka je da čitav niz iteracija generisan algoritmom globalno konvergira ka rešenju sistema jednačina bez pretpostavke o regularnosti. Pored toga, ovi postupci ne koriste funkciju cilja i izvode i baziraju se samo na izračunavanju vrednosti funkcije sistema $F(\mathbf{x})$. Zbog toga su pogodni za rešavanje sistema sa singularnim rešenjima, kao i neglatkih monotoni sistema.

Kako projektni metod predstavlja osnovu za čitavu klasu projektnih postupaka u nastavku biće izložena osnovna ideja o upotrebi Dai–Liao metode konjugovanih gradijenata za rešavanje sistema nelinearnih jednačina. Preciznije, biće prikazan način proširivanja Dai–Liao metode iz [18] za rešavanje problema (7.1). Dobijeni metod je kombinacija Dai–Liao konjugovano gradijentnog metoda za rešavanje problema bezuslovne optimizacije i tehnike projekcije hiperravnih.

Iterativna šema za rešavanje problema (7.1) počinje od početne tačke $\mathbf{x}_0 \in \mathbb{R}^n$ i generiše niz $\{\mathbf{x}_k, k \geq 0\}$ koji ima opšti oblik

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k, \quad k \geq 0, \quad (7.5)$$

gde je $\mathbf{s}_k = \alpha_k \mathbf{d}_k$, $\alpha_k > 0$ i predstavlja dužinu koraka dobijenu odgovarajućom tehnikom linijskog pretraživanja, a vektor $\mathbf{d}_k$ je pravac pretraživanja. Ključni problem je pronaći pravac pretraživanja $\mathbf{d}_k$ i pogodnu veličinu dužine koraka α_k .

Na osnovu pravca pretraživanja $\mathbf{d}_k$ potrebno je odrediti pomoćnu iteraciju $\mathbf{z}_k = \mathbf{x}_k + \mathbf{s}_k$ koja se koristi za kreiranje hiperravnih

$$H_k = \{\mathbf{x} \in \mathbb{R}^n : F(\mathbf{z}_k)^T(\mathbf{x} - \mathbf{z}_k) = 0\}. \quad (7.6)$$

Tačka $\mathbf{z}_k$ se određuje na taj način da hiperravan H_k koja je kreirana tom tačkom razdvaja skup rešenja sistema nelinearnih monotoni jednačina (7.1) od trenutne iteracije $\mathbf{x}_k$. Međutim, način na koji je određena tačka $\mathbf{z}_k$ zahteva upotrebu linijskog pretraživanja koje određuje dužinu koraka α_k tako da hiperravan H_k kreirana sa $\mathbf{z}_k = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ striktno razdvaja trenutnu iteraciju od skupa rešenja sistema nelinearnih monotoni jednačina (7.1). Da bi se ovo ostvarilo, potrebno je da važi uslov

$$F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k) > 0, \quad (7.7)$$

koji obezbeđuje ovu osobinu tačke $\mathbf{z}_k$. Sa druge strane, kako funkcija $F(\mathbf{x})$ zadovoljava uslov monotonosti, sledi da za svako rešenje $\mathbf{x}^*$ sistema nelinearnih monotoni jednačina (7.1) važi

$$F(\mathbf{z}_k)^T(\mathbf{x}^* - \mathbf{z}_k) = (F(\mathbf{z}_k) - F(\mathbf{x}^*))^T(\mathbf{x}^* - \mathbf{z}_k) \leq 0. \quad (7.8)$$

Na osnovu uslova koji važe u (7.7) i (7.8) sledi da hiperravan H_k striktno razdvaja trenutnu iteraciju $\mathbf{x}_k$ od rešenja sistema nelinearnih monotoni jednačina $\mathbf{x}^*$. Koristeći ovu činjenicu, Solodov i Svaiter u [150] predlažu da se naredna iteracija $\mathbf{x}_{k+1}$ dobija projekcijom trenutne iteracije $\mathbf{x}_k$ na hiperravan H_k , na sledeći način:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} F(\mathbf{z}_k). \quad (7.9)$$

Složenost algoritma (metode) za rešavanje nelinearnih monotoni sistema jednačina (7.1) direktno zavisi od definisanja pravca pretraživanja $\mathbf{d}_k$ i načina izračunavanja dužine koraka

α_k , tako da je vrlo bitno odabrati ispravnu tehniku za linijsko pretraživanje. Linijsko pretraživanje bez korišćenja izvoda i vrednosti funkcije cilja dato u radovima [4, 97] pogodno je za rešavanje sistema velikih dimenzija, jer se zasniva samo na izračunavanju vrednosti funkcije $F(\mathbf{x})$. Dužina koraka $\alpha_k > 0$ je najveća dužina $\alpha_k = r^{m_k}$, $r \in (0, 1)$, gde je m_k najmanji nenegativan ceo broj takav da zadovoljava sledeći uslov

$$-F(\mathbf{x}_k + \alpha_k \mathbf{d}_k)^T \mathbf{d}_k \geq \sigma \alpha_k \|F(\mathbf{x}_k + \alpha_k \mathbf{d}_k)\| \|\mathbf{d}_k\|^2, \quad \sigma \in (0, 1), \quad (7.10)$$

a $\mathbf{d}_k$ pravac pretraživanja koji zadovoljava dovoljan uslov opadanja

$$F(\mathbf{x}_k)^T \mathbf{d}_k \leq -c \|F(\mathbf{x}_k)\|^2, \quad \text{za } c > 0. \quad (7.11)$$

U radu [1] autori su definisali pravac pretraživanja $\mathbf{d}_k$ na sledeći način:

$$\mathbf{d}_k = \begin{cases} -F(\mathbf{x}_k), & k=0, \\ -F(\mathbf{x}_k) + \beta_k \mathbf{d}_{k-1}, & k \geq 1, \end{cases} \quad (7.12)$$

gde je β_k Dai-Liao konjugovani gradijentni parametar oblika

$$\beta_k = \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{y}_{k-1}^T \mathbf{d}_{k-1}} - t_k \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{y}_{k-1}^T \mathbf{d}_{k-1}}, \quad (7.13)$$

$$t_k = p \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} - q \frac{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}{\|\mathbf{s}_{k-1}\|^2}, \quad p \geq \frac{1}{4}, \quad q \leq 0, \quad (7.14)$$

pri čemu su $\mathbf{s}_k$ i $\mathbf{y}_k$ određeni na sledeći način: $\mathbf{s}_k = \mathbf{z}_k - \mathbf{x}_k = \alpha_k \mathbf{d}_k$ i $\mathbf{y}_k = F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k)$.

Konjugovani gradijentni parametar β_k u (7.13) uopštava dva poznata konjugovana gradijetna parametra za različite vrednosti konstanti p i q . Ako je $p = 2$ i $q = 0$, tada je t_k iz (7.14) oblika

$$t_k = 2 \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{y}_{k-1}^T \mathbf{s}_{k-1}}. \quad (7.15)$$

Zamenom t_k iz (7.15) u (7.13) dobijamo konjugovani gradijentni parametar β_k koji je poznat kao CG-DESCENT metod o kome je bilo više reči u Odeljku 5.2. Takođe, ako je $p = 1$ i $q = 0$, β_k u (7.13) se svodi na konjugovani gradijentni parametar β_k (obeležen sa DK) i definisan od strane Dai i Kou u radu [41], gde je t_k oblika

$$t_k = \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{y}_{k-1}^T \mathbf{s}_{k-1}}. \quad (7.16)$$

Uticaj parametra t_k u Dai-Liao klasi konjugovanih gradijenata je vrlo značajan, pa zbog toga postoji veliki broj istraživanja i pronalaska različitih načina određivanja vrednosti parametra t_k . U numeričkim rezultatima u ovom poglavlju koristićemo još neke varijante parametra t_k koje su definisane u Odeljku 5.2 kako bi smo uporedili i analizirali više agloritama koji pripadaju Dai-Liao klasi.

U nastavku je dat Algoritam 7.1.1 za Dai–Liao projektni metod, odnosno skraćeno DLPM metod [1].

Algoritam 7.1.1 Dai–Liao projektni metod (DLPM).

Ulaz: Izabrati početnu tačku $\mathbf{x}_0$, realne konstante $r, \sigma \in (0, 1)$, uslov zaustavljanja $\epsilon > 0$ i postaviti $k := 0$.

- 1: Izračunati $F(\mathbf{x}_k)$. Ako je kriterijum zaustavljanja zadovoljen $\|F(\mathbf{x}_k)\| \leq \epsilon$, onda STOP, inače pređi na Korak 2.
- 2: Izračunati pravac pretraživanja $\mathbf{d}_k$ na osnovu (7.12). STOP ako je $\mathbf{d}_k = 0$, inače pređi na Korak 3.
- 3: Izračunati dužinu koraka α_k na osnovu (7.10).
- 4: Izračunati $\mathbf{z}_k = \mathbf{x}_k + \alpha_k \mathbf{d}_k$. Ako je $\|F(\mathbf{z}_k)\| = 0$, tada $\mathbf{x}_{k+1} = \mathbf{z}_k$, STOP. Inače pređi na sledeći korak.
- 5: Izračunati $\mathbf{x}_{k+1}$ koristeći obrazac (7.9).
- 6: Odrediti t_{k+1} koristeći (7.14).
- 7: Izračunati β_{k+1} koristeći (7.13).
- 8: Postaviti $k := k + 1$ i pređi na Korak 1.

Izlaz: $\mathbf{x}_k$ i $F(\mathbf{x}_k)$.

7.2 MSM Dai–Liao projektni metod (MSMDLPM)

Polazna tačka naše iterativne šeme datira iz MSM šeme definisane u [86]. Adaptacija MSM metode na problem (7.1) postaje

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (\varphi_k + \varphi_k^2 - \varphi_k^3)(\gamma_k^{\text{MSMDLPM}})^{-1}F(\mathbf{x}_k), \quad (7.17)$$

gde je $\gamma_k^{\text{MSMDLPM}} > 0$ odgovarajuće definisan parametar ubrzanja, a $\varphi_k \in (0, 1]$ je dodatni parametar. Izraz $1 + \varphi_k - \varphi_k^2 > 1$ biće kratko označen sa ψ_k i korišćen za dodatno ubrzanje. Prema ovom dogovoru, pravac pretraživanja u (7.17) je definisan na sledeći način

$$\mathbf{d}_k = -\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1}F(\mathbf{x}_k). \quad (7.18)$$

Vrednost φ_k se određuje pomoću (7.10), tj. $\varphi_k = \alpha_k$.

Parametar ubrzanja $\gamma_k^{\text{MSMDLPM}}$ u (7.18) određen je primenom Tejlorovog razvoja prvog reda na funkciju $F(\mathbf{x}_{k+1})$ u tački $\mathbf{x}_{k+1}$

$$\begin{aligned} F(\mathbf{x}_{k+1}) &\approx F(\mathbf{x}_k) + \nabla F(\xi)(\mathbf{x}_{k+1} - \mathbf{x}_k) \\ &\approx F(\mathbf{x}_k) + \nabla F(\xi)\varphi_k \mathbf{d}_k \\ &\approx F(\mathbf{x}_k) - \nabla F(\xi)\varphi_k \psi_k (\gamma_k^{\text{MSMDLPM}})^{-1}F(\mathbf{x}_k), \quad \xi \in [\mathbf{x}_k, \mathbf{x}_{k+1}]. \end{aligned} \quad (7.19)$$

Sledeći [86], aproksimacija matrice funkcije F data je dijagonalnom matricom

$$\nabla F(\xi) \approx \gamma_{k+1}^{\text{MSMDLPM}} I.$$

Onda aproksimacija u (7.19) postaje

$$F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k) \approx -\gamma_{k+1}^{\text{MSMDLPM}} \varphi_k \psi_k (\gamma_k^{\text{MSMDLPM}})^{-1} F(\mathbf{x}_k). \quad (7.20)$$

Iz (7.20) i $\mathbf{y}_k = F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k)$, sledi

$$\mathbf{y}_k = -\gamma_{k+1}^{\text{MSMDLPM}} \varphi_k \psi_k (\gamma_k^{\text{MSMDLPM}})^{-1} F(\mathbf{x}_k). \quad (7.21)$$

Množenjem obe strane jednakosti u (7.21) sa $\mathbf{y}_k^T$, parametar ubrzanja $\gamma_{k+1}^{\text{MSMDLPM}}$ može se izraziti iz (7.21) na sledeći način:

$$\gamma_{k+1}^{\text{MSMDLPM}} = -\frac{\mathbf{y}_k^T \mathbf{y}_k}{\varphi_k \psi_k (\gamma_k^{\text{MSMDLPM}})^{-1} \mathbf{y}_k^T F(\mathbf{x}_k)}. \quad (7.22)$$

Da bi ispunili potreban i dovoljan uslov optimalnosti drugog reda, neželjene vrednosti $\gamma_{k+1}^{\text{MSMDLPM}} < 0$ biće zamenjene sa $\gamma_{k+1}^{\text{MSMDLPM}} = 1$ [86, 155].

Nepoznati parametar t_k u (7.14) može se odrediti iz (7.12), (7.13) i (7.18). Zamenom (7.13) u (7.12), $\mathbf{d}_k$ postaje

$$\begin{aligned} \mathbf{d}_k &= -F(\mathbf{x}_k) + \beta_k \mathbf{d}_{k-1} \\ &= -F(\mathbf{x}_k) + \left(\frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right) \mathbf{d}_{k-1}. \end{aligned} \quad (7.23)$$

Imajući u vidu (7.18), nakon zamene $\mathbf{d}_k = -\psi_k (\gamma_k^{\text{MSMDLPM}})^{-1} F(\mathbf{x}_k)$ u (7.23) dobija se

$$-\psi_k (\gamma_k^{\text{MSMDLPM}})^{-1} F(\mathbf{x}_k) = -F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{d}_{k-1} - t_k \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{d}_{k-1}. \quad (7.24)$$

Sledeća jednačina se dobija množenjem (7.24) sa $F(\mathbf{x}_k)^T$:

$$\begin{aligned} -\psi_k (\gamma_k^{\text{MSMDLPM}})^{-1} F(\mathbf{x}_k)^T F(\mathbf{x}_k) &= -F(\mathbf{x}_k)^T F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1} \\ &\quad - t_k \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}. \end{aligned} \quad (7.25)$$

Iz prethodne jednakosti (7.25) sledi

$$\begin{aligned}
 & t_k \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1} \\
 &= -\|F(\mathbf{x}_k)\|^2 + \psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} \|F(\mathbf{x}_k)\|^2 + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1} \\
 &= (\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} - 1) \|F(\mathbf{x}_k)\|^2 + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}.
 \end{aligned}$$

Radi jednostavnosti, označite

$$R_k(F) := (\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} - 1) \|F(\mathbf{x}_k)\|^2. \quad (7.26)$$

U tom slučaju parametar t_k može se izraziti u sledećem obliku

$$t_k = \frac{R_k(F) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}{\frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}. \quad (7.27)$$

Kako je $\mathbf{s}_{k-1} = \alpha_{k-1} \mathbf{d}_{k-1} = \varphi_{k-1} \mathbf{d}_{k-1}$ i $\varphi_{k-1} \in (0, 1]$, nakon zamene $\mathbf{d}_{k-1} = \mathbf{s}_{k-1} / \varphi_{k-1}$ u (7.27) i nakon pojednostavljenja, dobija se

$$\begin{aligned}
 t_k &= \frac{R_k(F) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\frac{1}{\varphi_{k-1}} \mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \frac{1}{\varphi_{k-1}} \mathbf{s}_{k-1}}{\frac{\frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\frac{1}{\varphi_{k-1}} \mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \frac{1}{\varphi_{k-1}} \mathbf{s}_{k-1}}{}} \\
 &= \frac{R_k(F) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\frac{(F(\mathbf{x}_k)^T \mathbf{s}_{k-1})^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}} \\
 &= \frac{R_k(F) \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{(F(\mathbf{x}_k)^T \mathbf{s}_{k-1})^2}. \quad (7.28)
 \end{aligned}$$

Da bi se obezbedio uslov opadanja u novoj Dai–Liao metodi, definicija za t_k u (7.28) je modifikovana korišćenjem principa iz [5, 107] u sledećem obliku

$$t_k^{\text{MSMDLPM}} = \max \left\{ t_k, \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \right\}, \quad \theta > \frac{1}{4}. \quad (7.29)$$

Uzimajući u obzir $t_k := t_k^{\text{MSMDLPM}}$ u definiciji β_k u (7.13), predlažemo novo poboljšanje Dai–Liao konjugovano gradijentnog parametra β_k definisano sledećim pravilom:

$$\beta_k^{\text{MSMDLPM}} = \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^{\text{MSMDLPM}} \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}. \quad (7.30)$$

Iteracije definisane iz opšte DLPM klase, koje su određene sa β_k^{MSMDLPM} , označavaju se kao MSM Dai-Liao projekтивni metod, ili skraćeno MSMDLPM.

Algoritam 7.2.1 je formalni opis predloženog MSMDLPM metoda.

Algoritam 7.2.1 MSM Dai-Liao projekтивni metod (MSMDLPM).

Ulaz: Izabrati početnu tačku $\mathbf{x}_0 \in \mathbb{R}^n$, realne konstante $r, \sigma \in (0, 1)$, uslov zaustavljanja $\varepsilon > 0$ i postaviti $k := 0$.

- 1: Izračunati $F(\mathbf{x}_k)$. Ako je kriterijum zaustavljanja zadovoljen $\|F(\mathbf{x}_k)\| \leq \varepsilon$, onda STOP, inače pređi na Korak 2.
- 2: Izračunati pravac pretraživanja $\mathbf{d}_k$ koristeći (7.12). STOP ako je $\mathbf{d}_k = 0$, inače pređi na sledeći Korak.
- 3: Izračunati dužinu koraka α_k na osnovu (7.10) i postaviti $\varphi_k = \alpha_k$.
- 4: Izračunati $\mathbf{z}_k = \mathbf{x}_k + \varphi_k \mathbf{d}_k$. Ako je $\|F(\mathbf{z}_k)\| = 0$, onda STOP i $\mathbf{x}_{k+1} = \mathbf{z}_k$, inače pređi na sledeći Korak.
- 5: Izračunati $\mathbf{x}_{k+1}$ koristeći (7.9).
- 6: Izračunati $F(\mathbf{x}_{k+1})$, $\mathbf{s}_k = \mathbf{z}_k - \mathbf{x}_k = \varphi_k \mathbf{d}_k$ i $\mathbf{y}_k = F(\mathbf{x}_{k+1}) - F(\mathbf{x}_k)$.
- 7: Odrediti $\gamma_{k+1}^{\text{MSMDLPM}}$ koristeći (7.22). Ako je $\gamma_{k+1}^{\text{MSMDLPM}} < 0$, onda postaviti $\gamma_{k+1}^{\text{MSMDLPM}} = 1$.
- 8: Odrediti t_{k+1}^{MSMDLPM} koristeći (7.29).
- 9: Izračunati $\beta_{k+1}^{\text{MSMDLPM}}$ koristeći (7.30).
- 10: Postaviti $k := k + 1$ i pređi na Korak 1.

Izlaz: $\mathbf{x}_k$ i $F(\mathbf{x}_k)$.

7.3 Analiza konvergenције

U ovom odeljku istražujemo globalnu konvergenciju i brzinu konvergenције MSMDLPM metode definisane pomoću Algoritma 7.2.1.

7.3.1 Globalna konvergenција

U ovom pododeljku, pod standardnim pretpostavkama biće dokazana globalna konvergenција MSMDLPM metode. Da bi se dokazala globalna konvergenција MSMDLPM metode, odnosno Algoritma 7.2.1. potrebne su sledeće pretpostavke.

Pretpostavka 7.3.1. (P1) Funkcija $F(\mathbf{x})$ je uniformno monotona na $\mathbb{R}^n$, ako postoji $c \geq 0$ tako da važi

$$(F(\mathbf{x}) - F(\mathbf{y}))^T(\mathbf{x} - \mathbf{y}) \geq c\|\mathbf{x} - \mathbf{y}\|^2, \quad (7.31)$$

za svako $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

(P2) Funkcija $F(\mathbf{x})$ je Lipšic neprekidna, ako postoji pozitivna realna konstanta $L > 0$ tako da važi

$$\|F(\mathbf{x}) - F(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad (7.32)$$

za svako $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

Iz Pretpostavka 7.3.1 (P2) proizilazi da postoji pozitivna konstanta ω takva da je

$$\|F(\mathbf{x})\| \leq \omega. \quad (7.33)$$

Lema 7.3.1. *Neka je niz $\{\mathbf{x}_k\}$ generisan MSMDLPM metodom, odnosno Algoritmom 7.2.1. Onda pravac pretraživanja $\mathbf{d}_k$ zadovoljava sledeći uslov:*

$$(\forall k \geq 0) \quad F(\mathbf{x}_k)^T \mathbf{d}_k \leq -\lambda \|F(\mathbf{x}_k)\|^2, \quad \lambda > 0. \quad (7.34)$$

Dokaz. Za $k = 0$ i $\mathbf{d}_0 = -F(\mathbf{x}_0)$, početna aproksimacija zadovoljava jednakost

$$F(\mathbf{x}_0)^T \mathbf{d}_0 = -F(\mathbf{x}_0)^T F(\mathbf{x}_0) = -\|F(\mathbf{x}_0)\|^2.$$

Iz koje sledi da nejednakost (7.34) važi za $\lambda = 1$. Za $k \geq 1$, $\mathbf{s}_{k-1} = \varphi_{k-1} \mathbf{d}_{k-1}$ i $\mathbf{d}_k = -F(\mathbf{x}_k) + \beta_k^{\text{MSMDLPM}} \mathbf{d}_{k-1}$, imamo da važi

$$\begin{aligned} F(\mathbf{x}_k)^T \mathbf{d}_k &= -\|F(\mathbf{x}_k)\|^2 + \beta_k^{\text{MSMDLPM}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1} \\ &= -\|F(\mathbf{x}_k)\|^2 + \left(\frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - t_k^{\text{MSMDLPM}} \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right) F(\mathbf{x}_k)^T \mathbf{d}_{k-1} \\ &= -\|F(\mathbf{x}_k)\|^2 + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1} - t_k^{\text{MSMDLPM}} \frac{\varphi_{k-1} (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}. \end{aligned} \quad (7.35)$$

Na osnovu definicije t_k^{MSMDLPM} u (7.29) razlikujemo dva slučaja.

(i): Ako je

$$t_k^{\text{MSMDLPM}} = \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} = \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\varphi_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \quad \text{ i } \quad \theta > 1/4,$$

onda se iz jednakosti (7.35) dobija

$$F(\mathbf{x}_k)^T \mathbf{d}_k = -\|F(\mathbf{x}_k)\|^2 + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} F(\mathbf{x}_k)^T \mathbf{d}_{k-1} - \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\varphi_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \frac{\varphi_{k-1} (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}},$$

odnosno,

$$\begin{aligned} F(\mathbf{x}_k)^T \mathbf{d}_k &= \frac{1}{(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2} \left\{ -\|F(\mathbf{x}_k)\|^2 (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2 \right. \\ &\quad + (F(\mathbf{x}_k)^T \mathbf{y}_{k-1}) (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}) (F(\mathbf{x}_k)^T \mathbf{d}_{k-1}) \\ &\quad \left. - \theta \|\mathbf{y}_{k-1}\|^2 (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2 \right\}. \end{aligned} \quad (7.36)$$

Primenom nejednakosti $\mathbf{u}^T \mathbf{v} \leq \frac{1}{2}(\|\mathbf{u}\|^2 + \|\mathbf{v}\|^2)$ na nejednakost (7.36), gde je $\mathbf{u} =$

$\frac{1}{\sqrt{2\theta}}(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})F(\mathbf{x}_k)$ i $\mathbf{v} = \sqrt{2\theta}(F(\mathbf{x}_k)^T \mathbf{d}_{k-1})\mathbf{y}_{k-1}$, dobija se

$$F(\mathbf{x}_k)^T \mathbf{d}_k \leq -\|F(\mathbf{x}_k)\|^2 + \frac{Q_k(F)}{p_k^2},$$

gde je $p_k = \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}$ i

$$\begin{aligned} Q_k(F) &= \frac{1}{2} \left(\left\| \frac{p_k}{\sqrt{2\theta}} F(\mathbf{x}_k) \right\|^2 + \left\| \sqrt{2\theta} (F(\mathbf{x}_k)^T \mathbf{d}_{k-1}) \mathbf{y}_{k-1} \right\|^2 \right) - \theta \|\mathbf{y}_{k-1}\|^2 (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2 \\ &= \frac{p_k^2}{4\theta} \|F(\mathbf{x}_k)\|^2 + \theta (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2 \|\mathbf{y}_{k-1}\|^2 - \theta \|\mathbf{y}_{k-1}\|^2 (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2 \\ &= \frac{p_k^2}{4\theta} \|F(\mathbf{x}_k)\|^2. \end{aligned}$$

Na osnovu svega prethodnog sledi

$$F(\mathbf{x}_k)^T \mathbf{d}_k \leq -\|F(\mathbf{x}_k)\|^2 + \frac{1}{4\theta} \|F(\mathbf{x}_k)\|^2 = -\left(1 - \frac{1}{4\theta}\right) \|F(\mathbf{x}_k)\|^2. \quad (7.37)$$

Kako je $\theta > \frac{1}{4}$, nejednakost (7.34) je zadovoljena za $k \geq 1$ i $\lambda = 1 - \frac{1}{4\theta}$ u (7.37).

(ii): Ako je $R_k(F)$ definisano kao u (7.26) i ako je

$$\begin{aligned} t_k^{\text{MSMDLPM}} &= \frac{R_k(F) \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{(F(\mathbf{x}_k)^T \mathbf{s}_{k-1})^2} \\ &= \frac{R_k(F) \varphi_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1} + \varphi_{k-1} F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}{\varphi_{k-1}^2 (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2} \\ &= \frac{R_k(F) \mathbf{d}_{k-1}^T \mathbf{y}_{k-1} + F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}{\varphi_{k-1} (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2}, \end{aligned} \quad (7.38)$$

onda se iz jednakosti (7.35) dobija

$$\begin{aligned} F(\mathbf{x}_k)^T \mathbf{d}_k &= -\|F(\mathbf{x}_k)\|^2 + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\ &\quad - \frac{R_k(F) \mathbf{d}_{k-1}^T \mathbf{y}_{k-1} + F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}{\varphi_{k-1} (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2} \cdot \frac{\varphi_{k-1} (F(\mathbf{x}_k)^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\ &= -\|F(\mathbf{x}_k)\|^2 + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - \frac{R_k(F) \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\ &\quad - \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{d}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\ &= -\|F(\mathbf{x}_k)\|^2 - R_k(F) = -\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} \|F(\mathbf{x}_k)\|^2. \end{aligned} \quad (7.39)$$

Kako je $\gamma_k^{\text{MSMDLPM}} > 0$ i $\varphi_k \in (0, 1]$, onda je i $\psi_k > 0$, na osnovu čega sledi da je nejednakost (7.34) zadovoljena za $\lambda = \psi_k(\gamma_k^{\text{MSMDLPM}})^{-1}$ u (7.39). $\square$

Lema 7.3.2 ([150]). *Pretpostavimo da Pretpostavka 7.3.1 važi. Ako je $\mathbf{x}^*$ rešenje sistema $F(\mathbf{x}^*) = 0$ i ako je niz $\{\mathbf{x}_k\}$ generisan Algoritmom 7.2.1 onda važi*

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 \leq \|\mathbf{x}_k - \mathbf{x}^*\|^2 - \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2, \quad (7.40)$$

i niz $\{\mathbf{x}_k\}$ je ograničen. Takođe, ili je niz $\{\mathbf{x}_k\}$ konačan pri čemu je rešenje problema (7.1) poslednja iteracija ili je niz $\{\mathbf{x}_k\}$ beskonačan i

$$\sum_{k=0}^{\infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| < \infty, \quad (7.41)$$

što implicira

$$\lim_{k \rightarrow \infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| = 0.$$

Lema 7.3.3. *Neka je niz $\{\mathbf{x}_k\}$ generisan MSMDLPM metodom, onda važi*

$$\lim_{k \rightarrow \infty} \varphi_k \|\mathbf{d}_k\| = 0. \quad (7.42)$$

Dokaz. Koristeći definiciju za $\mathbf{x}_{k+1}$ u (7.9) i uslov (7.10), imamo da važi

$$\begin{aligned} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| &= \left\| \mathbf{x}_k - \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} F(\mathbf{z}_k) - \mathbf{x}_k \right\| \\ &= \frac{|F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)|}{\|F(\mathbf{z}_k)\|^2} \|F(\mathbf{z}_k)\| = \frac{|F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)|}{\|F(\mathbf{z}_k)\|}, \end{aligned}$$

odnosno,

$$\begin{aligned} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| &= \frac{|\varphi_k F(\mathbf{z}_k)^T \mathbf{d}_k|}{\|F(\mathbf{z}_k)\|} = \frac{\varphi_k |F(\mathbf{z}_k)^T \mathbf{d}_k|}{\|F(\mathbf{z}_k)\|} \\ &\geq \frac{\sigma \varphi_k^2 \|F(\mathbf{z}_k)\| \|\mathbf{d}_k\|^2}{\|F(\mathbf{z}_k)\|} \\ &= \sigma \varphi_k^2 \|\mathbf{d}_k\|^2 \geq 0. \end{aligned}$$

Iz Leme 7.3.2 proizlazi da je niz $\{\|\mathbf{x}_{k+1} - \mathbf{x}_k\|\}$ nerastući i konvergentan, dakle ograničen. Takođe, niz $\{\mathbf{x}_k\}$ je ograničen i

$$\lim_{k \rightarrow \infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| = 0. \quad (7.43)$$

Na kraju, iz poslednje dve jednačine sledi (7.42). $\square$

Lema 7.3.4. *Pretpostavimo da je funkcija $F(\mathbf{x})$ Lipšic neprekidna i da je niz $\{\mathbf{x}_k\}$ generisan MSMDLPM metodom ograničen, tada postoji konstanta $M > 0$ takva da*

$$\|\mathbf{d}_k\| \leq M, \quad \forall k \in \mathbb{N} \cup \{0\}. \quad (7.44)$$

Dokaz. Za $k = 0$, iz $\mathbf{d}_0 = -F(\mathbf{x}_0)$ u (7.12) i (7.33), sledi

$$\|\mathbf{d}_0\| = \|-F(\mathbf{x}_0)\| \leq \omega. \quad (7.45)$$

Za $k \geq 1$, iz (7.12), (7.30) i $\mathbf{s}_{k-1} = \varphi_{k-1}\mathbf{d}_{k-1}$, sledi

$$\begin{aligned} \|\mathbf{d}_k\| &= \|-F(\mathbf{x}_k) + \beta_k^{\text{MSMDLPM}} \mathbf{d}_{k-1}\| \\ &= \left\| -F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{d}_{k-1} - t_k^{\text{MSMDLPM}} \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{d}_{k-1} \right\| \\ &= \left\| -F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\varphi_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \varphi_{k-1} \mathbf{d}_{k-1} - t_k^{\text{MSMDLPM}} \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\varphi_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \varphi_{k-1} \mathbf{d}_{k-1} \right\| \\ &= \left\| -F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{s}_{k-1} - t_k^{\text{MSMDLPM}} \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{s}_{k-1} \right\|. \end{aligned} \quad (7.46)$$

Da bismo dovršili dokaz leme, korišćićemo nejednačine (7.31), (7.32) i (7.33). Pored toga, na osnovu definicije za t_k^{MSMDLPM} u (7.29) razlikujemo dva slučaja.

(i): Ako je $t_k^{\text{MSMDLPM}} = \theta \|\mathbf{y}_{k-1}\|^2 / (\mathbf{s}_{k-1}^T \mathbf{y}_{k-1})$, onda se iz jednakosti (7.46) dobija

$$\begin{aligned} \|\mathbf{d}_k\| &= \left\| -F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{s}_{k-1} - \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \cdot \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{s}_{k-1} \right\| \\ &\leq \|F(\mathbf{x}_k)\| + \frac{\|F(\mathbf{x}_k)\| \|\mathbf{y}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \|\mathbf{s}_{k-1}\| + \theta \frac{\|\mathbf{y}_{k-1}\|^2}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \cdot \frac{\|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \|\mathbf{s}_{k-1}\| \\ &\leq \|F(\mathbf{x}_k)\| + \frac{\|F(\mathbf{x}_k)\| L \|\mathbf{s}_{k-1}\|}{c \|\mathbf{s}_{k-1}\|^2} \|\mathbf{s}_{k-1}\| + \theta \frac{L^2 \|\mathbf{s}_{k-1}\|^2}{c \|\mathbf{s}_{k-1}\|^2} \cdot \frac{\|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\|}{c \|\mathbf{s}_{k-1}\|^2} \|\mathbf{s}_{k-1}\| \\ &= \|F(\mathbf{x}_k)\| + \frac{L}{c} \|F(\mathbf{x}_k)\| + \theta \frac{L^2}{c^2} \|F(\mathbf{x}_k)\| \\ &\leq \left(1 + \frac{L}{c} + \theta \frac{L^2}{c^2} \right) \omega. \end{aligned} \quad (7.47)$$

(ii): Ako je

$$t_k^{\text{MSMDLPM}} = \frac{R_k(F) \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{(F(\mathbf{x}_k)^T \mathbf{s}_{k-1})^2}, \quad (7.48)$$

gde je $R_k(F)$ definisano kao u (7.26), onda se iz jednakosti (7.46) dobija se

$$\begin{aligned}
 \|\mathbf{d}_k\| &= \left\| -F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{s}_{k-1} \right. \\
 &\quad \left. - \frac{R_k(F) \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{(F(\mathbf{x}_k)^T \mathbf{s}_{k-1})^2} \cdot \frac{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{s}_{k-1} \right\| \\
 &= \left\| -F(\mathbf{x}_k) + \frac{F(\mathbf{x}_k)^T \mathbf{y}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{s}_{k-1} \right. \\
 &\quad \left. - \frac{R_k(F) \mathbf{s}_{k-1}^T \mathbf{y}_{k-1} + F(\mathbf{x}_k)^T \mathbf{y}_{k-1} F(\mathbf{x}_k)^T \mathbf{s}_{k-1}}{F(\mathbf{x}_k)^T \mathbf{s}_{k-1}} \cdot \frac{\mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \right\| \\
 &\leq \|F(\mathbf{x}_k)\| + \frac{\|F(\mathbf{x}_k)\| \|\mathbf{y}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \|\mathbf{s}_{k-1}\| \\
 &\quad + \frac{R_k(F) \|\mathbf{s}_{k-1}\| \|\mathbf{y}_{k-1}\| + \|F(\mathbf{x}_k)\| \|\mathbf{y}_{k-1}\| \|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\|}{\|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\|} \cdot \frac{\|\mathbf{s}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \\
 &= \|F(\mathbf{x}_k)\| + \frac{\|F(\mathbf{x}_k)\| \|\mathbf{y}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \|\mathbf{s}_{k-1}\| \\
 &\quad + \frac{(\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} - 1) \|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\| \|\mathbf{y}_{k-1}\| + \|\mathbf{y}_{k-1}\| \|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}.
 \end{aligned}$$

Dalje imamo da važi

$$\begin{aligned}
 \|\mathbf{d}_k\| &\leq \|F(\mathbf{x}_k)\| + \frac{\|F(\mathbf{x}_k)\| L \|\mathbf{s}_{k-1}\|}{c \|\mathbf{s}_{k-1}\|^2} \|\mathbf{s}_{k-1}\| \\
 &\quad + \frac{(\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} - 1) \|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\| L \|\mathbf{s}_{k-1}\| + L \|\mathbf{s}_{k-1}\| \|F(\mathbf{x}_k)\| \|\mathbf{s}_{k-1}\|}{c \|\mathbf{s}_{k-1}\|^2} \\
 &= \|F(\mathbf{x}_k)\| + \frac{L}{c} \|F(\mathbf{x}_k)\| + \frac{(\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} - 1) L \|F(\mathbf{x}_k)\| + L \|F(\mathbf{x}_k)\|}{c} \quad (7.49) \\
 &= \left(1 + \frac{L}{c} + \frac{\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} L}{c} \right) \|F(\mathbf{x}_k)\| \\
 &\leq \left(1 + \frac{L}{c} + \frac{\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} L}{c} \right) \omega.
 \end{aligned}$$

Ako je konstanta M na osnovu (7.45), (7.47) i (7.49) definisana sa

$$M := \max \left\{ \omega, \left(1 + \frac{L}{c} + \theta \frac{L^2}{c^2} \right) \omega, \left(1 + \frac{L}{c} + \frac{\psi_k(\gamma_k^{\text{MSMDLPM}})^{-1} L}{c} \right) \omega \right\},$$

onda nejednakost (7.44) važi za svako $k \in \mathbb{N} \cup \{0\}$. $\square$

Narednom teoremom biće uspostavljena globalna konvergencija MSMDLPM metode.

Teorema 7.3.1. *Neka su $\{\mathbf{x}_k\}$ i $\{\mathbf{z}_k\}$ nizovi generisani MSMDLPM metodom (Algoritam 7.2.1), onda važi*

$$\liminf_{k \rightarrow \infty} \|F(\mathbf{x}_k)\| = 0. \quad (7.50)$$

Dokaz. Ukoliko pretpostavimo da (7.50) ne važi, onda postoji konstanta $\nu > 0$ takva da

$$\|F(\mathbf{x}_k)\| \geq \nu, \quad \forall k \geq 0. \quad (7.51)$$

Prvo dokazujemo da je niz $\{\mathbf{d}_k\}$ ograničen datim uslovima. Na osnovu Cauchy-Schwarz-ove nejednakosti i (7.34) dobija se

$$\lambda \|F(\mathbf{x}_k)\|^2 \leq -F(\mathbf{x}_k)^T \mathbf{d}_k \leq \|F(\mathbf{x}_k)\| \|\mathbf{d}_k\|. \quad (7.52)$$

Iz nejednčina (7.52) i (7.51) dobija se

$$\|\mathbf{d}_k\| \geq \lambda \|F(\mathbf{x}_k)\| \geq \lambda \nu > 0, \quad \forall k \geq 0. \quad (7.53)$$

Prema tome, koristeći (7.42) i (7.53) dobija se

$$\lim_{k \rightarrow \infty} \varphi_k = 0. \quad (7.54)$$

Pretpostavimo da $r^{-1}\varphi_k$ ne zadovoljava uslov linijskog pretraživanja (7.10), odnosno da važi

$$-F(\mathbf{x}_k + r^{-1}\varphi_k \mathbf{d}_k)^T \mathbf{d}_k < \sigma r^{-1}\varphi_k \|F(\mathbf{x}_k + r^{-1}\varphi_k \mathbf{d}_k)\| \|\mathbf{d}_k\|^2.$$

Onda iz (7.34) i (7.33) imamo da važi

$$\lambda \|F(\mathbf{x}_k)\|^2 \leq -F(\mathbf{x}_k)^T \mathbf{d}_k \quad (7.55)$$

$$\begin{aligned} &= (F(\mathbf{x}_k + r^{-1}\varphi_k \mathbf{d}_k) - F(\mathbf{x}_k))^T \mathbf{d}_k - F(\mathbf{x}_k + r^{-1}\varphi_k \mathbf{d}_k)^T \mathbf{d}_k \\ &\leq L r^{-1}\varphi_k \|\mathbf{d}_k\|^2 + \sigma r^{-1}\varphi_k \|F(\mathbf{x}_k + r^{-1}\varphi_k \mathbf{d}_k)\| \|\mathbf{d}_k\|^2 \\ &\leq L r^{-1}\varphi_k \|\mathbf{d}_k\|^2 + \sigma r^{-1}\varphi_k \omega \|\mathbf{d}_k\|^2 \\ &\leq (L + \sigma \omega) r^{-1}\varphi_k \|\mathbf{d}_k\|^2. \end{aligned} \quad (7.56)$$

Iz prethodne nejednakosti, (7.51) i Leme 7.3.4, dobija se

$$\varphi_k \|\mathbf{d}_k\| \geq \frac{r \lambda \|F(\mathbf{x}_k)\|^2}{(L + \sigma \omega) \|\mathbf{d}_k\|} \geq \frac{r \lambda \nu^2}{(L + \sigma \omega) M}, \quad (7.57)$$

što je u suprotnosti sa (7.42). Prema tome, važi (7.50). $\square$

7.3.2 Analiza brzine konvergencije

Na osnovu Teoreme 7.3.1, niz $\{\mathbf{x}_k\}$ konvergira ka rešenju problema (7.1). Prema tome, pretpostavimo da $\mathbf{x}_k \rightarrow \mathbf{x}^*$ kada $k \rightarrow \infty$, gde $\mathbf{x}^*$ pripada skupu rešenja Ω problema (7.1). Da bi analizirali brzinu konvergencije MSMDLPM metode, potrebna je sledeća pretpostavka.

Pretpostavka 7.3.2. *Za svako $\mathbf{x}^* \in \Omega$, postoje pozitivne konstante m i δ takve da*

$$m \operatorname{dist}(\mathbf{x}, \Omega) \leq \|F(\mathbf{x})\|^2, \quad \forall \mathbf{x} \in \mathcal{N}(\mathbf{x}^*, \delta), \quad (7.58)$$

gde $\operatorname{dist}(\mathbf{x}, \Omega)$ označava rastojanje od $\mathbf{x}$ do skupa rešenja Ω i $\mathcal{N}(\mathbf{x}^*, \delta) := \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{x}^*\| \leq \delta\}$.

Dokaz Teoreme 7.3.2 je sličan dokazima u [103, 104].

Teorema 7.3.2. *Pretpostavimo da Pretpostavke 7.3.1 i 7.3.2 važe i da niz $\{\mathbf{x}_k\}$ generisan MSMDLPM metodom (Algoritmom 7.2.1) konvergira ka $\mathbf{x}^* \in \Omega$. Onda niz $\{\operatorname{dist}(\mathbf{x}_k, \Omega)\}$ konvergira Q -linearo ka 0, stoga čitav niz $\{\mathbf{x}_k\}$ R -linearo konvergira ka $\mathbf{x}^*$.*

Dokaz. Uzmite u obzir najbliže rešenje $\mathbf{x}_k$ definisano sa $\mathbf{v}_k := \arg \min\{\|\mathbf{x}_k - \mathbf{v}\| \mid \mathbf{v} \in \Omega\}$, odnosno

$$\|\mathbf{x}_k - \mathbf{v}_k\| = \operatorname{dist}(\mathbf{x}_k, \Omega).$$

Iz monotonosti za F , važi sledeće za $\mathbf{z}_k = \mathbf{x}_k + \varphi_k \mathbf{d}_k$:

$$(F(\mathbf{z}_k) - F(\mathbf{v}_k))^T(\mathbf{z}_k - \mathbf{v}_k) = F(\mathbf{z}_k)^T(\mathbf{z}_k - \mathbf{v}_k) \geq 0.$$

Onda iz (7.10) sledi

$$\begin{aligned} F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{v}_k) &\geq F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k) \\ &= F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{x}_k - \varphi_k \mathbf{d}_k) \\ &= -\varphi_k F(\mathbf{z}_k)^T \mathbf{d}_k \\ &\geq \sigma \varphi_k^2 \|F(\mathbf{z}_k)\| \|\mathbf{d}_k\|^2 \geq 0. \end{aligned} \quad (7.59)$$

Uzimajući (7.9) u obzir, sledi da važi

$$\begin{aligned} \|\mathbf{x}_{k+1} - \mathbf{v}_k\|^2 &= \left\| \mathbf{x}_k - \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} F(\mathbf{z}_k) - \mathbf{v}_k \right\|^2 = \left\| \mathbf{x}_k - \mathbf{v}_k - \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} F(\mathbf{z}_k) \right\|^2 \\ &= \|\mathbf{x}_k - \mathbf{v}_k\|^2 - 2 \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{v}_k) \cdot F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} \\ &\quad + \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k) \cdot F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2}. \end{aligned} \quad (7.60)$$

Iz prve nejednakosti (7.59) i (7.60) dobija se

$$\begin{aligned}
 \|\mathbf{x}_{k+1} - \mathbf{v}_k\|^2 &\leq \|\mathbf{x}_k - \mathbf{v}_k\|^2 - 2 \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k) \cdot F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} \\
 &\quad + \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k) \cdot F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} \\
 &= \|\mathbf{x}_k - \mathbf{v}_k\|^2 - \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k) \cdot F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2}. \tag{7.61}
 \end{aligned}$$

Iz definicije za $\text{dist}(\mathbf{x}_k, \Omega)$, gornjih nejednakosti (7.61) i (7.59) važi

$$\begin{aligned}
 \text{dist}(\mathbf{x}_{k+1}, \Omega)^2 &\leq \|\mathbf{x}_{k+1} - \mathbf{v}_k\|^2 \\
 &\leq \|\mathbf{x}_k - \mathbf{v}_k\|^2 - \frac{F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k) \cdot F(\mathbf{z}_k)^T(\mathbf{x}_k - \mathbf{z}_k)}{\|F(\mathbf{z}_k)\|^2} \\
 &\leq \text{dist}(\mathbf{x}_k, \Omega)^2 - \frac{\sigma \varphi_k^2 \|F(\mathbf{z}_k)\| \|\mathbf{d}_k\|^2 \cdot \sigma \varphi_k^2 \|F(\mathbf{z}_k)\| \|\mathbf{d}_k\|^2}{\|F(\mathbf{z}_k)\|^2} \\
 &= \text{dist}(\mathbf{x}_k, \Omega)^2 - \frac{\sigma^2 \varphi_k^4 \|F(\mathbf{z}_k)\|^2 \|\mathbf{d}_k\|^4}{\|F(\mathbf{z}_k)\|^2} \\
 &= \text{dist}(\mathbf{x}_k, \Omega)^2 - \sigma^2 \varphi_k^4 \|\mathbf{d}_k\|^4. \tag{7.62}
 \end{aligned}$$

Onda iz (7.53) i (7.58) sledi

$$\begin{aligned}
 \text{dist}(\mathbf{x}_{k+1}, \Omega)^2 &\leq \text{dist}(\mathbf{x}_k, \Omega)^2 - \sigma^2 \varphi_k^4 \lambda^4 \|F(\mathbf{x}_k)\|^4 \\
 &\leq \text{dist}(\mathbf{x}_k, \Omega)^2 - \sigma^2 \varphi_k^4 \lambda^4 m^2 \text{dist}(\mathbf{x}_k, \Omega)^2 \\
 &= (1 - \sigma^2 \varphi_k^4 \lambda^4 m^2) \text{dist}(\mathbf{x}_k, \Omega)^2. \tag{7.63}
 \end{aligned}$$

Poslednja nejednakost pokazuje da niz $\{\text{dist}(\mathbf{x}_k, \Omega)\}$ konvergira Q-linearano ka 0. Prema tome, niz $\{\mathbf{x}_k\}$ konvergira R-linearano ka $\mathbf{x}^*$. $\square$

7.4 Numerički rezultati i primena

Ovaj odeljak se sastoji iz dva pododeljka. U prvom pododeljku su dati numerički rezultati sa ciljem da se pokaže efikasnost predloženog MSMDLPM metoda (Algoritam 7.2.1) u rešavanju velikih sistema nelinearnih monotonihi jednačina u odnosu na neke varijante DLPM metode (Algoritma 7.1.1).

Primena MSMDLPM metode u restauraciji slika predstavljena je u drugom pododeljku ovog odeljka.

7.4.1 Numeričkih rezultati za MSMDLPM algoritam

Osnovni cilj u ovom odeljku je da se pokaže efikasnost MSMDLPM metoda (Algoritam 7.2.1) ne samo nad DLPM metodom (Algoritam 7.1.1), već i nad nekim njenim varijantama. Mnogi istraživači su posvetili pažnju na poboljšanju efikasnosti Dai–Liao metode definisanjem novih aproksimacija parametra t_k [19, 41, 73, 107]. Na osnovu postojećih istraživanja, koristićemo DLPM metod (Algoritam 7.1.1) kao polaznu tačku za kreiranje različitih varijanti DLPM algoritma na osnovu različitih vrednosti parametra t_k . Nazivi odgovarajućih algoritama definišu se na osnovu naziva konjugovano gradijentnog parametra β_k dodavanjem sufiksa PM (projektivni metod), jer je i sam naziv DLPM metode nastao odgovarajućom kombinacijom Dai–Liao metode i projektivnog metoda.

Ako se t_k u 6. koraku Algoritma 7.1.1 zameni sa:

- t_k iz (7.16), onda se novodobijeni algoritam označava sa DKPM;
- t_k iz (5.21), onda se novodobijeni algoritam označava sa MDLPM.

Za sve algoritme koristimo iste vrednosti konstanti $\sigma = 0.01$ i $r = 0.6$ u linijskom pretraživanju (7.10). Uslov zaustavljanja za sve algoritme je $\|F(\mathbf{x}_k)\| \leq 10^{-6}$, odnosno $\varepsilon = 10^{-6}$ ili kada algoritam potriši više od 1000 (sek.) CPU vremena na jednom test primeru. Pored navedenih konstanti, DLPM metod koristi konstante $p = 0.8$ i $q = -0.1$. U MSMDLPM i MDLPM metodama konstanta θ ima vrednost 0.26, dok konstante j i C u MDLPM metodi imaju iste vrednosti kao u [107], odnosno $j = \theta\|F(\mathbf{x}_{k-1})\|$ i $C = 1$.

Kodovi svih algoritama (metoda) koji su korišćeni u ovom odeljku napisani su u Matlab R2017a i izvršeni na laptop računaru sa Intel Core i3 2.0 GHz CPU-om, 8 GB RAM-a i Windows 10 operativnim sistemom. Tokom eksperimenta testirana je kolekcija od 11 test problema sa različitim izborom broja promenljivih $n \in \{1000, 2000, 3000, 5000, 6000, 8000, 10000, 15000, 30000, 60000\}$.

Test problemi su oblika $F(\mathbf{x}) = (F_1(\mathbf{x}), F_2(\mathbf{x}), \dots, F_n(\mathbf{x}))^T$, gde je $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, dok su početne tačke definisane za svaki test problem u okviru kolekcije od 11 test problema.

Problem 1. *Preuzet je iz [185] i definisan na sledeći način:*

$$\begin{aligned} F_1(\mathbf{x}) &= 2x_1 + 0.5h^2(x_1 + h)^3 - x_2, \\ F_i(\mathbf{x}) &= 2x_i + 0.5h^2(x_i + hi)^3 - x_{i-1} + x_{i+1}, \quad i = 2, 3, \dots, n-1, \\ F_n(\mathbf{x}) &= 2x_n + 0.5h^2(x_n + hn)^3 - x_{n-1}, \quad \text{gde je } h = \frac{1}{n+1}. \\ \text{Početna tačka: } \mathbf{x}_0 &= (h(h-1), h(2h-1), \dots, h(nh-1))^T. \end{aligned}$$

Problem 2. *Preuzet je iz [24] i definisan na sledeći način:*

$$\begin{aligned} F_1(\mathbf{x}) &= x_1 - e^{\cos(h(x_1+x_2))}, \\ F_i(\mathbf{x}) &= x_i - e^{\cos(h(x_{i-1}+x_i+x_{i+1}))}, \quad i = 2, 3, \dots, n-1, \\ F_n(\mathbf{x}) &= x_n - e^{\cos(h(x_{n-1}+x_n))}, \quad \text{gde je } h = \frac{1}{n+1}. \\ \text{Početna tačka: } \mathbf{x}_0 &= (1, 1, \dots, 1)^T. \end{aligned}$$

Problem 3. *Preuzet je iz [94] i definisan na sledeći način:*

$$\begin{aligned} F_i(\mathbf{x}) &= 2c(x_i - 1) + 4x_i \sum_{j=1}^n (x_j^2 - x_i), \quad i = 1, 2, \dots, n, \quad \text{gde je } c = 10^{-5}. \\ \text{Početna tačka: } \mathbf{x}_0 &= (0.9, 0.9, 0.9, \dots, 0.9)^T. \end{aligned}$$

Problem 4. Preuzet je iz [94] i definisan na sledeći način:

$$F_i(\mathbf{x}) = e^{x_i} - 2, \quad i = 1, 2, 3, \dots, n,$$
$$\text{Početna tačka: } \mathbf{x}_0 = (1, 1, \dots, 1).$$

Problem 5. Preuzet je iz [192] i definisan na sledeći način:

$$F_i(\mathbf{x}) = x_i - \sin(x_i), \quad i = 1, 2, 3, \dots, n.$$
$$\text{Početna tačka: } \mathbf{x}_0 = (1, 1, \dots, 1)^T.$$

Problem 6. Preuzet je iz [182, 93] i definisan na sledeći način:

$$F_i(\mathbf{x}) = x_i - \sin(|x_i|), \quad i = 1, 2, 3, \dots, n.$$
$$\text{Početna tačka: } \mathbf{x}_0 = (1, 1, \dots, 1)^T.$$

Problem 7. Preuzet je iz [2, 37] i definisan na sledeći način:

$$F_i(\mathbf{x}) = \frac{i}{n} e^{x_i} - 1, \quad i = 1, 2, 3, \dots, n,$$
$$\text{Početna tačka: } \mathbf{x}_0 = (1, 1, \dots, 1)^T.$$

Problem 8. Preuzet je iz [2] i definisan na sledeći način:

$$F_i(\mathbf{x}) = x_i - \sin(|x_i - 1|), \quad i = 1, 2, 3, \dots, n.$$
$$\text{Početna tačka: } \mathbf{x}_0 = (1, 1, \dots, 1)^T.$$

Problem 9. Preuzet je iz [95] i definisan na sledeći način:

$$F_i(\mathbf{x}) = x_i - \sin(|x_i| - 1), \quad i = 1, 2, 3, \dots, n.$$
$$\text{Početna tačka: } \mathbf{x}_0 = (1, 1, \dots, 1)^T.$$

Problem 10. Preuzet je iz [93] i definisan na sledeći način:

$$F_1(\mathbf{x}) = (3 - x_1)x_1 - 2x_2 + 1,$$
$$F_i(\mathbf{x}) = (3 - x_i)x_i - x_{i-1} + 2x_{i+1} + 1, \quad i = 2, 3, \dots, n-1,$$
$$F_n(\mathbf{x}) = (3 - x_n)x_n - x_{n-1} + 1.$$
$$\text{Početna tačka: } \mathbf{x}_0 = (-1, -1, \dots, -1)^T.$$

Problem 11. Preuzet je iz [16] i definisan na sledeći način:

$$F_1(\mathbf{x}) = e^{x_1} - 1,$$
$$F_i(\mathbf{x}) = e^{x_i} + x_{i-1} - 1, \quad i = 2, 3, \dots, n.$$
$$\text{Početna tačka: } \mathbf{x}_0 = (1, \frac{1}{2}, \frac{1}{3}, \dots, \frac{1}{n})^T.$$

Prilikom testiranja test problema praćeni su indikatori za svaki testirani algoritam: broj iteracija, broj evaluacija funkcije $F(\mathbf{x})$, CPU vreme i vrednost norme funkcije $F(\mathbf{x}^*)$ pri približnom rešenju $\mathbf{x}^*$. Tabele 7.1–7.4 prikazuju numeričke rezultate (broj iteracija, broj evaluacija funkcije $F(\mathbf{x})$, CPU vreme i normu funkcije $F(\mathbf{x})$) za DLPM, DKPM, MDLPM i MSMDLPM algoritam.

Neki direktni zaključci o efikasnosti određenih algoritama na određene klase problema nisu pouzdani. Na primer, MSMDLPM algoritam je najbolji po svim kriterijumima i za sve dimenzije test Problema 6. Ukoliko pogledate test Probleme 4 i 7 možete uočiti da su definisani sličnim sistemima jednačina, ali su rezultati po algoritmima različiti. Kod test Problema 4 MSMDLPM algoritam je najbolji po broju iteracija, dok je kod test Problema 7 DLPM algoritam najbolji po broju iteracija.

Tabela 7.1: Numerički rezultati za DLPM, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUts) i $\|F(\mathbf{x})\|$ za Probleme 1, 2 i 3.

Test problem	$\frac{n}{1000}$	DLPM		DKPM		MDLPM		MSMDLPM	
		NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $
Problem 1	1	25/187/0.094/9.85E-07	24/178/0.078/9.31E-07	25/126/0.078/8.44E-07	26/142/0.094/7.83E-07	26/142/0.094/7.83E-07	26/142/0.094/7.83E-07	26/142/0.094/7.83E-07	26/142/0.094/7.83E-07
	2	18/119/0.234/4.19E-07	22/168/0.344/9.47E-07	27/151/0.25/8.42E-07	28/161/0.313/7.80E-07	28/161/0.313/7.80E-07	28/161/0.313/7.80E-07	28/161/0.313/7.80E-07	28/161/0.313/7.80E-07
	3	20/152/0.469/6.72E-07	18/127/0.594/9.34E-07	26/138/0.375/4.88E-07	28/152/0.516/8.31E-07	28/152/0.516/8.31E-07	28/152/0.516/8.31E-07	28/152/0.516/8.31E-07	28/152/0.516/8.31E-07
	5	21/144/0.609/6.50E-07	23/175/0.734/7.20E-07	22/110/0.609/8.61E-07	23/120/0.563/9.55E-07	23/120/0.563/9.55E-07	23/120/0.563/9.55E-07	23/120/0.563/9.55E-07	23/120/0.563/9.55E-07
	6	21/148/0.641/7.58E-07	21/153/0.781/7.63E-07	22/123/0.469/5.22E-07	24/129/0.672/6.28E-07	24/129/0.672/6.28E-07	24/129/0.672/6.28E-07	24/129/0.672/6.28E-07	24/129/0.672/6.28E-07
	8	18/116/0.734/8.31E-07	18/135/0.891/9.97E-07	21/106/0.531/9.36E-07	26/135/0.953/7.85E-07	26/135/0.953/7.85E-07	26/135/0.953/7.85E-07	26/135/0.953/7.85E-07	26/135/0.953/7.85E-07
	10	18/124/1.078/9.68E-07	16/130/1.172/8.74E-07	22/108/0.875/5.99E-07	21/104/0.906/8.52E-07	21/104/0.906/8.52E-07	21/104/0.906/8.52E-07	21/104/0.906/8.52E-07	21/104/0.906/8.52E-07
	15	18/137/1.656/8.04E-07	18/146/1.813/8.44E-07	23/124/1.5/5.03E-07	22/110/1.406/4.95E-07	22/110/1.406/4.95E-07	22/110/1.406/4.95E-07	22/110/1.406/4.95E-07	22/110/1.406/4.95E-07
	30	16/107/2.172/8.10E-07	21/159/3.531/7.12E-07	21/110/2.391/7.73E-07	23/135/3.266/8.20E-07	23/135/3.266/8.20E-07	23/135/3.266/8.20E-07	23/135/3.266/8.20E-07	23/135/3.266/8.20E-07
	60	19/141/5.406/5.68E-07	16/115/5.063/5.62E-07	18/91/3.547/7.16E-07	18/93/3.984/7.25E-07	18/93/3.984/7.25E-07	18/93/3.984/7.25E-07	18/93/3.984/7.25E-07	18/93/3.984/7.25E-07
	1	12/43/0.047/1.96E-07	20/81/0.063/5.96E-07	31/95/0.047/6.20E-07	13/53/0.109/5.21E-07	13/53/0.109/5.21E-07	13/53/0.109/5.21E-07	13/53/0.109/5.21E-07	13/53/0.109/5.21E-07
	2	11/37/0.031/9.51E-07	20/81/0.016/8.45E-07	32/98/0.047/6.84E-07	13/53/0.047/7.37E-07	13/53/0.047/7.37E-07	13/53/0.047/7.37E-07	13/53/0.047/7.37E-07	13/53/0.047/7.37E-07
Problem 2	3	14/55/0.031/1.31E-07	21/85/0.047/4.14E-07	33/101/0.063/5.57E-07	13/53/0.344/9.02E-07	13/53/0.344/9.02E-07	13/53/0.344/9.02E-07	13/53/0.344/9.02E-07	13/53/0.344/9.02E-07
	5	10/33/0.031/2.99E-07	21/85/0.203/5.34E-07	33/101/0.203/9.96E-07	14/57/0.234/2.98E-07	14/57/0.234/2.98E-07	14/57/0.234/2.98E-07	14/57/0.234/2.98E-07	14/57/0.234/2.98E-07
	6	12/55/0.125/5.52E-07	21/85/0.156/5.85E-07	34/104/0.25/6.11E-07	14/57/0.188/3.27E-07	14/57/0.188/3.27E-07	14/57/0.188/3.27E-07	14/57/0.188/3.27E-07	14/57/0.188/3.27E-07
	8	12/44/0.141/7.60E-07	21/85/0.25/6.76E-07	34/104/0.297/8.44E-07	14/57/0.188/3.77E-07	14/57/0.188/3.77E-07	14/57/0.188/3.77E-07	14/57/0.188/3.77E-07	14/57/0.188/3.77E-07
	10	10/34/0.125/8.05E-07	22/90/0.359/4.84E-07	36/111/0.422/8.03E-07	14/58/0.281/7.81E-07	14/58/0.281/7.81E-07	14/58/0.281/7.81E-07	14/58/0.281/7.81E-07	14/58/0.281/7.81E-07
	15	10/34/0.172/7.53E-07	22/90/0.359/5.92E-07	37/114/0.453/6.44E-07	16/67/0.516/3.18E-07	16/67/0.516/3.18E-07	16/67/0.516/3.18E-07	16/67/0.516/3.18E-07	16/67/0.516/3.18E-07
	30	11/40/0.328/7.31E-07	23/96/0.547/6.57E-07	39/121/0.813/5.77E-07	17/74/0.594/4.21E-07	17/74/0.594/4.21E-07	17/74/0.594/4.21E-07	17/74/0.594/4.21E-07	17/74/0.594/4.21E-07
	60	12/46/0.609/8.27E-07	24/102/1.063/7.28E-07	40/124/1.422/6.34E-07	18/79/1.125/3.73E-07	18/79/1.125/3.73E-07	18/79/1.125/3.73E-07	18/79/1.125/3.73E-07	18/79/1.125/3.73E-07
	1	18/359/3.688/5.71E-07	20/400/4.109/5.40E-07	14/241/2.547/3.54E-07	14/241/3.813/3.54E-07	14/241/3.813/3.54E-07	14/241/3.813/3.54E-07	14/241/3.813/3.54E-07	14/241/3.813/3.54E-07
	2	19/400/12.625/8.39E-07	16/337/14.719/2.15E-07	8/148/4.359/1.56E-07	8/148/6.594/1.56E-07	8/148/6.594/1.56E-07	8/148/6.594/1.56E-07	8/148/6.594/1.56E-07	8/148/6.594/1.56E-07
	3	21/462/44.391/4.05E-07	17/374/36.547/6.32E-07	11/212/19.156/5.25E-07	11/212/22.906/5.25E-07	11/212/22.906/5.25E-07	11/212/22.906/5.25E-07	11/212/22.906/5.25E-07	11/212/22.906/5.25E-07
	5	21/483/107.594/8.71E-07	18/414/94.063/3.65E-07	12/243/54.625/1.35E-07	12/243/58.469/1.35E-07	12/243/58.469/1.35E-07	12/243/58.469/1.35E-07	12/243/58.469/1.35E-07	12/243/58.469/1.35E-07
Problem 3	6	18/415/122.859/2.15E-07	13/300/95.922/9.38E-07	22/464/134.578/4.06E-07	22/464/168.313/4.06E-07	22/464/168.313/4.06E-07	22/464/168.313/4.06E-07	22/464/168.313/4.06E-07	22/464/168.313/4.06E-07
	8	25/600/301.5/5.95E-07	21/504/266.438/4.73E-07	14/297/150.438/4.78E-07	14/297/186.188/4.78E-07	14/297/186.188/4.78E-07	14/297/186.188/4.78E-07	14/297/186.188/4.78E-07	14/297/186.188/4.78E-07
	10	18/433/332.672/4.62E-07	14/337/276.828/2.47E-07	22/486/373.938/8.73E-07	22/486/452.203/8.73E-07	22/486/452.203/8.73E-07	22/486/452.203/8.73E-07	22/486/452.203/8.73E-07	22/486/452.203/8.73E-07
	15	23/576/966.281/4.75E-07	19/476/868.797/2.37E-07	11/246/413.719/6.36E-08	11/246/669.719/6.36E-08	11/246/669.719/6.36E-08	11/246/669.719/6.36E-08	11/246/669.719/6.36E-08	11/246/669.719/6.36E-08
	30	*/*/1019.469/*	*/*/1013.516/*	*/*/1020.922/*	*/*/1025.734/*	*/*/1025.734/*	*/*/1025.734/*	*/*/1025.734/*	*/*/1025.734/*
	60	*/*/1051.016/*	*/*/1065.578/*	*/*/1052.219/*	*/*/1087.141/*	*/*/1087.141/*	*/*/1087.141/*	*/*/1087.141/*	*/*/1087.141/*

Tabela 7.2: Numerički rezultati za DLP, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUts) i $\|F(\mathbf{x})\|$ za Probleme 4, 5 i 6.

Test problem	$\frac{n}{1000}$	DLP	DKPM	MDLPM	MSMDLPM
		NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $	NI/NFE/CPUts/ $\ F(\mathbf{x})\ $
Problem 4	1	16/81/0.047/4.50E-07	13/66/0.031/6.49E-07	16/51/0.031/3.99E-07	12/50/0.422/8.80E-07
	2	16/81/0/6.36E-07	13/66/0/9.18E-07	16/51/0.047/6.58E-07	13/54/0.047/3.19E-07
	3	16/81/0.047/7.79E-07	14/71/0.047/3.15E-07	16/51/0.031/9.31E-07	13/54/0.016/3.90E-07
	5	17/86/0.078/3.54E-07	14/71/0.047/4.07E-07	17/54/0.047/4.29E-07	13/54/0.078/5.04E-07
	6	17/86/0.094/3.88E-07	14/71/0.031/4.45E-07	17/54/0.109/4.87E-07	13/54/0.063/5.52E-07
	8	17/86/0.094/4.48E-07	14/71/0.109/5.14E-07	17/54/0.094/6.03E-07	13/54/0.109/6.37E-07
	10	17/86/0.219/5.00E-07	14/71/0.234/5.75E-07	17/54/0.094/7.21E-07	13/54/0.109/7.13E-07
	15	17/86/0.375/6.13E-07	14/71/0.25/7.04E-07	18/57/0.281/3.35E-07	13/54/0.094/8.73E-07
	30	17/86/0.563/8.67E-07	14/71/0.5/9.96E-07	18/57/0.359/5.34E-07	14/58/0.234/3.16E-07
	60	18/91/0.703/4.31E-07	15/76/0.734/3.94E-07	18/57/0.516/9.46E-07	14/58/0.484/4.47E-07
Problem 5	1	100942/302827/16.297/1.00E-06	90847/272542/11.063/1.00E-06	90852/272557/13.484/1.00E-06	19/58/0.031/6.74E-07
	2	127181/381544/25.547/1.00E-06	114462/343387/18.766/1.00E-06	114467/343402/23.188/1.00E-06	19/58/0.047/9.54E-07
	3	145587/436762/108.047/1.00E-06	131028/393085/73.734/1.00E-06	131033/393100/93.688/1.00E-06	20/61/0.047/5.02E-07
	5	172615/517846/343.328/1.00E-06	155352/466057/254.563/1.00E-06	155358/466075/308.219/1.00E-06	20/61/0.094/6.49E-07
	6	183431/550294/423.453/1.00E-06	165087/495262/314.844/1.00E-06	165093/495280/361.984/1.00E-06	20/61/0.063/7.11E-07
	8	201893/605680/549.453/1.00E-06	181703/545110/405.688/1.00E-06	181708/545125/456.844/1.00E-06	20/61/0.156/8.20E-07
	10	217483/652450/775.625/1.00E-06	195734/587203/591.344/1.00E-06	195740/587221/676.703/1.00E-06	20/61/0.109/9.17E-07
	15	*/*/1000.047/*	224061/672184/957.266/1.00E-06	*/*/1000.016/*	21/64/0.172/4.83E-07
	30	*/*/1000.047/*	*/*/1000.031/*	*/*/1000.047/*	21/64/0.297/6.83E-07
	60	*/*/1000.016/*	*/*/1000.047/*	*/*/1000.016/*	21/64/0.703/9.67E-07
Problem 6	1	100942/302827/13.891/1.00E-06	90847/272542/10/1.00E-06	90852/272557/13.141/1.00E-06	19/58/0.047/6.74E-07
	2	127181/381544/26.141/1.00E-06	114462/343387/18.891/1.00E-06	114467/343402/25.844/1.00E-06	19/58/0.047/9.54E-07
	3	145587/436762/110.766/1.00E-06	131028/393085/79.094/1.00E-06	131033/393100/94.891/1.00E-06	20/61/0.063/5.02E-07
	5	172615/517846/347.297/1.00E-06	155352/466057/249.891/1.00E-06	155358/466075/299.641/1.00E-06	20/61/0.125/6.49E-07
	6	183431/550294/422.656/1.00E-06	165087/495262/315.188/1.00E-06	165093/495280/359.25/1.00E-06	20/61/0.078/7.11E-07
	8	201893/605680/568.141/1.00E-06	181703/545110/403.484/1.00E-06	181708/545125/457.016/1.00E-06	20/61/0.109/8.20E-07
	10	217483/652450/786.828/1.00E-06	195734/587203/586.297/1.00E-06	195740/587221/679.641/1.00E-06	20/61/0.109/9.17E-07
	15	*/*/1000.016/*	224061/672184/956.953/1.00E-06	*/*/1000.047/*	21/64/0.219/4.83E-07
	30	*/*/1000.031/*	*/*/1000.016/*	*/*/1000.016/*	21/64/0.328/6.83E-07
	60	*/*/1000.016/*	*/*/1000.016/*	*/*/1000.016/*	21/64/0.703/9.67E-07

Tabela 7.3: Numerički rezultati za DLP, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUs) i $\|F(\mathbf{x})\|$ za Probleme 7, 8 i 9.

Test problem	$\frac{n}{1000}$	DLP		DKPM		MDLPM		MSMDLPM	
		NI/NFE/CPUs/ $\ F(\mathbf{x})\ $	NI/NFE/CPUs/ $\ F(\mathbf{x})\ $	NI/NFE/CPUs/ $\ F(\mathbf{x})\ $	NI/NFE/CPUs/ $\ F(\mathbf{x})\ $	NI/NFE/CPUs/ $\ F(\mathbf{x})\ $	NI/NFE/CPUs/ $\ F(\mathbf{x})\ $		
Problem 7	1	18/82/0.031/1.74E-07	26/122/0.047/4.31E-07	31/99/0.063/7.68E-07	21/91/0.031/5.98E-07				
	2	18/84/0.031/8.31E-07	24/117/0.016/5.80E-07	33/105/0.047/9.42E-07	25/107/0.047/9.18E-07				
	3	24/140/0.031/4.04E-07	23/99/0.031/9.25E-07	32/107/0.109/9.31E-07	28/119/0.094/7.00E-07				
	5	18/75/0.063/8.97E-07	24/103/0.125/5.38E-07	34/105/0.156/6.99E-07	29/123/0.234/9.83E-07				
	6	21/96/0.156/4.92E-07	26/115/0.188/4.22E-07	34/105/0.203/9.65E-07	32/135/0.188/5.97E-07				
	8	21/90/0.141/3.48E-07	26/116/0.141/4.20E-07	35/108/0.219/7.70E-07	21/92/0.141/3.94E-07				
	10	23/106/0.25/9.07E-07	23/95/0.25/9.56E-07	38/117/0.328/5.67E-07	32/154/0.453/4.80E-07				
	15	19/81/0.344/7.42E-07	27/136/0.609/8.12E-07	39/124/0.406/6.48E-07	36/174/0.641/9.24E-07				
	30	23/99/0.594/7.77E-07	27/134/0.719/9.85E-07	41/156/0.859/8.48E-07	32/148/0.719/7.58E-07				
	60	24/120/0.969/2.42E-07	25/120/1.078/8.54E-07	42/148/1.109/7.90E-07	31/136/0.953/5.95E-07				
Problem 8	1	19/96/0.031/4.34E-07	16/81/0.047/4.21E-07	18/57/0.016/5.33E-07	13/54/0.016/3.59E-07				
	2	19/96/0.047/6.14E-07	16/81/0/5.96E-07	18/57/0.047/9.13E-07	13/54/0.047/5.08E-07				
	3	19/96/0/7.52E-07	16/81/0.031/7.30E-07	19/60/0.063/4.32E-07	13/54/0.031/6.22E-07				
	5	19/96/0.031/9.71E-07	16/81/0.063/9.42E-07	19/60/0.063/6.80E-07	13/54/0.031/8.03E-07				
	6	20/101/0.063/4.18E-07	17/86/0.094/3.37E-07	19/60/0.125/7.72E-07	13/54/0.047/8.80E-07				
	8	20/101/0.172/4.83E-07	17/86/0.094/3.89E-07	19/60/0.125/9.44E-07	14/58/0.094/2.60E-07				
	10	20/101/0.188/5.40E-07	17/86/0.188/4.34E-07	20/63/0.125/3.88E-07	14/58/0.109/2.91E-07				
	15	20/101/0.313/6.62E-07	17/86/0.234/5.32E-07	20/63/0.203/5.43E-07	14/58/0.188/3.56E-07				
	30	20/101/0.5/9.36E-07	17/86/0.484/7.52E-07	20/63/0.375/9.33E-07	14/58/0.359/5.04E-07				
	60	21/106/0.672/5.21E-07	18/91/0.594/3.47E-07	21/66/0.672/5.68E-07	14/58/0.594/7.12E-07				
Problem 9	1	18/88/0.031/6.82E-07	17/84/0.094/5.96E-07	20/61/0.031/5.37E-07	13/56/0.016/6.80E-07				
	2	18/88/0/9.65E-07	17/84/0.031/8.42E-07	20/61/0.031/9.32E-07	13/56/0.031/9.61E-07				
	3	19/93/0.047/4.65E-07	18/89/0/3.36E-07	21/64/0.063/4.43E-07	14/60/0/3.01E-07				
	5	19/93/0.047/6.00E-07	18/89/0.047/4.34E-07	21/64/0.047/7.02E-07	14/60/0.078/3.89E-07				
	6	19/93/0.031/6.57E-07	18/89/0.047/4.76E-07	21/64/0.094/7.95E-07	14/60/0.063/4.26E-07				
	8	19/93/0.109/7.59E-07	18/89/0.109/5.49E-07	21/64/0.109/9.73E-07	14/60/0.109/4.92E-07				
	10	19/93/0.172/8.49E-07	18/89/0.125/6.14E-07	22/67/0.188/4.01E-07	14/60/0.172/5.50E-07				
	15	20/98/0.25/5.23E-07	19/94/0.313/4.34E-07	24/74/0.297/3.90E-07	15/68/0.141/4.19E-07				
	30	18/88/0.438/4.16E-07	20/100/0.5/3.91E-07	25/78/0.391/6.58E-07	16/80/0.578/5.05E-07				
	60	21/105/0.625/7.74E-07	21/106/0.703/4.21E-07	26/81/0.547/3.80E-07	17/89/0.922/2.90E-07				

Tabela 7.4: Numerički rezultati za DLPm, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUts) i $\|F(\mathbf{x})\|$ za Probleme 10 i 11.

Test problem	$\frac{n}{1000}$	DLPm		DKPM		MDLPM		MSMDLPM	
		NI/NFE/CPUts	$\ F(\mathbf{x})\ $	NI/NFE/CPUts	$\ F(\mathbf{x})\ $	NI/NFE/CPUts	$\ F(\mathbf{x})\ $	NI/NFE/CPUts	$\ F(\mathbf{x})\ $
Problem 10	1	40/329/0.188/8.79E-07		39/342/0.141/7.53E-07		44/267/0.047/9.41E-07		43/263/0.219/6.77E-07	
	2	32/258/0.031/7.56E-07		26/205/0.047/8.29E-07		44/266/0.016/9.83E-07		47/283/0.078/7.78E-07	
	3	38/315/0.078/9.72E-07		38/308/0.094/7.17E-07		47/289/0.094/6.38E-07		46/280/0.141/7.47E-07	
	5	37/306/0.297/6.04E-07		35/289/0.234/5.64E-07		44/277/0.234/9.97E-07		44/277/0.234/8.28E-07	
	6	31/239/0.203/6.61E-07		40/339/0.297/5.88E-07		46/281/0.359/9.61E-07		49/300/0.422/8.83E-07	
	8	35/287/0.266/8.97E-07		23/170/0.188/8.47E-07		44/277/0.281/8.91E-07		47/298/0.375/4.80E-07	
	10	35/284/0.313/7.00E-07		23/163/0.141/7.96E-07		47/288/0.359/5.95E-07		47/287/0.375/9.76E-07	
	15	36/286/0.391/5.91E-07		38/319/0.375/8.21E-07		51/337/0.5/8.97E-07		49/383/0.609/9.87E-07	
	30	31/236/0.703/9.42E-07		*/1000.047/*		46/285/0.75/7.13E-07		48/315/0.797/6.85E-07	
	60	39/309/1.422/6.37E-07		38/309/1.125/9.80E-07		45/279/1.328/9.33E-07		45/279/1.219/9.33E-07	
Problem 11	1	2638/14402/1.359/9.59E-07		2690/15599/1.469/9.28E-07		3344/11131/1.297/9.64E-07		3097/15991/1.922/8.17E-07	
	2	4987/26711/2.141/9.80E-07		5005/28551/2.094/9.66E-07		6346/20940/1.859/9.96E-07		5992/31017/2.297/9.31E-07	
	3	7230/38616/13.906/9.89E-07		7245/40550/10.547/9.42E-07		9211/30351/9.141/9.38E-07		8795/45158/13.625/9.69E-07	
	5	11653/61904/39.375/9.83E-07		11613/64102/36.656/9.99E-07		15101/49400/38.047/9.84E-07		14446/73479/42.875/9.83E-07	
	6	13798/72574/48.125/9.80E-07		13829/75384/49.078/9.97E-07		17918/58509/49.594/9.98E-07		17268/87568/63.719/9.01E-07	
	8	18148/95128/82.891/9.93E-07		18086/97930/76.141/9.97E-07		23717/77363/86.859/9.65E-07		22835/116432/116.906/9.91E-07	
	10	22506/117909/146.047/9.99E-07		22414/120780/162.375/9.74E-07		29619/96400/146.906/9.93E-07		28346/143060/226.703/8.95E-07	
	15	33302/174933/296.563/9.97E-07		33084/177466/338.516/1.00E-06		43995/143002/271.984/9.68E-07		42144/212121/423.672/9.73E-07	
	30	*/1000.047/*		*/1000.031/*		88103/285526/951.375/9.86E-07		*/1000.016/*	
	60	*/1000.016/*		*/1000.016/*		*/1000.031/*		*/1000.047/*	

Analiza numeričkih rezultata datih u Tabelama 7.1–7.4 otkriva da MSMDLPM algoritam uspešno rešava 96.36% problema (DLPM – 90.91%, DKPM – 91.82%, MDLPM – 91.82%). Međutim, 3 (2.73%) test problema nije uspeo da reši nijedan od testiranih algoritama. Pored ovih zaključaka u Tabeli 7.5 data je dodatna analiza numeričkih rezultata iz Tabela 7.1–7.4. U Tabeli 7.5 dat je broj test primera (procenat) u kojima je svaki od algoritama (DLPM, DKPM, MDLPM i MSMDLPM) postigao najbolje rezultate u odnosu na broj iteracija, broj evaluacija funkcije $F(\mathbf{x})$ i CPU vreme. Kategorija "Neopredeljen" u Tabela 7.5 prikazuje broj test primera (procenat) u kojima su dva ili više algoritma postigla najbolje rezultate u pogledu broja iteracija, broja evaluacija funkcije $F(\mathbf{x})$ i CPU vremena.

Tabela 7.5: Analiza rezultata iz Tabela 7.1–7.4 koja prikazuje broj test primera (procenat) u kojima je posmatrani algoritam postigao minimalnu vrednost za broj iteracija, broj evaluacija funkcije $F(\mathbf{x})$ i CPU vreme.

Metod	Broj iteracija	Procenat	Br. evaluacija funkcije	Procenat	CPU vreme	Procenat
DLPM	25	22.73%	21	19.09%	22	20.00%
DKPM	17	15.45%	8	7.27%	13	11.82%
MDLPM	1	0.91%	20	18.18%	21	19.09%
MSMDLPM	49	44.55%	45	40.91%	33	30.00%
Neopredeljen	15	13.64%	13	11.82%	18	16.36%

Iz Tabele 7.5 se vidi da je MSMDLPM algoritam rešio 44.55% (49 od 110) testiranih problema u sprovedenim eksperimentima sa najmanjim brojem iteracija u poređenju sa DLPM, DKPM i MDLPM algoritmima, koji su zabeležili 22.73% (25 od 110), 15.45% (17 od 110) i 0.91% (1 od 110) respektivno. Rezultati dati u vrsti "Neopredeljen" u Tabeli 7.5 pokazuju da je 13.64% (15 od 110) problema rešeno sa istim minimalnim brojem iteracija od strane 2, 3 ili sva 4 algoritma uključena u eksperimentu.

U Tabeli 7.5, takođe se vidi da je MSMDLPM algoritam rešio 40.91% (45 od 110) testiranih problema sa minimalnim brojem evaluacija funkcije $F(\mathbf{x})$ u poređenju sa DLPM, DKPM i MDLPM algoritmima, koji su zabeležili 19.09% (21 od 110), 7.27% (8 od 110) i 18.18% (20 od 110) respektivno. Osim toga, iz vrste "Neopredeljen" u Tabele 7.5 može se uočiti da je 11.82% (13 od 110) problema rešeno sa istim minimalnim brojem evaluacija funkcije $F(\mathbf{x})$ od strane 2, 3 ili sva 4 algoritma uključena u eksperimentu.

Konačno, vidi se iz analiziranih numeričkih rezultata u Tabeli 7.5 da je MSMDLPM algoritam rešio 30.00% (33 od 110) testiranih problema za najmanje CPU vreme u odnosu na DLPM, DKPM i MDLPM algoritme, koji su rešili 20.00% (22 od 110), 11.82% (13 od 110) i 19.09% (18 od 110) respektivno. Takođe, u Tabeli 7.5 se vidi da je 16.36% (18 od 110) problema rešeno sa jednakim CPU vremenom od strane 2 ili više algoritma.

Analiza numeričkih rezultata pokazuje da MSMDLPM Algoritam 7.2.1 postiže bolje rezultate od bazičnog DLPM Algoritma 7.1.1 i modifikovanih varijanti DLPM algoritama.

Samim tim dokazano je da način na koji je definisan MSMDLPM Algoritam, odnosno upotreba parametra ubrzanja $\gamma_k^{\text{MSMDLPM}}$ i dodatnog parametra ψ_k utiče na dominantnost MSMDLPM Algoritma u numeričkim rezultatima.

Za dodatna upoređivanja numeričkih rezultata korišćeni su profili performansi predloženi od strane Dolan-a i Moré-a u [58] za broj iteracija, broj evaluacija funkcije i CPU vreme. Ako je sa n_y označen broj algoritama (metoda) u skupu algoritama $\mathcal{A}$, a sa n_x broj test problema u skupu test problema $\mathcal{P}$, onda je mera performansi algoritma y na test problemu x definisana sa $t_{x,y}$. U našem slučaju, $t_{x,y}$ označava broj iteracija ili broj evaluacija funkcije ili CPU vreme potrebno da se reši problem x od strane algoritma y .

Za bilo koji par (x, y) problema $x \in \mathcal{P}$ i algoritma $y \in \mathcal{A}$, količnik profila performansi je definisan sa

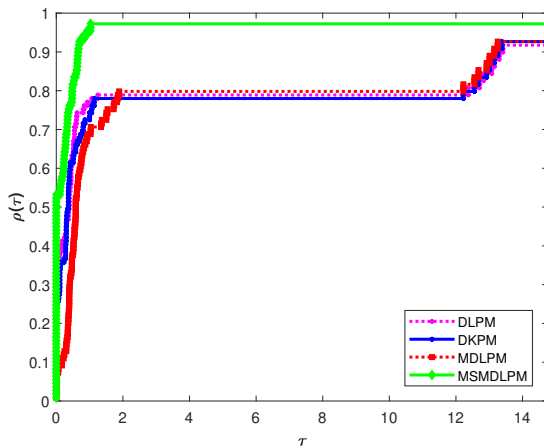
$$r_{x,y} = \frac{t_{x,y}}{\min\{t_{x,y} : y \in \mathcal{A}\}},$$

gde $t_{x,y}$ označava meru performansi i upoređuje se sa najboljom merom performansi koja je postignuta nekim od algoritama iz skupa $\mathcal{A}$ za posmatrani problem x .

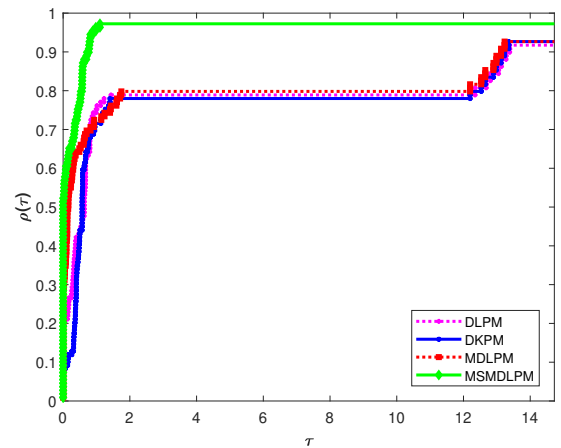
Profil performansi je definisan pomoću

$$\rho_y(\tau) = \frac{1}{n_x} \text{size}\{x \in \mathcal{P} : \log_2 r_{x,y} \leq \tau\},$$

gde je $\rho_y(\tau)$ verovatnoća da je količnik performansi $r_{x,y}$ algoritma $y \in \mathcal{A}$ unutar faktora $\tau \in \mathbb{R}$ najboljeg mogućeg količnika. Funkcija $\rho_y(\tau)$ se naziva kumulativna funkcija raspodele količnika performansi $r_{x,y}$. U graficima gornja kriva odgovara vrednostima algoritma koji pokazuje najbolje performanse u odnosu na odabrani profil performansi.



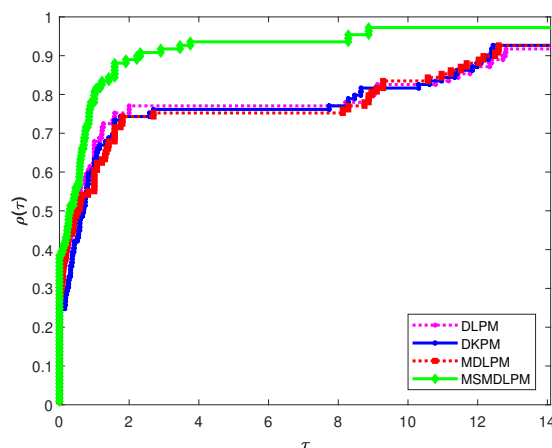
(a) Bazirani na broju iteracija.



(b) Bazirani na broju evaluacija funkcije $F(\mathbf{x})$.

Slika 7.1: Profili performansi za DLPM, DKPM, MDLPM i MSMDLPM metod.

Profili performansi za DLPM, DKPM, MDLPM i MSMDLPM algoritme su kreirani na osnovu numeričkih rezultata datih u Tabelama 7.1–7.4 i prikazani na Slikama 7.1 i 7.2.



Slika 7.2: Profili performansi bazirani na CPU vremenu za DLPM, DKPM, MDLPM i MSMDLPM metod.

Lako se uočava da je MSMDLPM algoritam najefikasniji u pogledu sva tri indikatora (broj iteracija, broj evaluacija funkcije $F(\mathbf{x})$ i CPU vreme). MSMDLPM algoritam je uspešno rešio 53.21% (DLPM – 29.36%, DKPM – 22.02%, MDLPM – 7.34%) problema sa najmanjim brojem iteracija, 52.29% (DLPM – 20.18%, DKPM – 8.26%, MDLPM – 29.36%) problema sa najmanjim brojem evaluacija funkcije $F(\mathbf{x})$ i 38.53% (DLPM – 29.36%, DKPM – 23.85%, MDLPM – 27.52%) problema u najkraćem CPU vremenu.

Treba naglasiti da ukoliko su dva ili više algoritma završila test problem sa istim brojem iteracija, ili istim brojem evaluacija funkcije $F(\mathbf{x})$, ili sa istim CPU vremenom, onda su svi oni pobednici za dati test problem.

Analizom rezultata prikazanih u Tabelama 7.1–7.4 i na Slikama 7.1 i 7.2, možemo zaključiti da je MSMDLPM algoritam postigao najbolje rezultate. Ovo zapažanje nas dovodi do konačnog zaključka da je MSMDLPM algoritam najefikasniji od posmatranih algoritama za rešavanje sistema nelinearnih monotonih jednačina u pogledu sva tri posmatrana indikatora: broja iteracija, broja evaluacija funkcije $F(\mathbf{x})$ i utrošeno CPU vreme.

7.4.2 Primena MSMDLPM algoritma na probleme restauracija slike

Problem restauracije slike igra važnu ulogu u medicinskim naukama, biološkom inženjerstvu i drugim oblastima nauke i inženjerstva [21, 31].

U ovom odeljku, MSMDLPM algoritam se primenjuje za rešavanje problema koji proizilaze iz komprimovanog očitavanja signala (*eng. compressive sensing*), posebno problema sa uklanjanjem zamućenja slike (*eng. image deblurring*). Naš glavni cilj u primeni MSMDLPM algoritma na problem smanjenja šuma (*eng. noise reduction*) i uklanjanja zamućenja slike je da pronađemo sliku koja odgovara originalnoj slici što je više moguće. Model za ovaj

problem, odnosno za postizanje našeg cilja, definisan je sledećim sistemom:

$$\mathbf{b} = L\mathbf{x} + \mathbf{y} \quad (7.64)$$

gde $\mathbf{b} \in \mathbb{R}^m$ predstavlja posmatranu sliku, L je linearno preslikavanje ($m \times n$ matrica zamućenja (*eng. blurring matrix*)), $\mathbf{x} \in \mathbb{R}^n$ je nepoznata slika i $\mathbf{y} \in \mathbb{R}^m$ je šum. Dobro je poznato da se regularizacione metode (*eng. regularization method*) koriste u problemima restauracije slike. Jedan od moćnijih alata za uklanjanje (smanjenje) šuma iz slike je ℓ_1 -regularizacija:

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) := \frac{1}{2} \|L\mathbf{x} - \mathbf{b}\|_2^2 + \mu \|\mathbf{x}\|_1, \quad (7.65)$$

gde $\|\cdot\|_2$ označava Evklidovu normu, μ je pozitivni parametar regularizacije, a $\|\cdot\|_1$ je ℓ_1 -regularizacioni član. Dokaz da je monotona jednačina (7.1) ekvivalentna (7.65) može se naći u radovima [79, 80, 173, 174].

Da bi se demonstrirala efikasnost MSMDLPM algoritma u restauraciji određenih zamućenja na slikama upoređen je sa Algoritmom 1 iz [3] i jednim standardnim CGD algoritmom [174] koji je efikasniji od SGCS algoritma [173]. Prilikom implementacije MSMDLPM algoritma u eksperimentima su dodeljenje sledeće vrednosti konstantama $\sigma = 10^{-4}$, $r = 0.5$ i $\varepsilon = 10^{-3}$. Parametri za Algoritam 1 i CGD algoritam imaju iste vrednosti kao u radovima [3] i [174].

Da bismo imali korektno upoređivanje algoritama, svaki algoritam je pokrenut iz iste početne tačke $\mathbf{x}_0 = A^T \mathbf{b}$ i zaustavljen ispunjenjem uslova

$$\frac{\|\mathbf{x}_k - \mathbf{x}_{k-1}\|}{\|\mathbf{x}_{k-1}\|} < 10^{-3} \quad \text{ili} \quad \frac{\|f(\mathbf{x}_k) - f(\mathbf{x}_{k-1})\|}{\|f(\mathbf{x}_{k-1})\|} < 10^{-3}, \quad (7.66)$$

gde je

$$f(\mathbf{x}_k) = \frac{1}{2} \|L\mathbf{x}_k - \mathbf{b}\|_2^2 + \mu \|\mathbf{x}_k\|_1.$$

U eksperimentu je testirano pet slika u boji različitih veličina (prva kolona na Slikama 7.3 i 7.4). Ove slike su izobličene (zamućene) korišćenjem Gausovog šuma sa standardnom devijacijom od 3.6 (druga kolona na Slikama 7.3 i 7.4).

Za merenje performansi i kvaliteta restauracije slika za svaki testirani algoritam korišćene su sledeće metrike: broj iteracija, CPU vreme, indeks strukturne sličnosti (*eng. structural similarity index*, SSIM) [167], odnos signal-šum (*eng. signal-to-noise ratio*, SNR) i maksimalni odnos signal-šum (*eng. peak signal-to-noise ratio*, PSNR) [27].

Indeks strukturne sličnosti se koristi za merenje sličnosti između dve slike, uzimajući u obzir promene u osvetljenju, kontrastu i strukturi, što ga čini korisnim alatom za procenu kvaliteta slike. Odnos signala i šuma se koristi za opisivanje proporcije između nivoa željenog signala i nivoa pozadinskog šuma, često izražen u decibelima (dB). Visok odnos signal-šum ukazuje na čistiji signal sa manje smetnji. Maksimalni odnos signala i šuma se koristi za merenje kvaliteta rekonstruisanih ili komprimovanih slika u poređenju sa originalnim

slikama. Izražava se u decibelima (dB) i veće vrednosti označavaju bolji kvalitet slike sa manje izobličenja uzrokovanih šumom.

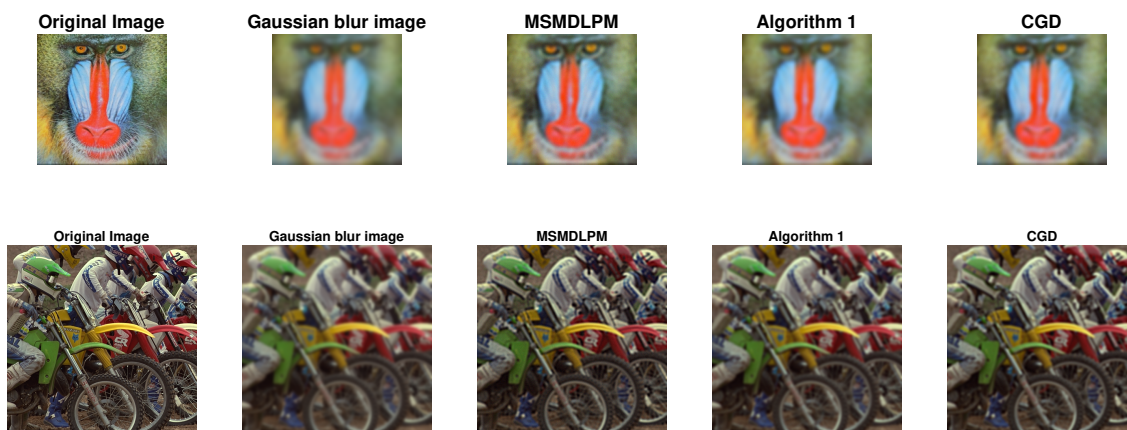
Tabela 7.6: Performanse MSMDLPM, Algoritma 1 i CGD algoritma pri uklanjanju zamućenja na slikama.

Metod	Indikator	Naziv slike (Dimeznizja)				
		Lena (512x512)	Barbara (720x576)	Yacht (512x480)	Baboon (200x200)	Motobikes (494x494)
MSMDLPM	Broj iteracija	17	18	24	18	27
	CPU vreme	8.984	15.203	9.688	1.859	12.125
	SSIM	0.956	0.707	0.753	0.704	0.633
	SNR	22.346	16.731	17.528	17.657	12.502
	PSNR	27.802	23.273	23.968	23.243	21.240
Algoritam1	Broj iteracija	25	43	45	64	61
	CPU vreme	11.047	28.250	15.266	4.500	20.125
	SSIM	0.941	0.676	0.705	0.670	0.566
	SNR	19.907	15.743	15.577	16.831	11.183
	PSNR	25.366	22.347	22.083	22.439	20.078
CGD	Broj iteracija	33	43	47	56	57
	CPU vreme	14.453	31.453	16.031	4.297	21.266
	SSIM	0.949	0.693	0.736	0.688	0.600
	SNR	21.172	16.243	16.431	17.272	11.788
	PSNR	26.613	22.808	22.893	22.857	20.600



Slika 7.3: Originalne slike (prva kolona), zamućene slike (druga kolona), restaurirane slike MSMDLPM algoritmom (treća kolona), Algoritmom 1 (četvrta kolona) i CGD algoritmom (poslednja kolona).

Eksperimentalni rezultati generisani pomoću MSMDLPM algoritma, Algoritma 1 i CGD algoritma predstavljeni su u Tabeli 7.6. Na osnovu numeričkih rezultat u Tabeli 7.6 lako se zaključuje da su za sve test slike restaurirane slike pomoću MSMDLPM algoritma bliže originalu od onih koje su restaurirane Algoritmom 1 i CGD algoritmom. MSMDLPM algoritam ima manje vrednosti u poređenju sa druga dva algoritma kada se upoređuju broj iteracija i CPU vreme, dok ima veće vrednosti kada se upoređuju metrike koje mere kvalitet restauriranih slika (SSIM, SNR i PSNR).



Slika 7.4: Originalne slike (prva kolona), zamućene slike (druga kolona), restaurirane slike MSMDLPM algoritmom (treća kolona), Algoritmom 1 (četvrta kolona) i CGD algoritmom (poslednja kolona).

Slike 7.3 i 7.4 su generisane da prikažu rezultate restauracije različitih slika dobijenih MSMDLPM algoritmom (treća kolona), Algoritam 1 (četvrta kolona) i CGD algoritmom (poslednja kolona). Kao što se vidi na Slikama 7.3 i 7.4, sva tri algoritma su postigli sličan kvalitet restauracije, ali je MSMDLPM efikasniji po svakom posmatranom indikatoru.

Uzimajući sve u obzir, uočavamo da MSMDLPM algoritam pruža validan pristup rešavanju problema uklanjanja zamućenja na slikama, a njegove performanse su bolje od performansi konkurentskih algoritama, tako da može se zaključiti da je MSMDLPM algoritam efikasniji u ovom upoređivanju.

Poglavlje 8

Gradijentni algoritmi za rešavanje problema bezuslovne optimizacije bazirani na neutrosofiji

Evidentan je uticaj neutrosofije na mnoge oblasti nauke i tehnologije, kao i njene brojne primene. U ovom poglavlju istraživanje je usmereno na dva pravca i oba se odnose na primenu neutrosofije po prvi put u cilju unapređenja metoda za rešavanje problema bezuslovne optimizacije.

Prvi pravac istraživanja je usmeren na poboljšanju metode linijskog pretraživanja za rešavanje problema bezuslovne optimizacije, a drugi pravac je usmeren na poboljšanju Dai–Liao konjugovano gradijentnog metoda, a samim tim i cele klase Dai–Liao konjugovano gradijentnih metoda. Tačnije, predlaže se i istražuje novi neutrosofsko logički sistem namenjen za izračunavanje parametra t potrebnog u Dai–Liao konjugovano gradijentnim iteracijama. Kada je u pitanju prvi pravac, istraživanja su bazirana na primeni simetrije uključene u neutrosofskoj logici (*eng. neutrosophic logic*) u određivanju odgovarajuće dužine koraka za klasu metoda sa opadajućim pravcem.

U ovom poglavlju biće predstavljene FMSM i FDL metode i naučni rezultati iz [153, 154] koji predstavljaju deo istraživanja ove doktorske disertacije. U radu [153] predložen je novi pravac pretraživanja koji proizilazi iz odgovarajućeg parametara ubrzanja dobijenog nakon nadogradnje ubrzano gradijentnog metoda sa neutrosofijom. Glavni cilj predloženih metoda je povezivanje ubrzano gradijentnih metoda i neutrosofije u cilju poboljšanja određivanja odgovarajuće dužine koraka i stvaranja efikasne metode za rešavanje problema bezuslovne optimizacije. Globalna konvergencija FMSM metode je teorijski istražena i dokazana, dok numerički rezultati potvrđuju efikasnost metoda. Kao posledica rešavanja problema bezuslovne optimizacije data je efikasna primena FMSM metode u regresionoj analizi i samim tim data je veza između dve naučne oblasti: nelinearna optimizacija i regresiona analiza.

U radu [154] je dato novo pravilo za određivanje parametra t u Dai–Liao konjugovano gradijentnoj metodi. To pravilo je bazirano na neutrosofiji. Definisanjem novog načina određivanja vrednosti parametra t došlo se i do nove varijante Dai–Liao konjugovano gradi-

jentnog metoda poznatog kao FDL metod. Za FDL metod u [154] dat je algoritam, dokazana globalna konvergencija i numerička efikasnost.

8.1 Neutrosofija i bezuslovna optimizacija

Fazi logika, intuicionistička fazi logika i neutrosofska logika se pojavljuju kao efikasni alati za rukovanje matematičkim modelima sa neizvesnošću, nejasnošću, dvosmislenošću, nepreciznošću, nepotpunom sigurnošću, nepotpunošću, nedоследnošću i suvišnošću. Neutrosofska logika se može smatrati jednom od novijih teorija zasnovanih na fundamentalnim principima neutrosofije, koja zapravo spada u grupu mnogovrednosnih logika i zapravo predstavlja produžetak fazi logike. Neutrosofska logika se takođe može smatrati novom granom logike koja se bavi nedostacima fazi i klasične logike, kao i intuicionističke fazi logike. Neki od nedostataka fazi logike, kao što je neuspeh u rukovanju protivrečnim informacijama, značajno su smanjeni primenom neutrosofske logike. Istinitost i neistinitost u neutrosofskoj logici su nezavisne, dok su u intuicionističkoj fazi logici zavisne veličine. Neutrosofska logika može da manipuliše i nepotpunim i protivrečnim podacima. Stoga postoji potreba da se istraži upotreba neutrosofske logike u različitim oblastima koristeći nove napredne tehnike računarske inteligencije.

Neutrosofska logika je bolji izbor od fazi logike i intuicionističke fazi logike u predstavljanju podataka iz stvarnog sveta i njihovim izvršavanjem iz sledećih razloga:

- (a) Fazi logički i intuicionistički fazi logički sistemi zanemaruju značaj neodređenosti. Fazi logički kontroler se bazira na pripadnosti i nepripadnosti određenog elementa u određenom skupu i uzima u obzir neodređenu prirodu generisanih podataka.
- (b) Fazi logički ili intuicionistički fazi logički sistem je dodatno ograničen činjenicom da je zbir vrednosti za pripadnost i nepripadnost ograničen na 1. Više detalja je dostupno u [149].
- (c) Neutrosofska logika jasno rasuđuje razlike koncepta apsolutne istine i relativne istine, pretpostavljajući postojanje apsolutne istine sa dodeljenom vrednošću 1^+ .
- (d) Neutrosofska logika je pogodna u situaciji preklapanja regiona fazi sistema [11].

Posmatrajmo univerzalni skup $\mathcal{U}$. Teorija fazi skupova se oslanja na funkciju pripadnosti (*eng. membership function*) $T(u) \in [0, 1], u \in \mathcal{U}$ koja definiše stepen pripadnosti u u $\mathcal{N} \subset \mathcal{U}$ [186]. Pored toga, fazi skup $\mathcal{N}$ nad $\mathcal{U}$ je skup uređenih parova $\mathcal{N} = \{\langle u, T(u) \rangle \mid u \in \mathcal{U}\}$.

Intuicionistički fazi skup (*eng. intuitionistic fuzzy set*) je uveden na osnovu funkcije nepripadnosti (*eng. nonmembership function*) $F(u) \in [0, 1], u \in \mathcal{U}$ u [13]. Prateći filozofiju korišćenja dve suprotstavljene funkcije pripadnosti, intuicionistički fazi skup $\mathcal{N}$ nad $\mathcal{U}$ je definisan kao skup uređenih trojki $\mathcal{N} = \{\langle u, T(u), F(u) \rangle \mid u \in \mathcal{U}\}$, koji je baziran na nezavisnim članovima tj. $T(u), F(u) : \mathcal{U} \rightarrow [0, 1]$ i $0 \leq T(u) + F(u) \leq 1$.

Teoriju intuicionističkih fazi skupova su proširili Smarandache u [148] i Wang i ostali saradnici u [168] u neutrosofsku teoriju. Suština ovog proširenja je u uvođenju funkcije

pripadnost neodređenosti $I(u)$ (*eng. indeterminacy-membership function*), koja simbolizuje nesigurnost u procesu donošenja odluka. Kao posledica takvog pristupa u neutrosofiji su elementi skupa definisani sa tri individualne funkcije pripadnosti [148, 168] koje su definisane pravilama simetrije:

- $T(u)$ – funkcija pripadnost istini (*eng. truth-membership function*),
- $I(u)$ – funkcija pripadnost neodređenosti i
- $F(u)$ – funkcija pripadnost neistini (*eng. falsity-membership function*).

Neutrosofski skup sa jednom vrednošću (*eng. single-valued neutrosophic set*) $\mathcal{N}$ nad $\mathcal{U}$ je skup neutrosofskih brojeva oblika $\mathcal{N} = \{\langle u, T(u), I(u), F(u) \rangle \mid u \in \mathcal{U}\}$. Funkcije pripadnosti nezavisno uzimaju vrednosti iz $[0, 1]$, pri čemu iniciraju $T(u), I(u), F(u) : \mathcal{U} \rightarrow [0, 1]$ i $0 \leq T(u) + I(u) + F(u) \leq 3$.

Neutrosofski skup je simetričan po prirodi pošto se neodređenost (I) pojavljuje u sredini između istinitosti (T) i neistinitosti (F) [89, 115]. Štaviše, oplemenjeni neutrosofski skup sa dve neodređenosti I_1 i I_2 u sredini između T i F takođe uključuje neku vrstu simetrije [115].

Neutrosofski skupovi imaju značajnu primenu u uklanjanju šuma, grupisanju, segmentaciji i klasifikaciji u brojnim aplikacijama za obradu medicinskih slika. U [72] je predložena upotreba neutrosofske teorije u uklanjanju šuma medicinskih slika i njihovoj segmentaciji, tako da neutrosofsku sliku karakterišu tri skupa pripadnosti. Nekoliko primena neutrosofskih sistema opisano je u [34]. Primena neutrosofije u obradi prirodnog jezika i analizi osećanja je istraživana u [115].

Osnovni cilj u ovom poglavlju je poboljšanje nekih od glavnih metoda opadajućeg gradijenta za rešavanje problema bezuslovne optimizacije i klase Dai–Liao konjugovano gradijentnih metoda koristeći prednosti neutrosofskih sistema. Glavni rezultati istraživanja su istaknuti na sledeći način:

- (1) Istražena je primena neutrosofske logike u određivanju dodatne dužine koraka u metodama linijskog pretraživanja za rešavanje problema bezuslovne optimizacije.
- (2) Opisane su i istražene primene neutrosofske logike u metodama višestrukih dužina koraka za rešavanje problema bezuslovne optimizacije.
- (3) Ispitana je primena neutrosofske logike u određivanju vrednosti parametra t u Dai–Liao konjugovano gradijentnoj metodi.
- (4) Sprovedena je teorijska analiza kako bi se potvrdila globalna konvergencija predloženih iteracija pod istim uslovima kao i za odgovarajuće originalne metode.
- (5) Predstavljeno je numeričko upoređivanje između predloženih i originalnih algoritama koristeći profile performansi Dolan-a i Moré-a i statističko rangiranje optimizacionih metoda.
- (6) Istražena je primena fazi metoda optimizacije u regresionoj analizi.

Osnovna motivacija je bazirana na činjenici da je neutrosofski logički sistem efikasan alat da iskoristi ponašanje funkcije cilja u metodama bezuslovne optimizacije kroz iteracije i na taj način sprečiti divergenciju ili sporu konvergenciju. Efikasno praćenje promena u vrednostima funkcije cilja omogućeno je pravilno definisanim “if-then” pravilima i odgovarajućim funkcijama pripadnosti.

Očekuje se da će principi simetrije uključeni u osnovu neutrosofske logike (videti [153, 154, 115] za više detalja) i praćenje prethodnih vrednosti funkcije cilja dovesti do efikasnijih metoda za rešavanje problema bezuslovne optimizacije. Ovakva očekivanja su potvrđena predloženim metodama u nastavku poglavlja i numeričkim rezultatima dobijeni nizom numeričkih eksperimenata.

8.2 Fazi modifikovani SM metod

Primena neutrosofske logike prilikom određivanja dodatne dužine koraka u metodama opadajućeg gradijenta za rešavanje problema bezuslovne optimizacije (2.1) istražena je u ovom odeljku.

Opšte iterativno pravilo za rešavanje problema (2.1) je metod opadajućeg gradijenta (2.4). Međutim, u ovom odeljku istraživanje je usmereno na iterativnu šemu (3.22) poznatu kao algoritam (metod) poboljšanog opadajućeg gradijenta. Uloge dodatnog koraka γ_k i osnovne dužine koraka α_k su jasno razdvojene i međusobno se dopunjuju. Dužina koraka α_k se određuje korišćenjem netačnog linijskog pretraživanja, odnosno Algoritma 2.2.3, dok se γ_k izračunava na osnovu Tejlorovog razvoja funkcije $f(\mathbf{x})$.

Različiti oblici i poboljšanja iterativne šeme poboljšanog opadajućeg gradijenta (3.22) su predloženi u [126, 129, 131, 155, 156]. Metod predložen u [155] poznat kao SM metod definisan je iteracijama

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k (\gamma_k^{SM})^{-1} \mathbf{g}_k, \quad (8.1)$$

gde je $\gamma_k^{SM} > 0$ parametar ubrzanja određen korišćenjem Tejlorovog razvoja ciljne funkcije $f(\mathbf{x}_k - \alpha_k (\gamma_k^{SM})^{-1} \mathbf{g}_k)$ i definisan jednakošću

$$\gamma_{k+1}^{SM} = \mathcal{U} \left(2\gamma_k^{SM} \frac{\gamma_k^{SM} \Delta_k + \alpha_k \|\mathbf{g}_k\|^2}{\alpha_k^2 \|\mathbf{g}_k\|^2} \right),$$

pri čemu je $\Delta_k := f_{k+1} - f_k$, $f_k := f(\mathbf{x}_k)$ i $\mathcal{U}(u) = \begin{cases} u, & u > 0 \\ 1, & u \leq 0. \end{cases}$

Modifikacija *SM* metode definisana je u Odeljku 4.2 i poznata je kao *MSM* metod sa iteracijama oblika

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \tau_k (\gamma_k^{MSM})^{-1} \mathbf{g}_k, \quad (8.2)$$

gde je dužina koraka $\alpha_k \in (0, 1)$ određena linijskim pretraživanjem unazad, odnosno Algo-

ritmom 2.2.3, $\tau_k = 1 + \alpha_k - \alpha_k^2$ i

$$\gamma_{k+1}^{MSM} = \mathfrak{U} \left(2\gamma_k^{MSM} \frac{\gamma_k^{MSM} \Delta_k + \alpha_k \tau_k \|\mathbf{g}_k\|^2}{(\alpha_k \tau_k)^2 \|\mathbf{g}_k\|^2} \right). \quad (8.3)$$

U ovom odeljku predloženo je poboljšanje iterativnih pravila linijskog pretraživanja za rešavanje problema (2.1). Osnovna ideja je zasnovana na primeni neutrosofske logike u određivanju odgovarajuće dužine koraka za različita pravila opadajućeg gradijenta. Sama ideja je proizašla iz principa hibridizacije dat u [86, 129, 132], gde se koristi odgovarajući parametar korekcije Ω_k sa fiksnom vrednošću. Hibridizacija SM iteracija opisana je u Odeljku 3.6 i poznata je kao *HSM* metoda sa iterativnim pravilom

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (\Omega_k + 1)(\gamma_k^{HSM})^{-1} \alpha_k \mathbf{g}_k, \quad (8.4)$$

gde je $\Omega_k \in (0, 1)$ parametar korekcije, a γ_k^{HSM} je parametar ubrzanja definisan na sledeći način

$$\gamma_{k+1}^{HSM} = \mathfrak{U} \left(2\gamma_k^{HSM} \frac{\gamma_k^{HSM} \Delta_k + (\Omega_k + 1) \alpha_k \|\mathbf{g}_k\|^2}{(\Omega_k + 1)^2 \alpha_k^2 \|\mathbf{g}_k\|^2} \right).$$

Hibridizacija nekoliko metoda poboljšanog opadajućeg gradijenta, uključujući i MSM metod, predložene su i ispitane u Odeljku 4.3, kao i u radovima [86, 132]. Pregled metoda nastalih hibridizacijom iteracija poboljšanog opadajućeg gradijenta sa iterativnim procesima Picard–Mann, Ishikawa i Khan [82, 90, 134] dat je u [139], a takođe urađeno je i detaljno numeričko upoređivanje između hibridnih i nehibridnih metoda poboljšanog opadajućeg gradijenta.

Osnovni zadatak u ovom istraživanju je korišćenje prilagodljivog neutrosofsko logičkog parametra ν_k umesto fiksnog parametra korekcije $\Omega_k + 1$ u određivanju odgovarajućih dužina koraka za različite metode opadajućeg gradijenta. Parametar ν_k u svakoj iteraciji biće određen na osnovu neutrosofskog logičkog kontrolera (*eng. neutrosophic logic controller*, NLC).

Iteracije fazi opadajućeg pravca (*FDD*) su definisane kao modifikacija iteracija opadajućeg gradijenta (2.4), na sledeći način:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \nu_k \alpha_k \mathbf{d}_k, \quad (8.5)$$

gde je $\nu_k > 0$ odgovarajuće definisan fazi parametar, a $\mathbf{d}_k = -\mathbf{g}_k$ odgovarajući opadajući pravac. Generalno, ν_k bi trebalo da zadovolji sledeće uslove

$$\nu_k \begin{cases} < 1, & \text{ako je } \Delta_k > 0, \\ = 1, & \text{ako je } \Delta_k = 0, \\ > 1, & \text{ako je } \Delta_k < 0. \end{cases} \quad (8.6)$$

Glavna ideja korišćena u (8.6) je smanjenje složene dužine koraka $\nu_k \alpha_k$ u iteraciji (8.5)

u slučaju kada se vrednost funkcije f povećava i povećanje složene dužine koraka $\nu_k \alpha_k$ u slučaju kada se vrednost funkcije f smanjuje. U nastavku definišimo opštu iterativnu šemu za fazi metod opadajućeg gradijenta (FGD) koja je ekvivalentna iterativnoj šemi (8.5) na sledeći način

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \nu_k \alpha_k \mathbf{g}_k. \quad (8.7)$$

Fazi SM metod (FSM) je definisan kao modifikacija SM metode pomoću iterativne šeme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \nu_k \alpha_k (\gamma_k^{FSM})^{-1} \mathbf{g}_k, \quad (8.8)$$

gde je

$$\gamma_{k+1}^{FSM} = \mathfrak{U} \left(2\gamma_k^{FSM} \frac{\gamma_k^{FSM} \Delta_k + \nu_k \alpha_k \|\mathbf{g}_k\|^2}{(\nu_k \alpha_k)^2 \|\mathbf{g}_k\|^2} \right). \quad (8.9)$$

Na osnovu (8.2) definišemo fazi MSM metod ($FMSM$) pomoću iterativne šeme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \nu_k \alpha_k \tau_k (\gamma_k^{FMSM})^{-1} \mathbf{g}_k, \quad (8.10)$$

gde je

$$\gamma_{k+1}^{FMSM} = \mathfrak{U} \left(2\gamma_k^{FMSM} \frac{\gamma_k^{FMSM} \Delta_k + \nu_k \alpha_k \tau_k \|\mathbf{g}_k\|^2}{(\nu_k \alpha_k \tau_k)^2 \|\mathbf{g}_k\|^2} \right). \quad (8.11)$$

U Tabeli 8.1 prikazane su različite dužine koraka koje su korišćene u iterativnim šemama u ovom odeljku, a u kolonama gde stoji “-” označava odsustvo odgovarajućeg parametra. Tabela 8.1: Parametri u metodama opadajućeg gradijenta i neutrososkim modifikacijama.

Metod	Parametri za dužinu koraka		
	Prvi	Drugi	Treći
GD	α_k	-	-
FGD	ν_k	α_k	-
SM	α_k	$(\gamma_k^{SM})^{-1}$	-
FSM	ν_k	α_k	$(\gamma_k^{FSM})^{-1}$
MSM	$\alpha_k \tau_k$	$(\gamma_k^{MSM})^{-1}$	-
$FMSM$	ν_k	$\alpha_k \tau_k$	$(\gamma_k^{FMSM})^{-1}$

Algoritam 2.2.3 koristi se za određivanje dužine koraka α_k , dok Algoritam 8.2.1 opisuje opšti okvir klase fazi metoda opadajućeg pravca FDD .

Funkcija ocenjivanja (*eng. Score function*) u koraku 6. Algoritma 8.2.1 se koristi za kvantifikaciju i evaluaciju stepena pripadnosti, nepripadnosti i neodređenosti elemenata unutar fazi logičkih sistema.

Opšta struktura fazi neutrososkih metoda optimizacije prati proces prikazan na dijagramu Slike 8.1.

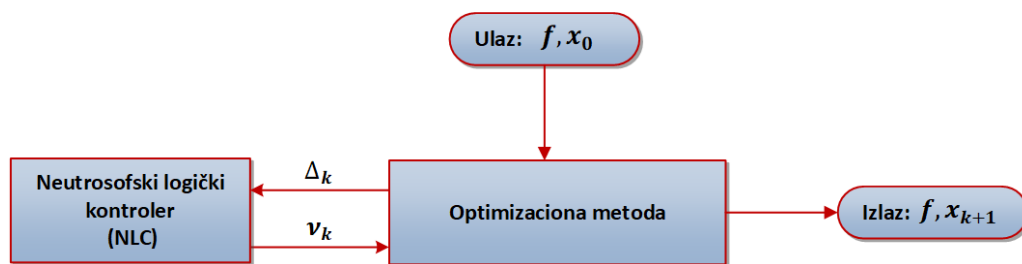
Da bismo definisali fazi modifikovani SM metod ($FMSM$ metod) potrebno je da definišemo 6., 7., 8. i 9. korak u Algoritmu 8.2.1, odnosno funkciju ocenjivanja, neutrosifikaciju (*eng.*

Algoritam 8.2.1 Opšti fazi metod opadajućeg pravca *FDD*.

Ulaz: Ciljna funkcija $f(\mathbf{x})$, početna tačka $\mathbf{x}_0 \in \text{dom}(f)$.

- 1: Postaviti $k = 0$, $\nu_0 = 1$, izračunati $f(\mathbf{x}_0)$, $\mathbf{g}_0 = \nabla f(\mathbf{x}_0)$ i generisati pravac pretraživanja $\mathbf{d}_0 = -\mathbf{g}_0$.
- 2: Ako su ispunjeni kriterijumi za kraj ciklusa tada prekinuti iteracije; inače preći na sledeći korak.
- 3: Odrediti dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.
- 4: Izračunati $\mathbf{x}_{k+1}$ koristeći (8.5).
- 5: Izračunati $f(\mathbf{x}_{k+1})$, $\mathbf{g}_{k+1} = \nabla f(\mathbf{x}_{k+1})$ i generisati pravac pretraživanja $\mathbf{d}_{k+1}$.
- 6: (Funkcija ocenjivanja) Izračunati $\Delta_k := f_{k+1} - f_k$.
- 7: (Neutrosifikacija) Izračunati $T(\Delta_k)$, $I(\Delta_k)$ i $F(\Delta_k)$ koristeći odgovarajuće funkcije pripadnosti.
- 8: Definirati neutrosofski mehanizam zaključivanja.
- 9: (Deneutrosifikacija) Izračunati $\nu_k(\Delta_k)$ koristeći pravilo deneutrosifikacije.
- 10: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.



Slika 8.1: Opšta struktura fazi metoda optimizacije.

neutrosophication), neutrosofski mehanizam zaključivanja (*eng. neutrosophic inference engine*) i deneutrosifikaciju (*eng. de-neutrosophication*).

- (1) *Neutrosifikacija*: Koristeći tri funkcije pripadnosti neutrosofska logika preslikava ulaz $\vartheta := f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)$ u neutrosofske trojke $(T(\vartheta), I(\vartheta), F(\vartheta))$.

Funkcija pripadnost istini je definisana kao sigmoidna funkcija oblika

$$T(\vartheta) = 1/(1 + e^{-c_1(\vartheta - c_2)}). \quad (8.12)$$

Parametar c_1 je odgovoran za nagib u tački preloma $\vartheta = c_2$.

Funkcija pripadnost neistini je definisana kao sigmoidna funkcija oblika

$$F(\vartheta) = 1/(1 + e^{c_1(\vartheta - c_2)}). \quad (8.13)$$

Funkcija pripadnost neodređenosti je Gausova funkcija oblika

$$I(\vartheta) = e^{-\frac{(\vartheta - c_2)^2}{2c_1^2}}, \quad (8.14)$$

gde parametar c_1 predstavlja standardnu devijaciju, a parametar c_2 matematičko očekivanje. Neutrosofikacija jasne vrednosti (*eng. crisp value*) $\vartheta \in \mathbb{R}$ koja se koristi u implementaciji je transformacija ϑ u $\langle \vartheta : T(\vartheta), I(\vartheta), F(\vartheta) \rangle$, gde su funkcije pripadnosti definisane funkcijama od (8.12) do (8.14). Budući da je krajnji cilj minimizirati $f(\mathbf{x})$, logično je koristiti Δ_k kao meru u neutrosofskom logičkom kontroleru. Dakle, razmatramo dinamički neutrosofski skup (*eng. dynamic neutrosophic set*, DNS) definisan sa $\mathfrak{D} := \{ \langle T(\Delta_k), I(\Delta_k), F(\Delta_k) \rangle; \Delta_k \in \mathbb{R} \}$.

- (2) *Neutrosofski mehanizam zaključivanja*: Neutrosofska pravila između fazi ulaznog skupa $\mathfrak{I}$ i fazi izlaznog skupa prema neutrosofskom formatu $\mathfrak{D} = \{T, I, F\}$ su opisana sledećim “IF–THEN” pravilima:

$$R_1 : \text{If } \mathfrak{I} = P \text{ then } \mathfrak{D} = \{T, I, F\}$$

$$R_2 : \text{If } \mathfrak{I} = N \text{ then } \mathfrak{D} = \{T, I, F\}.$$

Oznake P i N predstavljaju fazi skupove i ukazuju na pozitivnu i negativnu grešku. Korišćenjem ujedinenja $R = R_1 \cup R_2$, dobijamo $\mathfrak{D}_i = \mathfrak{I} \circ R_i$, $i = 1, 2$, gde $\circ$ simbolizuje fazi transformaciju. Štaviše, sledi da $\kappa_{\mathfrak{I} \circ R}(\zeta) = \kappa_{\mathfrak{I} \circ R_1} \vee \kappa_{\mathfrak{I} \circ R_2}$, $\kappa_{\mathfrak{I} \circ R}(\zeta) = \sup(\kappa_{\mathfrak{I}} \wedge \kappa_{\mathfrak{D}_i})$, i $i = 1, 2$, gde $\wedge$ (odnosno $\vee$) označava (min, max, max) operator, (odnosno (max, min, min) operator). Proces pretvaranja fazi izlaza u jedinstvenu jasnu izlaznu vrednost poznat je kao defazifikacija. Postoje različite metode defazifikacije koje se mogu koristiti za izvođenje ove procedure. Metod centroida, metod ponderisanog proseka i principi pripadanja max ili mean–max su neke od popularnih metoda defazifikacije. U ovom istraživanju, korišćen je sledeći metod defazifikacije, nazvan centroid, da bi se dobio vektor jasne izlazne vrednosti $\zeta^* = [T(\Delta_k), I(\Delta_k), F(\Delta_k)] \in \mathbb{R}^3$ od fazi vektora $\zeta = \{T(\Delta_k), I(\Delta_k), F(\Delta_k)\}$:

$$\zeta^* = \frac{\int_{\mathfrak{D}} \zeta \kappa_{\mathfrak{I} \circ R}(\zeta) d\zeta}{\int_{\mathfrak{D}} \kappa_{\mathfrak{I} \circ R}(\zeta) d\zeta}. \quad (8.15)$$

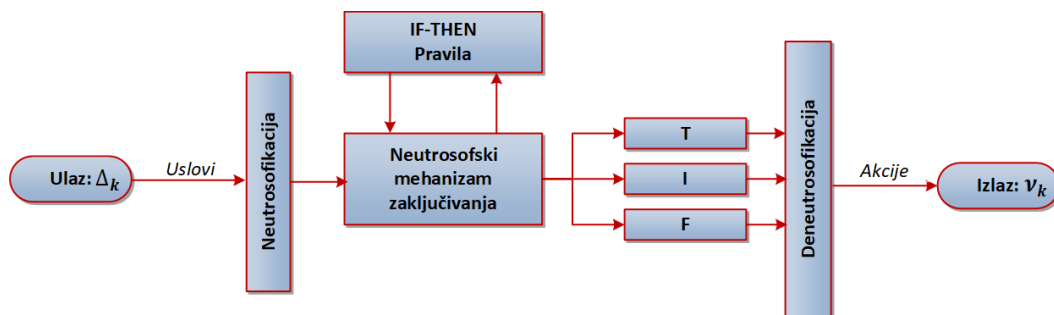
- (3) *Deneutrosofikacija*: Pretpostavlja konverziju $\langle T(\Delta_k), I(\Delta_k), F(\Delta_k) \rangle \rightarrow \nu_k(\Delta_k) \in \mathbb{R}$ što dovodi do jedne (jasne) vrednosti $\nu_k(\Delta_k)$. Sledeće pravilo deneutrosofikacije je predloženo za dobijanje parametra $\nu_k(\Delta_k)$ i prati navedena ograničenja u (8.6):

$$\nu_k(\Delta_k) = \begin{cases} 1 - (T(\Delta_k) + I(\Delta_k) + F(\Delta_k)) / c_1, & \Delta_k > 0 \\ 1, & \Delta_k = 0 \\ 3 - (T(\Delta_k) + I(\Delta_k) + F(\Delta_k)), & \Delta_k < 0. \end{cases} \quad (8.16)$$

Parametar $c_1 \geq 3$ očuvava ograničenje $\nu_k(\Delta_k) < 1$ u slučaju $\Delta_k > 0$. Štaviše, definicija

(8.16) pretpostavlja da funkcije pripadnosti moraju da zadovolje $T(\Delta_k) + I(\Delta_k) + F(\Delta_k) < 2$ u slučaj $\Delta_k < 0$.

Radi boljeg shvatanja, struktura neutrosofskog logičkog kontrolera razložena neutrosofskim pravilima je predstavljena na dijagramu Slike 8.2. Važno je znati da je struktura neutrosofskog logičkog kontrolera napravljena posebno da reši pitanja o kojima se raspravlja u ovom poglavlju, uključujući izabrane funkcije pripadnosti, broj izabranih fazi pravila i izabrani metod deneutrosofikacije. Kao rezultat toga, struktura neutrosofskog logičkog kontrolera je heuristička i različite strukture mogu biti zahtevane za različite aplikacije.



Slika 8.2: Struktura neutrosofskog logičkog kontrolera bazirana na neutrosofskim pravilima.

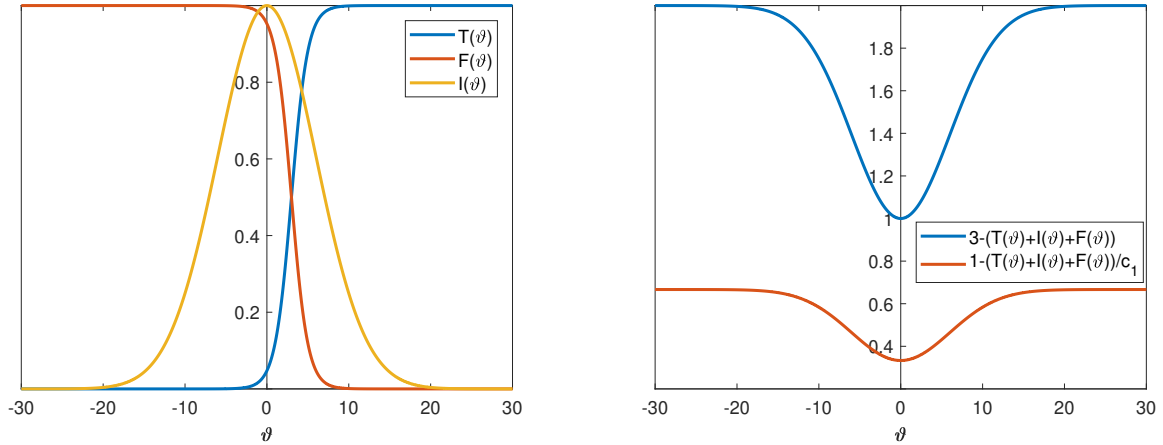
Potrebne vrednosti parametara za strukturu neutrosofskog logičkog kontrolera date su u Tabeli 8.2 i korišćene su u svim numeričkim eksperimentima i grafikonima ovog poglavlja.

Tabela 8.2: Preporučene vrednosti parametara u strukturi neutrosofskog logičkog kontrolera.

Grupa	Funkcije pripadnosti		c_1	c_2	Težina
Ulaz	Istini	Sigmoidna	1	3	1
	Neistini	Sigmoidna	1	3	1
	Neodređenosti	Gausova	6	0	1
Izlaz	Deneutrosofikacija (8.16)		3	-	1

Naš bezuslovni zahtev je $\nu_k(\Delta_k) \geq 0$. Ispunjenje ovog zahteva neposredno sledi iz vrednosti funkcija pripadnosti $T(\Delta_k)$, $F(\Delta_k)$, $I(\Delta_k)$ tokom procesa neutrosofikacije, koje su predstavljene na Slici 8.3a. Izlazna vrednost neutrosofskog logičkog kontrolera, $\nu_k(\Delta_k)$, tokom procesa deneutrosofikacije je predstavljena na Slici 8.3b.

Slika 8.3 jasno pokazuje da (8.16) zadovoljava osnovne zahteve nametnute u (8.6). Tačnije, na Slici 8.3b prikazani su grafici $1 - (T(\Delta_k) + I(\Delta_k) + F(\Delta_k)) / c_1 < 1$ u slučaju $\Delta_k > 0$ i $3 - (T(\Delta_k) + I(\Delta_k) + F(\Delta_k)) \geq 1$ u slučaju $\Delta_k < 0$.



(a) Neutrosifikacija na bazi funkcija (8.12)–(8.14). (b) Deneutrosifikacija na bazi funkcije (8.16).

Slika 8.3: Neutrosifikacija i deneutrosifikacija na osnovu parametara u Tabeli 8.2.

Napomena 8.2.1. Tokom iteracija, vrednost funkcije se smanjuje i teži ka minimumu, tako da $\lim_{k \rightarrow \infty} \Delta_k = 0$, odnosno $\lim_{k \rightarrow \infty} \nu_k(\Delta_k) = 1$. Ovo zapažanje dovodi do zaključka da parametar $\nu_k \rightarrow 1$ kako se približavamo minimumu funkcije, a samim tim i uticaj neutrosifikacije na gradijentne metode opada. Ovakvo poželjno ponašanje $\nu_k(\Delta_k)$ bila je naša namera od samog početka.

Na osnovu date diskusije, Algoritam 8.2.2 daje detaljan postupak za primenu fazi modifikovanog SM metoda (FMSM metod).

Algoritam 8.2.2 Fazi modifikovani SM metod (FMSM metod).

Ulaz: Funkcija cilja $f(\mathbf{x})$ i početna aproksimacija $\mathbf{x}_0 \in \text{dom}(f)$.

- 1: Postaviti $k = 0$, $\gamma_0 = 1$, $\nu_0 = 1$ i izračunati $f(\mathbf{x}_0)$, $\mathbf{g}_0 = \nabla f(\mathbf{x}_0)$.
- 2: Ako su ispunjeni kriterijumi za kraj ciklusa tada prekinuti iteracije; inače preći na sledeći korak.
- 3: Odrediti dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3 i pravac pretraživanja $\mathbf{d}_k = -\nu_k \tau_k (\gamma_k^{FMSM})^{-1} \mathbf{g}_k$.
- 4: Izračunati $\mathbf{x}_{k+1}$ koristeći (8.10) i odrediti $f(\mathbf{x}_{k+1})$ i $\mathbf{g}_{k+1} = \nabla f(\mathbf{x}_{k+1})$.
- 5: Odrediti γ_{k+1}^{FMSM} na osnovu (8.11).
- 6: Izračunati $\Delta_k := f_{k+1} - f_k$.
- 7: Odrediti $T(\Delta_k)$, $I(\Delta_k)$ i $F(\Delta_k)$ koristeći (8.12)–(8.14).
- 8: Izračunati $\zeta^* = [T(\Delta_k), I(\Delta_k), F(\Delta_k)]$ koristeći (8.15).
- 9: Odrediti $\nu_k := \nu_k(\Delta_k)$ koristeći (8.16).
- 10: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

8.3 Fazi neutrosofski Dai–Liao konjugovano gradijentni metod

Neutrosofska logika je primenjena u predhodnom odeljku, odnosno u radu [153] u određivanju odgovarajućih dužina koraka za klasu ubrzano opadajućih gradijentnih metoda optimizacije. Pristup u Odeljku 8.2 prisvaja dodatni fazi parametar koji stabilizuje ponašanje važne klase opadajućih gradijenata.

Motivisani tim pristupom, u ovom odeljku istražujemo primenu neutrosofske logike u poboljšanju performansi Dai–Liao metode. Na osnovu pregleda i analize klase Dai–Liao metoda, predlaže se novi postupak određivanja parametra t_k . Predloženi postupak definiše t_k kao izlaz koji generiše odgovarajući neutrosofski logički kontroler. Naša ideja je da zamenimo klasični parametar t_k u (5.9) prilagodljivim neutrosofskim logičkim parametrom ν_k , koji se dobija kao izlaz neutrosofskog logičkog kontrolera. Kako je $0 \leq t_k \leq 1$, odlučili smo da definišemo t_k kao vrednost ν_k , bez dodatnih parametara. Opravdanje za korišćenje fazi vrednosti ν_k definisane na bazi neutrosofske logike leži u činjenici da ne postoji ni najbolja vrednost za parametar t_k , niti jasno pravilo za njegovo određivanje.

Očigledno je da ne postoji jedinstveno rešenje za određivanje parametra t_k u Dai–Liao iteracijama (5.9). Sva dosadašnja rešenja su bazirana na određenim pravilima. Štaviše, jasno je da postoje bezbrojne mogućnosti za definisanje novih pravila za određivanje parametra t_k . Naš cilj je da poboljšamo ponašanje Dai–Liao klase za rešavanje problema bezuslovne optimizacije uz podršku odgovarajućeg neutrosofskog logičkog sistema. Neodređenost i neizvesnost u predviđanju optimalne vrednosti parametra t_k opravdavaju korišćenje fazi veličina kao i neutrosofskih principa u određivanju vrednosti parametra t_k . Posebna pogodnost u primeni neutrosofije u određivanju Dai–Liao parametra t_k proizilazi iz ograničenja $0 \leq t_k \leq 1$. Štaviše, neutrosofski logički sistem je pogodan alat za praćenje ponašanja funkcije cilja u određivanju potrebnih parametara uz pomoć odgovarajuće definisanih “if–then” pravila i odgovarajućih funkcija pripadnosti.

Fazi neutrosofski Dai–Liao konjugovano gradijentni metod je početno koncipiran kao modifikacija Dai–Liao konjugovano gradijentnog metoda (5.9), gde se pravci pretraživanja $\mathbf{d}_k$ izračunavaju na osnovu rekurentnih pravila

$$\mathbf{d}_k = \begin{cases} -\mathbf{g}_0, & k=0, \\ -\mathbf{g}_k + \beta_k^{\text{FDL}} \mathbf{d}_{k-1}, & k \geq 1, \end{cases} \quad (8.17)$$

gde je β_k^{FDL} parametar ažuriranja konjugovanog gradijenta definisan na sledeći način

$$\beta_k^{\text{FDL}} = \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - \nu_k \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}, \quad (8.18)$$

tako da je ν_k odgovarajući neutrosofski parametar. Osnovni zadatak je definisati ν_k kao funkciju $\nu_k := \nu_k(\Delta_k)$, gde je $\Delta_k := f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})$, pri čemu $\nu_k(\Delta_k)$ zadovoljava sledeće

ograničenje

$$0 \leq \nu_k(\Delta_k) \leq 1. \quad (8.19)$$

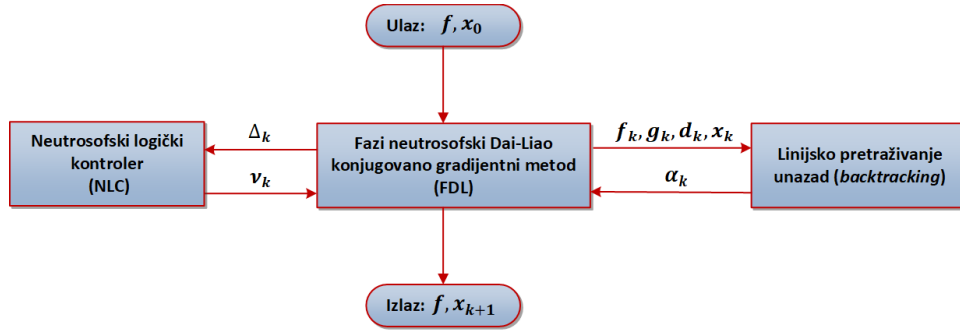
Poznato je da $\nu_k(\Delta_k) = 0$ transformiše (5.8) u

$$\mathbf{d}_k^T \mathbf{y}_{k-1} = 0. \quad (8.20)$$

Dakle, jednačina (8.20) može se smatrati odrazom uslova konjugacije koji u kombinaciji sa (5.1) određuje β_k^{HS} parametar ažuriranja konjugovanog gradijenta u [77]. Suprotno za $\nu_k(\Delta_k) = 1$ jednačina (5.8) se smatra uslovom konjugacije koji implicitno ispunjava kvazi-Njutnovne karakteristike. Za više detalja o ovim slučajevima pogledati [17, 42].

Ideja za definisanje novog parametra t_k u Dai–Liao konjugovano gradijentnom metodu (5.9) dolazi iz neutrosofske logike. Prema ovoj ideji, cilj je definisati $t_k := \nu_k(\Delta_k)$ sa vrednostima unutar intervala $[0, 1]$ prema neutrosofskim principima.

Opšta struktura fazi neutrosofskog Dai–Liao konjugovano gradijentnog metoda je data na dijagramu Slike 8.4. Ulaz za neutrosofsko logički kontroler prikazan na Slici 8.4 je $\Delta_k := f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})$, a izlaz je željena veličina neutrosofskog parametra ν_k . To znači, da je osnovna ideja definisati neutrosofski parametar ν_k na osnovu dve uzastopne vrednosti ciljne funkcije f . S druge strane, linijsko pretraživanje unazad je odgovorno za odgovarajuću dužinu koraka α_k u (2.2), dok je pravac opadanja $\mathbf{d}_k$ definisan sa (8.17). Koristeći ν_k , moguće je izračunati β_k^{FDL} u (8.18), a zatim na osnovu (2.2) generiše se nova iterativna tačka $\mathbf{x}_{k+1}$.



Slika 8.4: Opšta struktura fazi neutrosofskog Dai–Liao konjugovano gradijentnog metoda.

Da bismo razvili fazi neutrosofski Dai–Liao konjugovano gradijentni metod potrebno je da definišemo tri koraka: neutrosifikaciju, neutrosofski mehanizam zaključivanja i deneutrosifikaciju.

- (1) *Neutrosifikacija*: Neutrosofska logika preslikava ulaz $\Delta_k := f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})$ u neutrosofske uređene trojke $(T(\Delta_k), I(\Delta_k), F(\Delta_k))$ koristeći tri funkcije pripadnosti. Fun-

kcije pripadnosti su definisane da poboljšaju konjugovano gradijentna iterativna pravila korišćenjem numeričkog iskustva. Funkcija pripadnost istini $T(\Delta_k)$ je definisana kao sigmoidna funkcija (8.12), dok je funkcija pripadnost neistini $F(\Delta_k)$ definisana kao sigmoidna funkcija (8.13). I na kraju funkcija pripadnost neodređenosti $I(\Delta_k)$ je definisana kao Gausova funkcija (8.14). Dakle, neutrosofikacija je definisana kao prelaz $\Delta \rightarrow \langle T(\Delta), I(\Delta), F(\Delta) \rangle$, gde su funkcije pripadnosti određene jednačinama (8.12), (8.13) i (8.14).

- (2) *Neutrosofski mehanizam zaključivanja*: Neutrosofski mehanizam zaključivanja je identično definisan kao u Odeljku 8.2.
- (3) *Deneutrosofikacija*: Bazira se na transformaciji $\langle T(\Delta_k), I(\Delta_k), F(\Delta_k) \rangle \rightarrow \nu_k(\Delta_k) \in \mathbb{R}$, koja dovodi do jedne (jasne) vrednosti ν_k , definisane na sledeći način

$$\nu_k(\Delta_k) = 2 - (T(\Delta_k) + I(\Delta_k) + F(\Delta_k)). \quad (8.21)$$

Strukturu neutrosofskog logičkog kontrolera baziranog na neutrosofskim pravilima data je na dijagramu Slike 8.2. Potrebne vrednosti parametara za strukturu neutrosofskog logičkog kontrolera date su u Tabeli 8.3 i korišćene su u svim numeričkim eksperimentima i grafikonima ovog poglavlja.

Tabela 8.3: Preporučene vrednosti parametara u strukturi neutrosofskog logičkog kontrolera.

Grupa	Funkcije pripadnosti	c_1	c_2	Težina
Ulaz	Sigmoidna funkcija (8.12)	1	3	1
	Sigmoidna funkcija (8.13)	1	3	1
	Gausova funkcija (8.14)	120	0	1
Izlaz	Deneutrosofikacija (8.21)	-	-	1

Uz pomoć Leme 8.3.1 dokazana je ispunjenost ograničenja $0 \leq \nu_k(\Delta_k) \leq 1$ datog u (8.19).

Lema 8.3.1. *Nejednakost (8.19) važi za dati izbor funkcije ocenjivanja (8.21) i parametara datih u Tabeli 8.3.*

Dokaz. Da bi dokazali nejednakost (8.19), potrebno je vrednosti parametara iz Tabele 8.3 primeniti na funkcije pripadnosti (8.12), (8.13) i (8.14):

$$T(\Delta_k) = 1 / (1 + e^{-(\Delta_k - 3)}) , \quad F(\Delta_k) = 1 / (1 + e^{\Delta_k - 3}) \quad \text{ i } \quad I(\Delta_k) = e^{-\frac{\Delta_k^2}{2 \cdot 120^2}} .$$

Zatim funkcije pripadnosti zameniti u (8.21) i jednostavnim izračunavanjem dobija se

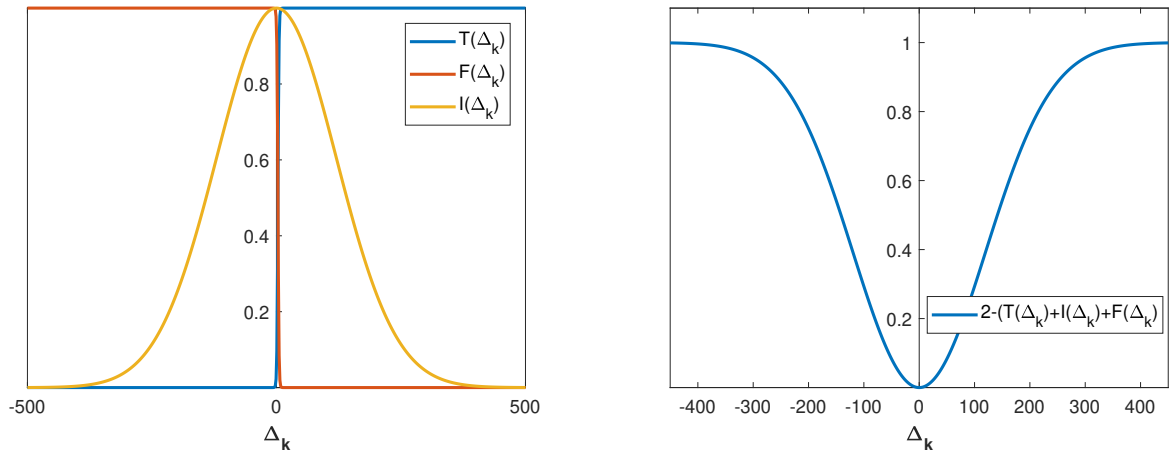
$$\nu_k(\Delta_k) = 2 - (T(\Delta_k) + I(\Delta_k) + F(\Delta_k)) = 1 - e^{-\frac{\Delta_k^2}{28800}}. \quad (8.22)$$

Analizom funkcije (8.22) unutar intervala $\Delta_k \in (-\infty, +\infty)$ dobija se da je $\min \nu_k(\Delta_k) = 0$ i $\max \nu_k(\Delta_k) = 1$, na osnovu čega sledi dokaz nejednakosti (8.19). $\square$

Grafici funkcija pripadnosti $T(\Delta_k)$, $I(\Delta_k)$ i $F(\Delta_k)$ su prikazani na Slici 8.5a. Ispunjenost uslova (8.19) za izlaznu vrednost ν_k parametra neutrososko logičkog kontrolera generisanog tokom opisane deneutrosifikacije prikazana je na Slici 8.5b.

Napomena 8.3.1. Tokom iteracija, vrednost funkcije cilja se smanjuje i teži ka minimumu, tako da $\lim_{k \rightarrow \infty} \Delta_k = 0$, odnosno $\lim_{k \rightarrow \infty} \nu_k(\Delta_k) = 0$. Takvo ponašanje dovodi do zaključka da parametar $\nu_k \rightarrow 0$ kako se približava minimumu funkcije f , a samim tim i uticaj predložene neutrososke strategije se smanjuje i nestaje, što se poklapa sa ciljem ovog istraživanja.

Napomena 8.3.2. Očigledno, veće vrednosti Δ_k dovode do porasta vrednosti $\nu_k(\Delta_k)$ približavajući se 1, što će biti označeno kao $\nu_k(\Delta_k) \nearrow 1$. Pored toga, na osnovu granice $\Delta_k \rightarrow 0$, predviđamo da će se manje vrednosti $\nu_k(\Delta_k)$ približavati 0, tj. $\nu_k(\Delta_k) \searrow 0$, u poslednjim (završnim) iteracijama. Kao rezultat, $\nu_k(\Delta_k)$ je pogodan kao podesivi regulator za parametar t u Dai–Liao konjugovano gradijentnom metodu.



(a) Neutrosifikacija na bazi funkcija (8.12)–(8.14). (b) Deneutrosifikacija na bazi funkcije (8.21).

Slika 8.5: Neutrosifikacija i deneutrosifikacija na osnovu parametara u Tabeli 8.3.

Metod linijskog pretraživanja unazad je dat pomoću Algoritma 2.2.3 u kome se određuje glavna dužina koraka α_k u (2.2), počevši od $\alpha_k = 1$ i nakon toga smanjujući vrednost

promenljive α_k tako da dovoljno smanji vrednost ciljne funkcije f . U nastavku je dat Algoritam 8.3.1 za fazi neutrosofski Dai–Liao konjugovano gradijentni metod, odnosno skraćeno FDL metod.

Algoritam 8.3.1 Fazi neutrosofski Dai–Liao (FDL) konjugovano gradijentni metod.

Ulaz: Funkcija cilja $f(\mathbf{x})$, početna tačka $\mathbf{x}_0$ i $0 < \epsilon, \delta < 1$.

1: Postaviti $k = 0$ i $\mathbf{d}_0 = -\mathbf{g}_0$.

2: Ako je

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{i} \quad \frac{|f_{k+1} - f_k|}{1 + |f_k|} \leq \delta,$$

onda prekinuti iteracije (STOP); inače preći na Korak 3.

3: Odrediti dužinu koraka $\alpha_k \in (0, 1]$ koristeći Algoritam 2.2.3.

4: Izračunati $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ i odrediti $\mathbf{g}_{k+1}$, $\mathbf{y}_k = \mathbf{g}_{k+1} - \mathbf{g}_k$ i $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$.

5: Izračunati $\Delta_k := f_k - f_{k+1}$ i odrediti $T(\Delta_k)$, $I(\Delta_k)$ i $F(\Delta_k)$ na osnovu (8.12)–(8.14).

6: Izračunati $\nu_k := \nu_k(\Delta_k)$ koristeći (8.21).

7: Izračunati β_{k+1}^{FDL} pomoću (8.18) i odrediti $\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_{k+1}^{\text{FDL}} \mathbf{d}_k$.

8: Postaviti $k := k + 1$ i preći na Korak 2.

Izlaz: $\mathbf{x}_k$ i $f(\mathbf{x}_k)$.

8.4 Analiza konvergencije

Postupak analize konvergencije FMSM i FDL metode je sličan postupku analize konvergencije MSM i EDL metode u Odeljcima 4.5 i 6.4 kao i ubrzanog gradijentnog SM metoda iz [155]. Dakle, kada je u pitanju FMSM metod prvo je sagledan skup uniformno konveksnih funkcija, a zatim i podskup strogo konveksnih kvadratnih funkcija, dok za FDL metod samo skup uniformno konveksnih funkcija.

U dokazima uzima se u obzir da važe Stav 4.5.1 i Lema 4.5.1 navedeni u Odeljku 4.5 za FMSM metod, dok za FDL metod se uzima da važi Stav 4.5.1. Preuzeti Stav i Lema važe na skupu uniformno konveksnih funkcija.

8.4.1 Ispitivanje konvergencije za FMSM metod

Neka je $\mathbf{d}_k$ vektor pretraživanja koji ima opadajući pravac i neka gradijent $\mathbf{g}(\mathbf{x})$ ispunjava Lipšicov uslov (4.32). Pored toga, neka važi Stav 4.5.2 naveden u Odeljku 4.5, odnosno neka dužina koraka α_k dobijena linijskim pretraživanjem unazad (Algoritam 2.2.3) zadovoljava nejednakost (4.37).

Radi jednostavnosti, označimo iteracije SM i MSM kao

$$\mathbf{x}_{k+1}^{(M)SM} = \mathbf{x}_k^{(M)SM} - \alpha_k \omega_k (\gamma_k^{(M)SM})^{-1} \mathbf{g}_k, \quad (8.23)$$

gde $\mathbf{x}_k^{(M)SM}$ označava $\mathbf{x}_k^{SM}$ (odnosno $\mathbf{x}_k^{MSM}$) u slučaju SM (odnosno MSM) metode i $\omega_k = 1$ (odnosno $\omega_k = \tau_k := 1 + \alpha_k - \alpha_k^2$). Slično, FSM i $FMSM$ iteracije su označene uobičajenom notacijom

$$\mathbf{x}_{k+1}^{F(M)SM} = \mathbf{x}_k^{F(M)SM} - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k, \quad (8.24)$$

gde $\mathbf{x}_k^{F(M)SM}$ označava $\mathbf{x}_k^{FSM}$ (odnosno $\mathbf{x}_k^{FMSM}$) u slučaju FSM (odnosno $FMSM$) metode i $\omega_k = 1$ (odnosno $\omega_k = \tau_k$).

Parametar $\gamma_k^{F(M)SM}$ u (8.24) određen je na osnovu (8.9) ili (8.11), odnosno

$$\gamma_{k+1}^{F(M)SM} = \mathfrak{U} \left(2\gamma_k^{F(M)SM} \frac{\gamma_k^{F(M)SM} \Delta_k + \nu_k \alpha_k \omega_k \|\mathbf{g}_k\|^2}{(\nu_k \alpha_k \omega_k)^2 \|\mathbf{g}_k\|^2} \right), \quad (8.25)$$

gde $\gamma_{k+1}^{F(M)SM}$ označava γ_{k+1}^{FSM} (odnosno γ_{k+1}^{FMSM}) u slučaju FSM (odnosno $FMSM$) metode i $\omega_k = 1$ (odnosno $\omega_k = \tau_k$).

Skalarna aproksimacija Hesijana omogućava da se izostavi pretpostavka da je funkcija f dva puta neprekidno diferencijabilna. Prema tome, umesto (4.34) i (4.33) koji pretpostavljaju da je funkcija f dva puta neprekidno diferencijabilna i uniformno konveksna, koristi se sledeći jednostavniji zahtev za $\gamma_k^{F(M)SM}$:

$$m \leq \gamma_k^{F(M)SM} \leq M, \quad 0 < m \leq 1 \leq M, \quad m, M \in \mathbb{R}. \quad (8.26)$$

Pored toga, funkcija $f : \mathbb{R}^n \rightarrow \mathbb{R}$ koja je dvaput neprekidno diferencijabilna i uniformno konveksna se svodi na uniformno konveksnu funkciju na $\mathbb{R}^n$.

Leme 8.4.1 i 8.4.2 procenjuje iterativno smanjenje funkcije f osigurano SM i MSM iteracijama, odnosno FSM i $FMSM$ iteracijama.

Lema 8.4.1 ([155, 86]). *Za dva puta neprekidno diferencijabilnu i uniformno konveksnu funkciju $f : \mathbb{R}^n \rightarrow \mathbb{R}$ i za niz $\{\mathbf{x}_k^{(M)SM}\}$ generisan pomoću $(M)SM$ iteracija (8.23) važi sledeća nejednakost*

$$f(\mathbf{x}_k^{(M)SM}) - f(\mathbf{x}_{k+1}^{(M)SM}) \geq \mu \|\mathbf{g}_k\|^2, \quad (8.27)$$

gde je

$$\mu = \min \left\{ \frac{\sigma}{M}, \frac{\sigma(1-\sigma)}{L} \beta \right\}. \quad (8.28)$$

Lema 8.4.2. *Neka je $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i uniformno konveksna funkcija i neka važi (8.26). Tada niz $\{\mathbf{x}_k^{F(M)SM}\}$ generisan pomoću iteracija (8.24) zadovoljava nejednakost*

$$f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM}) \geq \mu_{\nu_k} \|\mathbf{g}_k\|^2, \quad (8.29)$$

gde je

$$\mu_{\nu_k} = \min \left\{ \frac{\sigma \nu_k}{M}, \frac{\sigma(1-\sigma)}{L} \beta \right\}. \quad (8.30)$$

Dokaz. Iterativna šema (8.24) kojom su predstavljene iteracije FSM i $FMSM$ metode može se zapisati u obliku $\mathbf{x}_{k+1}^{F(M)SM} = \mathbf{x}_k^{F(M)SM} + \alpha_k \mathbf{d}_k$, gde je $\mathbf{d}_k = -\nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k$. Na osnovu uslova zaustavljanja koji se koriste u Algoritmu 2.2.3, sledi

$$f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM}) \geq -\sigma \alpha_k \mathbf{g}_k^T \mathbf{d}_k, \quad \forall k \in \mathbb{N}. \quad (8.31)$$

U slučaju $\alpha_k < 1$, koristeći (8.31) i $\mathbf{d}_k = -\nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k$ dobija se

$$f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM}) \geq -\sigma \alpha_k \mathbf{g}_k^T \mathbf{d}_k = -\sigma \alpha_k \mathbf{g}_k^T \left(-\nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k \right). \quad (8.32)$$

Sada, iz (4.37) imamo da je

$$\begin{aligned} \alpha_k &\geq -\frac{\beta(1-\sigma)}{L} \cdot \frac{\mathbf{g}_k^T \mathbf{d}_k}{\|\mathbf{d}_k\|^2} = -\frac{\beta(1-\sigma)}{L} \cdot \frac{\mathbf{g}_k^T \left(-\nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k \right)}{\left\| -\nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k \right\|^2} \\ &= \frac{\beta(1-\sigma)}{L} \cdot \frac{\nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \|\mathbf{g}_k\|^2}{\nu_k^2 \omega_k^2 (\gamma_k^{F(M)SM})^{-2} \|\mathbf{g}_k\|^2} \\ &= \frac{\beta(1-\sigma)}{L} \cdot \frac{\gamma_k^{F(M)SM}}{\nu_k \omega_k}. \end{aligned}$$

Primenom poslednje nejednačine u (8.32) dobija se

$$\begin{aligned} f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM}) &\geq \sigma \alpha_k \nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k^T \mathbf{g}_k \\ &\geq \sigma \frac{\beta(1-\sigma)}{L} \cdot \frac{\gamma_k^{F(M)SM}}{\nu_k \omega_k} \nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k^T \mathbf{g}_k \\ &\geq \sigma \frac{(1-\sigma)\beta}{L} \|\mathbf{g}_k\|^2. \end{aligned}$$

Za slučaj $\alpha_k = 1$ i na osnovu (8.26) važi nejednakost

$$\begin{aligned} f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM}) &\geq -\sigma \mathbf{g}_k^T \mathbf{d}_k = -\sigma \mathbf{g}_k^T (-\nu_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k) \\ &= \frac{\sigma \nu_k}{\gamma_k^{F(M)SM}} \|\mathbf{g}_k\|^2 \\ &\geq \frac{\sigma \nu_k}{M} \|\mathbf{g}_k\|^2. \end{aligned}$$

Konačno, iz prethodne dve nejednačine dobijene za oba moguća slučaja $\alpha_k < 1$ i $\alpha_k = 1$ dobija se (8.29) čime je dokaz završen. $\square$

Napomena 8.4.1. Na osnovu (8.29), odnosno (8.27), sledi $f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM}) \in [\mu_{\nu_k} \|\mathbf{g}_k\|^2, +\infty)$ odnosno $f(\mathbf{x}_k^{(M)SM}) - f(\mathbf{x}_{k+1}^{(M)SM}) \in [\mu \|\mathbf{g}_k\|^2, +\infty)$. Iz (8.6) sledi $\mu_{\nu_k} \geq \mu$ ako je $f(\mathbf{x}_{k+1}^{F(M)SM}) < f(\mathbf{x}_k^{F(M)SM})$. Prema tome, $f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM}) \in [\mu_{\nu_k} \|\mathbf{g}_k\|^2, +\infty) \subseteq [\mu \|\mathbf{g}_k\|^2, +\infty)$. Ovo znači da vrednosti $f(\mathbf{x}_k^{F(M)SM}) - f(\mathbf{x}_{k+1}^{F(M)SM})$ pripadaju intervalu

sa vrednostima većim ili jednakim u odnosu na vrednosti intervala koji uključuje vrednosti $f(\mathbf{x}_k^{(M)SM}) - f(\mathbf{x}_{k+1}^{(M)SM})$. Štaviše, to znači da su mogućnosti za smanjenje $f(\mathbf{x}_{k+1}^{F(M)SM})$ u poređenju sa $f(\mathbf{x}_k^{F(M)SM})$ veće ili jednake mogućnostima za smanjenje $f(\mathbf{x}_{k+1}^{(M)SM})$ u poređenju sa $f(\mathbf{x}_k^{(M)SM})$.

Teorema 8.4.1 potvrđuje linearnu konvergenciju $F(M)SM$ metode za uniformno konveksne funkcije.

Teorema 8.4.1. *Neka je $f : \mathbb{R}^n \rightarrow \mathbb{R}$ dva puta neprekidno diferencijabilna i uniformno konveksna funkcija i neka važi (8.26). Ako je niz $\{\mathbf{x}_k\}$ generisan iteracija (8.24), odnosno Algoritmom 8.2.2, tada*

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{F(M)SM}\| = 0, \quad (8.33)$$

i niz $\{\mathbf{x}_k\}$ konvergira ka $\mathbf{x}^*$ najmanje linearno.

Dokaz. Dokaz je sličan dokazu u [155, Teorema 4.1] i biće izostavljen. $\square$

U Lemi 8.4.3 ispitujemo konvergenciju $F(M)SM$ metode na klasi strogo konveksnih kvadratnih funkcija

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T A \mathbf{x} - b^T \mathbf{x}, \quad (8.34)$$

gde je A realna $n \times n$ simetrična pozitivno definitna matrica, a $b \in \mathbb{R}^n$ vektor. Označimo sa $\lambda_1 \leq \dots \leq \lambda_n$ sortirane sopstvene vrednosti simetrične pozitivno definitne matrice A .

Gradijent funkcije f date u (8.34) je oblika

$$\mathbf{g}_k = A \mathbf{x}_k - b. \quad (8.35)$$

Lema 8.4.3. *Najmanja i najveća sopstvena vrednost simetrične pozitivno definitne matrice $A \in \mathbb{R}^{n \times n}$ strogo konveksne kvadratne funkcije f definisane pomoću (8.34) zadovoljavaju nejednakost*

$$\lambda_1 \leq \frac{\gamma_{k+1}^{F(M)SM}}{\alpha_{k+1}} \leq \frac{2\lambda_n}{\sigma}, \quad k \in \mathbb{N}, \quad (8.36)$$

gde je parametar $\gamma_k^{F(M)SM}$ određen na osnovu (8.25), a dužina koraka α_k je određena Algoritmom 2.2.3.

Dokaz. Razlika između dve uzastopne vrednosti funkcije f definisane pomoću (8.34) iznosi

$$f(\mathbf{x}_{k+1}^{F(M)SM}) - f(\mathbf{x}_k)^{F(M)SM} = \frac{1}{2} \mathbf{x}_{k+1}^T A \mathbf{x}_{k+1} - b^T \mathbf{x}_{k+1} - \frac{1}{2} \mathbf{x}_k^T A \mathbf{x}_k + b^T \mathbf{x}_k. \quad (8.37)$$

Zamenom (8.24) u (8.37) dobija se

$$\begin{aligned}
 f(\mathbf{x}_{k+1}^{F(M)SM}) - f(\mathbf{x}_k^{F(M)SM}) &= \frac{1}{2} \left[\mathbf{x}_k - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k \right]^T A \left[\mathbf{x}_k - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k \right] \\
 &\quad - b^T [\mathbf{x}_k - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k] - \frac{1}{2} \mathbf{x}_k^T A \mathbf{x}_k + b^T \mathbf{x}_k \\
 &= -\frac{1}{2} \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{x}_k^T A \mathbf{g}_k - \frac{1}{2} \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k^T A \mathbf{x}_k \\
 &\quad + \frac{1}{2} (\nu_k \alpha_k \omega_k)^2 (\gamma_k^{F(M)SM})^{-2} \mathbf{g}_k^T A \mathbf{g}_k + \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} b^T \mathbf{g}_k.
 \end{aligned}$$

Primenom (8.35) u prethodnoj jednačini, dobija se

$$\begin{aligned}
 f(\mathbf{x}_{k+1}^{F(M)SM}) - f(\mathbf{x}_k^{F(M)SM}) &= \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} [b^T \mathbf{g}_k - \mathbf{x}_k^T A \mathbf{g}_k] \\
 &\quad + \frac{1}{2} (\nu_k \alpha_k \omega_k)^2 (\gamma_k^{F(M)SM})^{-2} \mathbf{g}_k^T A \mathbf{g}_k \\
 &= \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} [b^T - \mathbf{x}_k^T A] \mathbf{g}_k \\
 &\quad + \frac{1}{2} (\nu_k \alpha_k \omega_k)^2 (\gamma_k^{F(M)SM})^{-2} \mathbf{g}_k^T A \mathbf{g}_k \\
 &= -\nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k^T \mathbf{g}_k \\
 &\quad + \frac{1}{2} (\nu_k \alpha_k \omega_k)^2 (\gamma_k^{F(M)SM})^{-2} \mathbf{g}_k^T A \mathbf{g}_k.
 \end{aligned} \tag{8.38}$$

Nakon zamene (8.38) u (8.25), parametar $\gamma_{k+1}^{F(M)SM}$ postaje

$$\begin{aligned}
 \gamma_{k+1}^{F(M)SM} &= 2\gamma_k^{F(M)SM} \frac{\gamma_k^{F(M)SM} \left(f(\mathbf{x}_{k+1}^{F(M)SM}) - f(\mathbf{x}_k^{F(M)SM}) \right) + \nu_k \alpha_k \omega_k \|\mathbf{g}_k\|^2}{(\nu_k \alpha_k \omega_k)^2 \|\mathbf{g}_k\|^2} \\
 &= 2\gamma_k^{F(M)SM} \frac{-\nu_k \alpha_k \omega_k \|\mathbf{g}_k\|^2 + \frac{1}{2} (\nu_k \alpha_k \omega_k)^2 (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k^T A \mathbf{g}_k + \nu_k \alpha_k \omega_k \|\mathbf{g}_k\|^2}{(\nu_k (\alpha_k \omega_k))^2 \|\mathbf{g}_k\|^2} \\
 &= 2\gamma_k^{F(M)SM} \frac{\frac{1}{2} (\nu_k \alpha_k \omega_k)^2 (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k^T A \mathbf{g}_k}{(\nu_k \alpha_k \omega_k)^2 \|\mathbf{g}_k\|^2} \\
 &= \frac{\mathbf{g}_k^T A \mathbf{g}_k}{\|\mathbf{g}_k\|^2}.
 \end{aligned}$$

Poslednja jednačina za $\gamma_{k+1}^{F(M)SM}$ predstavlja Rejljev količnik realne simetrične matrice A i vektora $\mathbf{g}_k$. Prema tome, na osnovu Leme 1.1.3 važi sledeća nejednakost

$$\lambda_1 \leq \gamma_{k+1}^{F(M)SM} \leq \lambda_n, \quad k \in \mathbb{N}. \tag{8.39}$$

Dokaz leve nejednakosti u (8.36) sledi neposredna iz (8.39), budući da je $\alpha_{k+1} \in (0, 1]$. Da bismo dokazali desnu nejednakost u (8.36), koristimo gornju granicu iniciranu linijskim

pretraživanjem

$$\alpha_k \geq \frac{\beta(1-\sigma)\gamma_k}{L},$$

što dovodi do sledeće nejednakosti:

$$\frac{\gamma_{k+1}^{F(M)SM}}{\alpha_{k+1}} < \frac{L}{\beta(1-\sigma)}. \quad (8.40)$$

Uzimajući u obzir (8.35) i da je realna matrica A simetrična, sledi da je:

$$\|\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{y})\| = \|A\mathbf{x} - b - (A\mathbf{y} - b)\| = \|A\mathbf{x} - A\mathbf{y}\| \leq \|A\| \|\mathbf{x} - \mathbf{y}\| = \lambda_n \|\mathbf{x} - \mathbf{y}\|.$$

Na osnovu poslednje nejednačine možemo zaključiti da Lipšicovu konstantu L u (8.40) možemo zameniti vrednošću najveće sopstvene vrednosti λ_n matrice A . Uzimajući u obzir način na koji su parametri linijskog pretraživanja unazad $\sigma \in (0, 0.5)$ i $\beta \in (\sigma, 1)$ definisani, sledi da je

$$\frac{\gamma_{k+1}^{F(M)SM}}{\alpha_{k+1}} < \frac{L}{\beta(1-\sigma)} = \frac{\lambda_n}{\beta(1-\sigma)} < \frac{2\lambda_n}{\sigma}. \quad (8.41)$$

Prema tome, dokazana je i desna strana nejednakosti (8.36), čime je dokaz završen. $\square$

U Teoremi 8.4.2 se dokazuje konvergencija $F(M)SM$ i $F(M)SM$ iteracija za strogo konveksne kvadratne funkcije pod dodatnom pretpostavkom $\lambda_n < 2\lambda_1$.

Teorema 8.4.2. *Neka je f strogo konveksna kvadratna funkcija oblika (8.34). Ako sopstvene vrednosti matrice A zadovoljavaju dodatnu pretpostavku $\lambda_n < 2\lambda_1$, onda $F(M)SM$ iteracije (8.24) zadovoljavaju nejednakost*

$$(d_i^{k+1})^2 \leq \delta^2 (d_i^k)^2, \quad (8.42)$$

gde je

$$\delta = \max \left\{ 1 - \frac{\sigma\lambda_1}{2\lambda_n}, \frac{\lambda_n}{\lambda_1} - 1 \right\}, \quad (8.43)$$

i

$$\lim_{k \rightarrow \infty} \|\mathbf{g}_k^{F(M)SM}\| = 0. \quad (8.44)$$

Dokaz. Neka je $\{\mathbf{x}_k\}$ niz vrednosti dobijen $F(M)SM$ iteracijama (8.24), odnosno primenom Algoritma 8.2.2 i pretpostavimo da su $\{v_1, \dots, v_n\}$ ortonormirani sopstveni vektori simetrične pozitivno definitne matrice A . U ovom slučaju, za proizvoljni vektor $\mathbf{x}_k$ u (8.35), postoje realne konstante $d_1^k, d_2^k, \dots, d_n^k$ takve da

$$\mathbf{g}_k = \sum_{i=1}^n d_i^k v_i. \quad (8.45)$$

Sada, koristeći (8.24) u (8.35) dobija se

$$\begin{aligned}
 \mathbf{g}_{k+1} &= A\mathbf{x}_{k+1} - b \\
 &= A\left(\mathbf{x}_k - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \mathbf{g}_k\right) - b \\
 &= A\mathbf{x}_k - b - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} A\mathbf{g}_k \\
 &= \mathbf{g}_k - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} A\mathbf{g}_k \\
 &= \left(I - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} A\right) \mathbf{g}_k.
 \end{aligned}$$

Dalje, koristeći (8.45) dobija se

$$\mathbf{g}_{k+1} = \sum_{i=1}^n d_i^{k+1} v_i = \sum_{i=1}^n \left(1 - \nu_k \alpha_k \omega_k (\gamma_k^{F(M)SM})^{-1} \lambda_i\right) d_i^k v_i. \quad (8.46)$$

Da bi dokazali (8.42), dovoljno je pokazati da $\left|1 - \frac{\lambda_i}{(\nu_k \alpha_k \omega_k)^{-1} \gamma_k^{F(M)SM}}\right| \leq \delta$. Postoje dva slučaja. U prvom slučaju, ako je $\lambda_i \leq \frac{\gamma_k^{F(M)SM}}{\nu_k \alpha_k \omega_k}$ koristeći (8.36), možemo zaključiti sledeće

$$1 > \frac{\lambda_i}{(\nu_k \alpha_k \omega_k)^{-1} \gamma_k^{F(M)SM}} \geq \frac{\sigma \lambda_1}{2\lambda_n} \implies 1 - \frac{\lambda_i}{(\nu_k \alpha_k \omega_k)^{-1} \gamma_k^{F(M)SM}} \leq 1 - \frac{\sigma \lambda_1}{2\lambda_n} \leq \delta. \quad (8.47)$$

Ispitajmo sada drugi slučaj $\frac{\gamma_k^{F(M)SM}}{\nu_k \alpha_k \omega_k} < \lambda_i$. Kako važi

$$1 < \frac{\lambda_i}{(\nu_k \alpha_k \omega_k)^{-1} \gamma_k^{F(M)SM}} \leq \frac{\lambda_n}{\lambda_1}, \quad (8.48)$$

sledi da

$$\left|1 - \frac{\lambda_i}{(\nu_k \alpha_k \omega_k)^{-1} \gamma_k^{F(M)SM}}\right| \leq \frac{\lambda_n}{\lambda_1} - 1 \leq \delta. \quad (8.49)$$

Sada koristimo ortonormalnost sopstvenih vektora $\{v_1, \dots, v_n\}$ i (8.45) i dobija se

$$\|\mathbf{g}_k\|^2 = \sum_{i=1}^n (d_i^k)^2. \quad (8.50)$$

Kako je (8.42) zadovoljeno i važi $0 < \delta < 1$, na osnovu (8.50), sledi da važi (8.45), čime je dokaz završen. $\square$

8.4.2 Ispitivanje konvergencije za FDL metod

Globalna konvergencija predložene varijante Dai–Liao konjugovano gradijentnog metoda pod nazivom fazi neutrosofski Dai–Liao (FDL) konjugovano gradijentni metod je dokazana na osnovu standardnih pretpostavki i Leme 6.4.2 date u Odeljku 6.4.2.

Lema 8.4.4. *Posmatrajmo predloženi fazi neutrosofski Dai–Liao konjugovano gradijentni metod definisan pomoću (2.2), (8.17) i (8.18). Ako pravac pretraživanja zadovoljava uslov (6.49) za svako $k \geq 0$, onda važi sledeća nejednakost*

$$\mathbf{g}_k^T \mathbf{d}_k \leq -c \|\mathbf{g}_k\|^2, \text{ za neko } c \geq 0. \quad (8.51)$$

Dokaz. Za $k = 0$ imamo da važi $\mathbf{g}_0^T \mathbf{d}_0 = -\|\mathbf{g}_0\|^2$. Koristeći $c = 1$, može se zaključiti da je nejednakost (8.51) zadovoljena za $k = 0$. Pretpostavimo da (8.51) važi za neko $k \geq 1$. Ukoliko pomnožite sa vektorom $\mathbf{g}_k^T$ levu i desnu stranu u (8.17), dobija se

$$\begin{aligned} \mathbf{g}_k^T \mathbf{d}_k &= -\|\mathbf{g}_k\|^2 + \beta_k^{\text{FDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= -\|\mathbf{g}_k\|^2 + \left(\frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - \nu_k \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right) \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= -\|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - \nu_k \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= -\|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - \nu_k \frac{\alpha_{k-1} \mathbf{g}_k^T \mathbf{d}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= -\|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - \nu_k \frac{\alpha_{k-1} (\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}. \end{aligned} \quad (8.52)$$

Koristeći (8.19) zajedno sa (6.49) i $\alpha_{k-1} > 0$ imamo da važi

$$\nu_k \frac{\alpha_{k-1} (\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \geq 0. \quad (8.53)$$

Sada iz (8.52) i (8.53) sledi

$$\begin{aligned} \mathbf{g}_k^T \mathbf{d}_k &\leq -\|\mathbf{g}_k\|^2 + \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} \\ &= \frac{-\|\mathbf{g}_k\|^2 (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2 + (\mathbf{g}_k^T \mathbf{y}_{k-1}) (\mathbf{g}_k^T \mathbf{d}_{k-1}) (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})}{(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2}. \end{aligned} \quad (8.54)$$

Primenom nejednakosti $\mathbf{u}^T \mathbf{v} \leq \frac{1}{2}(\|\mathbf{u}\|^2 + \|\mathbf{v}\|^2)$ na nejednakost (8.54), gde je $\mathbf{u} = \frac{1}{\sqrt{2}}(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}) \mathbf{g}_k$ i $\mathbf{v} = \sqrt{2}(\mathbf{g}_k^T \mathbf{d}_{k-1}) \mathbf{y}_{k-1}$, dobija se

$$\begin{aligned}
 \mathbf{g}_k^T \mathbf{d}_k &\leq \frac{-\|\mathbf{g}_k\|^2 (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2 + \frac{1}{2} \left(\left\| \frac{1}{\sqrt{2}} (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}) \mathbf{g}_k \right\|^2 + \left\| \sqrt{2} (\mathbf{g}_k^T \mathbf{d}_{k-1}) \mathbf{y}_{k-1} \right\|^2 \right)}{(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2} \\
 &= \frac{-\|\mathbf{g}_k\|^2 (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2 + \frac{1}{2} \left(\frac{1}{2} (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2 \|\mathbf{g}_k\|^2 + 2 (\mathbf{g}_k^T \mathbf{d}_{k-1})^2 \|\mathbf{y}_{k-1}\|^2 \right)}{(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2} \\
 &= \frac{-\|\mathbf{g}_k\|^2 (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2 + \frac{1}{4} (\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2 \|\mathbf{g}_k\|^2 + (\mathbf{g}_k^T \mathbf{d}_{k-1})^2 \|\mathbf{y}_{k-1}\|^2}{(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2} \\
 &= -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{(\mathbf{g}_k^T \mathbf{d}_{k-1})^2 \|\mathbf{y}_{k-1}\|^2}{(\mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2} \\
 &\leq -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{\alpha_{k-1}^2 (\|\mathbf{g}_k^T \mathbf{d}_{k-1}\|)^2 \|\mathbf{y}_{k-1}\|^2}{(\alpha_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1})^2} \\
 &\leq -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{\alpha_{k-1}^2 \|\mathbf{g}_k\|^2 \|\mathbf{d}_{k-1}\|^2 \|\mathbf{y}_{k-1}\|^2}{(\mathbf{s}_{k-1}^T \mathbf{y}_{k-1})^2} \\
 &\leq -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{\|\mathbf{g}_k\|^2 \|\alpha_{k-1} \mathbf{d}_{k-1}\|^2 \|\mathbf{y}_{k-1}\|^2}{\theta^2 \|\mathbf{s}_{k-1}\|^4} \\
 &\leq -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{\|\mathbf{g}_k\|^2 \|\mathbf{s}_{k-1}\|^2 \|\mathbf{y}_{k-1}\|^2}{\theta^2 \|\mathbf{s}_{k-1}\|^4} \\
 &\leq -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{\|\mathbf{g}_k\|^2 \|\mathbf{y}_{k-1}\|^2}{\theta^2 \|\mathbf{s}_{k-1}\|^2} \\
 &\leq -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{L^2 \|\mathbf{g}_k\|^2 \|\mathbf{s}_{k-1}\|^2}{\theta^2 \|\mathbf{s}_{k-1}\|^2} \\
 &= -\|\mathbf{g}_k\|^2 + \frac{1}{4} \|\mathbf{g}_k\|^2 + \frac{L^2}{\theta^2} \|\mathbf{g}_k\|^2 \\
 &= -\left(1 - \frac{1}{4} - \frac{L^2}{\theta^2}\right) \|\mathbf{g}_k\|^2 = -\left(\frac{3}{4} - \frac{L^2}{\theta^2}\right) \|\mathbf{g}_k\|^2.
 \end{aligned} \tag{8.55}$$

Oдавде sledi da je nejednakost (8.51) zadovoljena za $c = \left(\frac{3}{4} - \frac{L^2}{\theta^2}\right)$ u (8.55) i proizvoljno $k \geq 0$. $\square$

Teorema 8.4.3 potvrđuje globalnu konvergenciju FDL metode.

Teorema 8.4.3. *Neka važe pretpostavke (H_1) i (H_2) iz Stava 4.5.1 i neka je funkcija f uniformno konveksna. Onda za niz $\{\mathbf{x}_k\}$ generisan pomoću FDL metode određene sa (2.2), (8.17) i (8.18) važi*

$$\liminf_{k \rightarrow \infty} \|\mathbf{g}_k\| = 0. \tag{8.56}$$

Dokaz. Pretpostavimo suprotno. Kako (8.56) ne važi, onda inicira postojanje konstante $c_1 > 0$ takve da

$$\|\mathbf{g}_k\| \geq c_1, \text{ za svako } k \geq 0. \tag{8.57}$$

Kvadriranjem obe strane izraza (8.17) dobija se

$$\|\mathbf{d}_k\|^2 = \|\mathbf{g}_k\|^2 - 2\beta_k^{\text{FDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} + (\beta_k^{\text{FDL}})^2 \|\mathbf{d}_{k-1}\|^2. \tag{8.58}$$

Uzimajući u obzir (8.18), imamo da važi

$$\begin{aligned}
 -2\beta_k^{\text{FDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} &= -2 \left(\frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - \nu_k \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right) \mathbf{g}_k^T \mathbf{d}_{k-1} \\
 &= -2 \left(\frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - \nu_k \frac{\alpha_{k-1} \mathbf{g}_k^T \mathbf{d}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} \right) \\
 &= -2 \left(\frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \mathbf{g}_k^T \mathbf{d}_{k-1} - \nu_k \frac{\alpha_{k-1} (\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right).
 \end{aligned} \tag{8.59}$$

Sada iz (8.53), sledi da je

$$\begin{aligned}
 -2\beta_k^{\text{EDL}} \mathbf{g}_k^T \mathbf{d}_{k-1} &\leq 2 \left| \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right| |\mathbf{g}_k^T \mathbf{d}_{k-1}| \leq 2 \frac{\|\mathbf{g}_k\| \|\mathbf{y}_{k-1}\|}{|\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}|} \|\mathbf{g}_k\| \|\mathbf{d}_{k-1}\| \\
 &= 2 \frac{\alpha_{k-1} \|\mathbf{g}_k\|^2 \|\mathbf{y}_{k-1}\| \|\mathbf{d}_{k-1}\|}{\alpha_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\
 &= 2 \frac{\|\mathbf{g}_k\|^2 \|\mathbf{y}_{k-1}\| \|\mathbf{s}_{k-1}\|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \\
 &\leq 2 \frac{\|\mathbf{g}_k\|^2 L \|\mathbf{s}_{k-1}\| \|\mathbf{s}_{k-1}\|}{\theta \|\mathbf{s}_{k-1}\|^2} \\
 &= \frac{2L}{\theta} \|\mathbf{g}_k\|^2.
 \end{aligned} \tag{8.60}$$

Dalje, iz (8.18) imamo da važi

$$\begin{aligned}
 \beta_k^{\text{FDL}} &= \frac{\mathbf{g}_k^T \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} - \nu_k \frac{\mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} = \frac{\mathbf{g}_k^T \mathbf{y}_{k-1} - \nu_k \mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\
 &\leq \left| \frac{\mathbf{g}_k^T \mathbf{y}_{k-1} - \nu_k \mathbf{g}_k^T \mathbf{s}_{k-1}}{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \right| = \alpha_{k-1} \frac{|\mathbf{g}_k^T \mathbf{y}_{k-1} - \nu_k \mathbf{g}_k^T \mathbf{s}_{k-1}|}{\alpha_{k-1} \mathbf{d}_{k-1}^T \mathbf{y}_{k-1}} \\
 &= \alpha_{k-1} \frac{|\mathbf{g}_k^T \mathbf{y}_{k-1} - \nu_k \mathbf{g}_k^T \mathbf{s}_{k-1}|}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}} \\
 &\leq \alpha_{k-1} \frac{|\mathbf{g}_k^T \mathbf{y}_{k-1} - \nu_k \mathbf{g}_k^T \mathbf{s}_{k-1}|}{\theta \|\mathbf{s}_{k-1}\|^2} = \alpha_{k-1} \frac{|\mathbf{g}_k^T (\mathbf{y}_{k-1} - \nu_k \mathbf{s}_{k-1})|}{\theta \alpha_{k-1}^2 \|\mathbf{d}_{k-1}\|^2} \\
 &\leq \frac{\|\mathbf{g}_k\| (\|\mathbf{y}_{k-1}\| + \nu_k \|\mathbf{s}_{k-1}\|)}{\theta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &\leq \frac{\|\mathbf{g}_k\| (L \|\mathbf{s}_{k-1}\| + \nu_k \|\mathbf{s}_{k-1}\|)}{\theta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} = \frac{\|\mathbf{g}_k\| (L + \nu_k) \|\mathbf{s}_{k-1}\|}{\theta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{\|\mathbf{g}_k\| (L + \nu_k) \alpha_{k-1} \|\mathbf{d}_{k-1}\|}{\theta \alpha_{k-1} \|\mathbf{d}_{k-1}\|^2} \\
 &= \frac{(L + \nu_k) \|\mathbf{g}_k\|}{\theta \|\mathbf{d}_{k-1}\|}.
 \end{aligned} \tag{8.61}$$

Sada na osnovu (8.19) i (8.61) imamo da važi

$$\beta_k^{\text{FDL}} \leq \frac{(L + 1) \|\mathbf{g}_k\|}{\theta \|\mathbf{d}_{k-1}\|}. \tag{8.62}$$

Koristeći (8.60) i (8.62) u (8.58), dobija se

$$\begin{aligned}
 \|\mathbf{d}_k\|^2 &\leq \|\mathbf{g}_k\|^2 + \frac{2L}{\theta} \|\mathbf{g}_k\|^2 + \frac{(L+1)^2 \|\mathbf{g}_k\|^2}{\theta^2 \|\mathbf{d}_{k-1}\|^2} \|\mathbf{d}_{k-1}\|^2 \\
 &= \|\mathbf{g}_k\|^2 + \frac{2L}{\theta} \|\mathbf{g}_k\|^2 + \frac{(L+1)^2 \|\mathbf{g}_k\|^2}{\theta^2} \\
 &= \left(1 + \frac{2L}{\theta} + \frac{(L+1)^2}{\theta^2}\right) \|\mathbf{g}_k\|^2 \\
 &= \left(\frac{\theta + 2L}{\theta} + \frac{(L+1)^2}{\theta^2}\right) \|\mathbf{g}_k\|^2 \\
 &= \frac{(\theta + 2L)\theta + (L+1)^2}{\theta^2} \|\mathbf{g}_k\|^2.
 \end{aligned} \tag{8.63}$$

Zatim, podelivši obe strane nejednačine (8.63) sa $\|\mathbf{g}_k\|^4$ i koristeći (8.57), imamo da važi

$$\begin{aligned}
 \frac{\|\mathbf{d}_k\|^2}{\|\mathbf{g}_k\|^4} &\leq \frac{(\theta + 2L)\theta + (L+1)^2}{\theta^2} \cdot \frac{1}{c_1^2}, \text{ odnosno} \\
 \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} &\geq \frac{\theta^2 \cdot c_1^2}{(\theta + 2L)\theta + (L+1)^2}.
 \end{aligned} \tag{8.64}$$

Nejednačine u (8.64) impliciraju

$$\sum_{k=0}^{\infty} \frac{\|\mathbf{g}_k\|^4}{\|\mathbf{d}_k\|^2} \geq \sum_{k=0}^{\infty} \frac{\theta^2 \cdot c_1^2}{(\theta + 2L)\theta + (L+1)^2} = \infty. \tag{8.65}$$

Prema tome, $\|\mathbf{g}_k\| \geq c_1$ izaziva kontradikciju sa Lemom 6.4.2. Oдавde sledi da (8.56) važi, čime je dokaz završen. $\square$

8.5 Numerički rezultati i primena

U ovom odeljku dokazujemo numeričku efikasnost gradijentnih metoda baziranih na neutrosfskoj logici. Odeljak se sastoji od četiri pododeljka, gde se u prvom i četvrtom ispituje efikasnost predloženih metoda na osnovu profila performansi Dolan-a i Moré-a. U drugom pododeljku se ispituje efikasnost optimizacionih metoda na osnovu rangiranja, a u trećem pododeljku je data primena fazi metoda optimizacije u regresionoj analizi.

Svi kodovi algoritama korišćeni u numeričkim eksperimentima za testirane metode su napisani u Matlab R2017a programskom jeziku i izvršeni na laptop računaru sa Intel Core i3 2.0 GHz CPU-om, 8 GB RAM-a i Windows 10 operativnim sistemom.

Algoritmi testiranih metoda se upoređuju korišćenjem linijskog pretraživanja unazad sa parametrima $\sigma = 0.0001$ i $\beta = 0.8$ i kriterijumima zaustavljanja

$$\|\mathbf{g}_k\| \leq \epsilon \quad \text{i} \quad \frac{|f_{k+1} - f_k|}{1 + |f_k|} \leq \delta,$$

gde je $\epsilon = 10^{-6}$ i $\delta = 10^{-16}$. Specifični parametri koji se koriste samo u *FSM*, *FGD* i *FMSM* metodama su dati u Tabeli 8.2, dok specifični parametri koji se koriste u *FDL* metodi su dati u Tabeli 8.3.

Numerička testiranja su bazirana na standardnim test funkcijama sa datim početnim tačkama iz [7, 26], pri čemu se metode upoređuju na osnovu tri kriterijuma: broj iterativnih koraka (broj iteracija), broj izračunavanja vrednosti funkcije u različitim tačkama (broj evaluacija funkcije) i procesorsko vreme, odnosno vreme izvršavanja algoritma (CPU vreme).

8.5.1 Numeričkih rezultati za *FSM*, *FGD* i *FMSM* metode

U ovom odeljku dati su numerički rezultati za *FSM*, *FGD* i *FMSM* metode bazirani na neutrosofskoj logici, a zatim su upoređeni sa numeričkim rezultatima dobijeni za *MSM*, *SM* i *GD* metode. Metode koje učestvuju u testiranjima su predstavljene u Tabeli 8.1. Za svaku test funkciju prilikom testiranja urađeno je 10 numeričkih testiranja sa različitim izborom broja promenljivih: 100, 500, 1000, 3000, 5000, 7000, 8000, 10000, 15000 i 20000.

Zbirni numerički rezultati za *MSM*, *SM*, *GD*, *FSM*, *FGD* i *FMSM* metode, dobijeni testiranjem 30 test funkcija (odnosno 300 testova), dati su u Tabelama 8.4–8.6. Tabele 8.4–8.6 uključuju numeričke rezultate dobijene praćenjem tri kriterijuma: broj iteracija, broj evaluacija funkcije i CPU vreme.

Profil performansi dati u [58] su primenjeni za upoređivanje numeričkih rezultata dobijenih *MSM*, *SM*, *GD*, *FSM*, *FGD* i *FMSM* metodama za broj iteracija, broj evaluacija funkcije i CPU vreme.

Na Slici 8.6 upoređujemo profile performansi *MSM*, *SM*, *GD*, *FSM*, *FGD* i *FMSM* metode za broj iteracija i broj evaluacija funkcije bazirani na numeričkim vrednostima datim u Tabelama 8.4 i 8.5. Detaljna analiza numeričkih rezultat otkriva da *FMSM* metod rešava 20.00% test problema sa najmanjim brojem iteracija u poređenju sa *MSM* (33.33%), *SM* (26.67%), *FSM* (33.33%), *GD* (13.33%) i *FGD* (10.00%). Na Slici 8.6a, vidi se da grafik *FMSM* metode prvi dostiže najviši nivo, odnosno vrednost 1, čime se potvrđuje da je *FMSM* metod efikasniji od ostalih metoda u odnosu na broj iteracija.

Sa Slike 8.6b vidi se da su *FMSM* i *FSM* metode efikasnije od *MSM*, *SM*, *GD* i *FGD* metode u odnosu na broj evaluacija funkcije, jer su *FMSM* (10.00%) i *FMS* (33.33%) rešili test probleme sa najmanjim brojem evaluacija funkcije u poređenju sa *MSM* (40.00%), *SM* (26.67%), *GD* (13.33%) i *FGD* (6.67%). Sa Slike 8.6b, može se primetiti da grafici *FMSM* i *FSM* metoda prvi dolaze na vrh, odnosno dostižu vrednost 1, tako da su *FMSM* i *FSM* pobednici u odnosu na broj evaluacija funkcije. S druge strane, *GD* i *FGD* su najsporije metode.

Slika 8.7 prikazuje profil performansi testiranih metoda baziranih na CPU vremenu za numeričke vrednosti date u Tabeli 8.6. Metod *FMSM* uspešno rešava 23.33% test problema sa najmanjim CPU vremenom u poređenju sa *MSM* (30.00%), *SM* (23.33%), *FSM* (23.33%), *GD* (6.67%) i *FGD* (0%). Prema Slici 8.7, grafici *FMSM* i *FSM* metode prvi dostižu gornji granični nivo 1, što potvrđuje njihovu dominaciju u odnosu na CPU vreme.

Tabela 8.4: Zbirni numerički rezultati MSM , SM , GD , FSM , FGD i $FMSM$ metoda za broj iterativnih koraka.

Test funkcija	Broj iteracija					
	MSM	$FMSM$	SM	FSM	GD	FGD
Extended Penalty	651	377	549	372	1255	1250
Perturbed Quadratic	44419	75431	77458	74473	372356	369992
Raydan 1	12965	12437	15913	11035	58743	58594
Raydan 2	90	87	90	94	67	129
Diagonal 1	52527	11571	8955	12189	41208	42290
Diagonal 2	26215	24866	30912	29957	543249	543054
Diagonal 3	7545	12586	13892	13050	62128	61072
Hager	28073	800	839	817	3104	2956
Generalized Tridiagonal 1	290	440	270	376	656	665
Extended TET	130	248	130	225	1974	1856
Extended quadratic penalty QP1	328	189	246	177	563	549
Extended quadratic penalty QP2	1538	2105	3302	3564	134401	122926
Quadratic QF2	44911	14203	83957	11488	409859	411364
Extended quadratic exponential EP1	87	100	64	109	496	528
Extended tridiagonal 2	568	421	419	415	1145	1099
Almost perturbed quadratic	44029	78452	80559	79793	374841	375518
ENGVAL1 (CUTE)	363	298	302	291	573	557
QUARTC (CUTE)	185	216	246	211	524612	524612
Diagonal 6	90	87	90	95	67	129
Generalized quartic	150	150	157	238	1453	1751
Diagonal 7	124	113	90	136	543	570
Diagonal 8	100	86	103	89	583	573
Diagonal 9	16920	17221	11487	17752	195362	195155
HIMMELH (CUTE)	100	90	100	90	90	90
Extended Rosenbrock	50	50	50	50	50	50
Extended BD1 (block diagonal)	189	204	191	223	650	682
NONDQUAR (CUTE)	42	39	42	35	33	30
DQDRTIC (CUTE)	827	635	1263	497	15320	15398
Extended Beale	480	980	639	831	12834	12826
EDENSCH (CUTE)	337	314	275	275	663	705

Štaviše, GD i FGD su najsporije metode.

Na osnovu podataka datih u Tabelama 8.4–8.6 i profila performansi na Slikama 8.6 i 8.7, uočava se da su $FMSM$ i FSM metode postigle bolji rezultat u poređenju sa MSM , SM , GD i FGD metodama u pogledu tri osnovna kriterijuma: broj iteracija, broj evaluacija funkcije i CPU vreme.

Tabela 8.7 sadrži prosečne vrednosti za broj iteracija, broj evaluacija funkcije i CPU vreme za 300 numeričkih testova.

Prosečni rezultati u Tabeli 8.7 potvrđuju da su prosečni rezultati za $FMSM$ i FSM metodu manji u odnosu na odgovarajuće vrednosti za MSM , SM , GD i FGD metodu. Takvo posmatranje nas dovodi do zaključka da upotreba neutrosofske logike u gradijentnim

Tabela 8.5: Zbirni numerički rezultati *MSM*, *SM*, *GD*, *FSM*, *FGD* i *FMSM* metoda za broj evaluacija testirane funkcije.

Test funkcija	Broj evaluacija funkcije					
	<i>MSM</i>	<i>FMSM</i>	<i>SM</i>	<i>FSM</i>	<i>GD</i>	<i>FGD</i>
Extended Penalty	3527	2585	2394	2388	47378	48057
Perturbed quadratic	257063	438335	439924	423195	16171466	16069927
Raydan 1	89508	69791	87508	61595	1667238	1658647
Raydan 2	190	233	190	235	144	291
Diagonal 1	526958	56914	47874	58155	1615828	1664760
Diagonal 2	158515	144005	171300	166567	1086508	1086118
Diagonal 3	41528	71024	76336	70540	2407025	2364254
Hager	271940	3402	3308	3165	56824	54818
Generalized tridiagonal 1	1012	1587	931	1445	10867	11432
Extended TET	440	681	440	601	19800	18859
Extended quadratic penalty QP1	1918	1992	2507	1842	10771	11268
Extended quadratic penalty QP2	10731	14285	24234	26528	3875768	3545317
Quadratic QF2	245407	102882	465615	80626	19072367	19141623
Extended quadratic exponential EP1	807	604	587	830	13643	14852
Extended tridiagonal 2	2550	2123	2285	2111	9570	9464
Almost perturbed quadratic	259487	452388	452360	445028	16285621	16309931
ENGVAL1 (CUTE)	1974	2700	2098	2315	8787	8593
QUARTC (CUTE)	420	492	542	472	1049274	1049304
Diagonal 6	229	335	229	263	158	332
Generalized quartic	409	470	423	781	19062	25071
Diagonal 7	458	547	293	1094	3348	4286
Diagonal 8	326	462	980	612	3921	4078
Diagonal 9	141781	90948	71353	89023	8449946	8455412
HIMMELH (CUTE)	210	190	210	190	190	190
Extended Rosenbrock	110	110	110	110	110	110
Extended BD1 (Block Diagonal)	558	696	598	691	7660	8452
NONDQUAR (CUTE)	2084	2085	2057	2060	2500	2501
DQDRTIC (CUTE)	4090	2805	6518	2542	395014	400147
Extended Beale	2200	4720	3277	3416	207852	208551
EDENSCH (CUTE)	1198	1213	956	872	9403	10615

metodama omogućava poboljšanje numeričkih rezultata.

8.5.2 Rangiranje optimizacionih metoda

U ovom pododeljku, performanse (broj iteracija, broj evaluacija funkcije, CPU vreme) *GD*, *SM*, *MSM*, *FGD*, *FSM* i *FMSM* metode optimizacije prilikom testiranja 30 test funkcija date u Tabelama 8.4 – 8.6 su rangirane od najbolje do najgore, odnosno rang vrednostima od 1 do 6. Nakon određivanja ranga za svaku test funkciju za svaki metod, potrebno je izračunati konačni rang metode. Konačni rang metode se bazira na proseku rangova dobijenih za svaki metod u odnosu na posmatrane test funkcije. Metod sa najmanjim

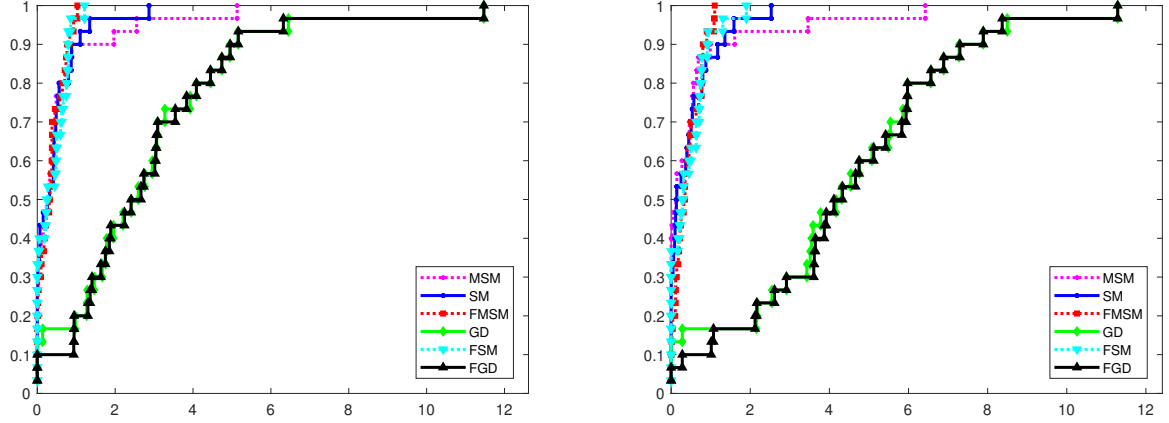
Tabela 8.6: Zbirni numerički rezultati MSM , SM , GD , FSM , FGD i $FMSM$ metoda za CPU vreme u sekundama.

Test funkcija	CPU vreme (sec)					
	MSM	$FMSM$	SM	FSM	GD	FGD
Extended penalty	3.734	1.969	1.969	1.844	17.672	19.078
Perturbed quadratic	167.063	323.266	298.813	317.250	10163.688	9771.406
Raydan 1	46.813	35.141	50.953	30.234	727.281	667.094
Raydan 2	0.453	0.281	0.281	0.344	0.250	0.531
Diagonal 1	522.703	86.500	59.297	99.953	1836.766	2091.281
Diagonal 2	236.531	228.188	271.094	276.281	2105.219	2158.156
Diagonal 3	75.484	172.250	139.859	157.594	3842.625	4025.688
Hager	384.438	9.594	9.453	9.250	116.922	118.609
Generalized tridiagonal 1	2.656	3.188	2.000	3.797	11.641	14.875
Extended TET	0.953	1.313	0.906	1.359	15.922	16.281
Extended quadratic penalty QP1	1.688	1.625	1.875	1.578	4.203	4.391
Extended quadratic penalty QP2	5.844	9.891	7.203	10.516	746.328	770.500
Quadratic QF2	124.344	47.875	243.688	35.359	7611.656	8436.359
Extended quadratic exponential EP1	0.969	0.594	0.469	1.109	5.281	7.297
Extended tridiagonal 2	1.906	1.313	1.609	1.266	3.359	3.766
Almost perturbed quadratic	135.484	314.953	238.625	267.750	9271.016	13902.047
ENGVAL1 (CUTE)	2.031	1.797	1.844	1.828	4.125	4.422
QUARTC (CUTE)	2.813	2.984	3.250	3.219	6253.828	8032.547
Diagonal 6	0.328	0.219	0.344	0.484	0.203	0.438
Generalized quartic	0.344	0.266	0.438	0.625	6.766	11.922
Diagonal 7	0.953	0.797	0.531	1.813	3.672	4.406
Diagonal 8	0.781	0.922	1.797	1.047	5.578	4.469
Diagonal 9	249.875	74.484	53.234	77.219	2478.422	2705.781
HIMMELH (CUTE)	0.797	0.594	0.781	0.797	0.609	0.641
Extended Rosenbrock	0.203	0.094	0.156	0.203	0.219	0.141
Extended BD1 (block diagonal)	0.766	0.766	0.859	0.969	4.984	4.469
NONDQUAR (CUTE)	7.266	8.891	7.797	9.047	9.406	10.406
DQDRTIC (CUTE)	2.516	1.500	2.906	1.500	118.250	127.844
Extended Beale	7.219	18.734	9.766	16.016	488.328	546.359
EDENSCH (CUTE)	6.141	6.422	4.016	5.063	24.672	36.766

Tabela 8.7: Prosečni numerički rezultati za 30 test funkcije testiranih na 10 različitim numeričkih testova.

Prosečne performanse	MSM	$FMSM$	SM	FSM	GD	FGD
Broj iteracija	9,477.43	8,493.20	11,086.33	8,631.57	91,962.60	91,565.67
Broj evaluacija funkcije	67,587.60	49,020.13	62,247.90	48,309.73	2,416,934.77	2,406,242.00
CPU vreme (sec)	66.44	45.21	47.19	44.51	1,529.30	1,783.27

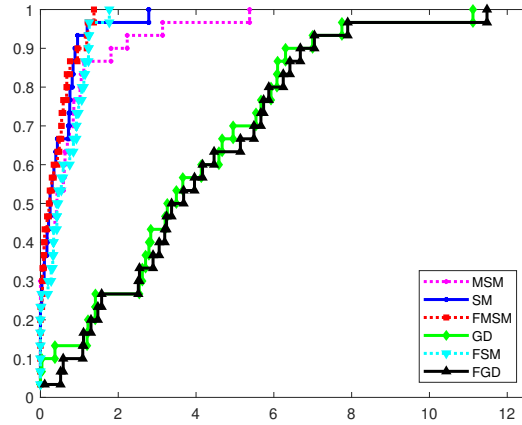
prosekom ima najviši rang, odnosno rang vrednost 1, dok metod sa najvećim prosekom ima najniži rang, odnosno rang vrednost 6. Sa n_m (odnosno n_{tf}) označavamo broj metoda



(a) Bazirani na broju iteracija.

(b) Bazirani na broju evaluacija funkcije.

Slika 8.6: Profili performansi za *MSM*, *SM*, *GD*, *FSM*, *FGD* i *FMSM* metod.



Slika 8.7: Profili performansi bazirani na CPU vremenu za *MSM*, *SM*, *GD*, *FSM*, *FGD* i *FMSM* metod.

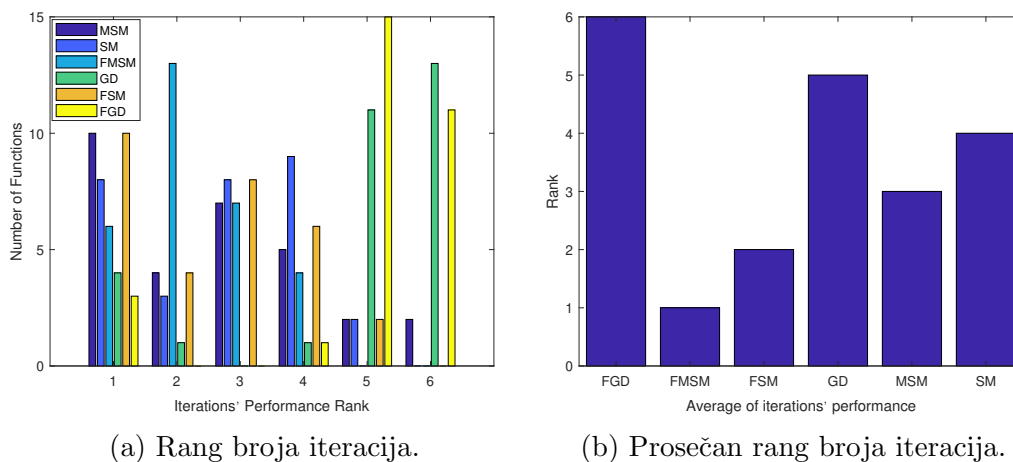
(odnosno broj test funkcija) korišćenih u ovom upoređivanju. Neka je dat skup metoda M i skup test funkcija F , rang metode x na test funkciji y je definisan sa $r_{x,y}$. U našem slučaju, $r_{x,y}$ predstavlja rang metode x za posmatranu test funkciju y i može imati rang vrednosti od 1 do 6. Prosečan rang metode $x \in M$ se izračunava na sledeći način:

$$AR_x = \frac{\sum_{y \in F} r_{x,y}}{n_{tf}},$$

gde AR_x predstavlja prosek svih rangova posmatrane metode x . Konačni prosečni rang u našem slučaju dobija se kada se svi prosečni rangovi posmatrane performanse (broj iteracija,

broj evaluacija funkcije, CPU vreme) rangiraju od najboljeg do najgoreg, odnosno rang vrednostima od 1 do 6.

Slika 8.8 prikazuje rang broja iteracija GD , SM , MSM , FGD , FSM i $FMSM$ metoda optimizacije na 30 test funkcija i njihov prosečan rang broja iteracija. Metod dobija rang vrednost 1 ukoliko ima najmanji broj iteracija od svih posmatranih metoda. Ako metod ima drugu najmanju vrednost za broj iteracija u poređenju sa posmatranim metodama onda dobija rang vrednost 2 itd. Konkretno, Slika 8.8a prikazuje broj test funkcija u kojima je svaki metod rangiran i dobio rang vrednost 1, 2, 3 itd., dok Slika 8.8b prikazuje konačni rang metode baziran na proseku rezultata prikazanih na Slici 8.8a.

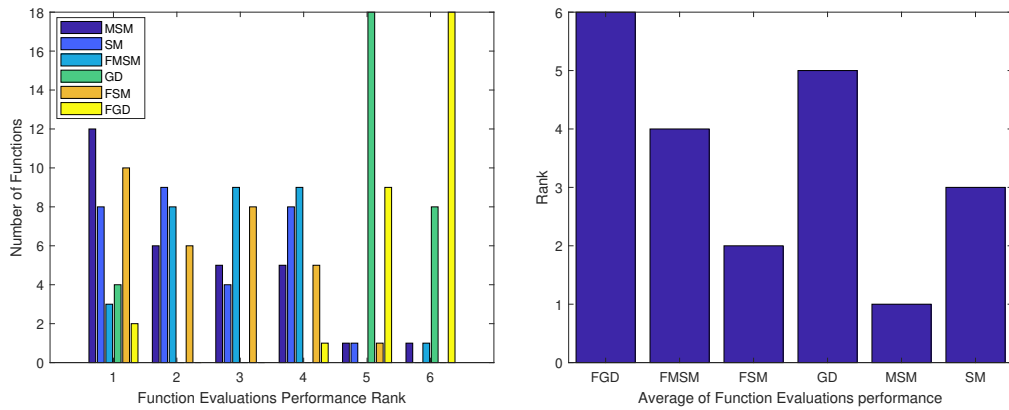


Slika 8.8: Rang broja iteracija za GD , SM , MSM , FGD , FSM i $FMSM$ metod na 30 test funkcija i njihov prosečan rang broja iteracija.

Na Slici 8.8a MSM metod je ostvario rang vrednost 1 u istom ili većem broju test funkcija od FSM i $FMSM$ metoda. Međutim, kako MSM metod ima rang vrednost 6 u više test funkcija od FSM i $FMSM$ metoda, zbog toga na Slici 8.8b MSM metod ima prosečnu rang vrednost 3, dok FSM metod ima prosečnu rang vrednost 2, a $FMSM$ metod ima prosečnu rang vrednost 1. Drugim rečima, $FMSM$ metod je efikasniji i bolji od FSM i MSM metode u pogledu broja iteracija. Štaviše, činjenica da $FMSM$ i FSM metode nadmašuju svoje odgovarajuće originalne MSM i SM metode je još jedan bitan zaključak na osnovu podataka sa Slike 8.8b.

Slika 8.9 prikazuje rang broja evaluacija funkcija GD , SM , MSM , FGD , FSM i $FMSM$ metoda optimizacije na 30 test funkcija i njihov prosečan rang broja evaluacija funkcija. Metod dobija rang vrednost 1 ako ima najmanji broj evaluacija funkcije od svih posmatranih metoda. Ako metod ima drugu najmanju vrednost za broj evaluacija funkcije u poređenju sa posmatranim metodama onda dobija rang vrednost 2 itd. Konkretno, Slika 8.9a prikazuje broj test funkcija u kojima je svaki metod rangiran i dobio rang vrednost od 1 do 6, dok Slika 8.9b prikazuje konačni rang broja evaluacija funkcija baziran na proseku rezultata prikazanih na Slici 8.9a.

Na Slici 8.9a MSM metod je ostvario rang vrednost 1 u većem broju test funkcija od



(a) Rang broja evaluacija funkcija. (b) Prosečan rang broja evaluacija funkcija.

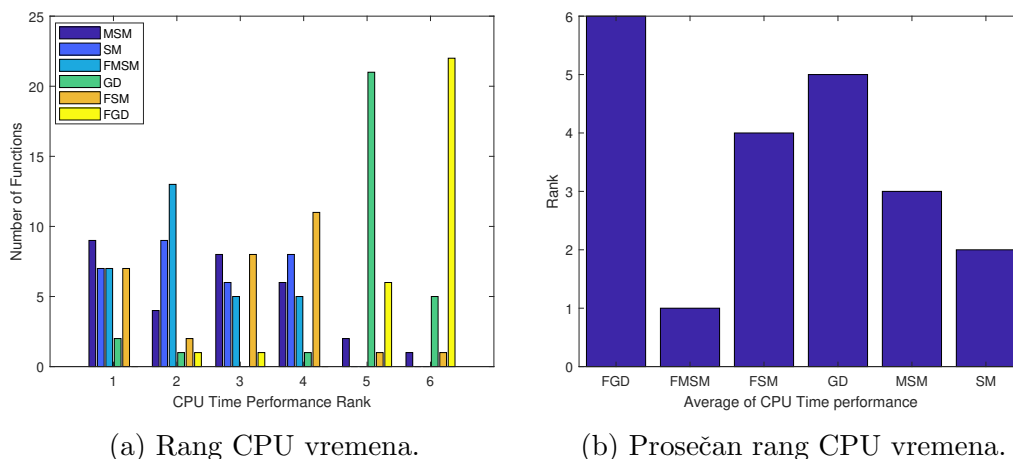
Slika 8.9: Rang broja evaluacija funkcija za GD , SM , MSM , FGD , FSM i $FMSM$ metod na 30 test funkcija i njihov prosečan rang broja evaluacija funkcija.

svih posmatranih metoda, dok je FGD metod postigao rang vrednost 6 u većem broju test funkcija od svih posmatranih metoda. Na osnovu podataka sa Slike 8.9a sledi da MSM metod ima prosečnu rang vrednost 1, a FGD metod ima prosečnu rang vrednost 6 na Slici 8.9b. Na osnovu ovog zaključka sledi da MSM metod nadmašuje sve posmatrane metode u pogledu broja evaluacija funkcije. Štaviše, činjenica je da fazi FSM metod nadmašuje originalni SM metod, što je još jedan bitan zaključak na osnovu podataka sa Slike 8.9b.

Slika 8.10 prikazuje rang potrošnje CPU vremena za posmatrane metode optimizacije na 30 test funkcija i njihov prosečan rang potrošnje CPU vremena. Metod dobija rang vrednost 1 ukoliko ima najmanje CPU vremena u poređenju sa svim posmatranim metodama. Metod dobija rang vrednost 2, ako ima drugu najmanju vrednost za CPU vremena u poređenju sa svim posmatranim metodama, i tako dalje. Konkretno, Slika 8.10a prikazuje broj test funkcija u kojima je svaki metod rangiran i dobio rang vrednost od 1 do 6, dok Slika 8.10b prikazuje konačni rang metoda baziranih na osnovu proseka rezultata prikazanih na Slici 8.10a.

Na Slici 8.10a MSM metod ima rang vrednost 1 u većem broju test funkcija od svih posmatranih metoda, dok FGD metod ima rang vrednost 6 u većem broju test funkcija od svih posmatranih metoda. Kao rezultat toga, MSM metod ima prosečnu rang vrednost 3, a FGD metod ima prosečnu rang vrednost 6 na Slici 8.10b. Ako pogledamo Sliku 8.10b, možemo uočiti da $FMSM$ metod nadmašuje sve posmatrane metode u pogledu potrošnje CPU vremena.

Da rezimiramo, sve fazi metode ($FMSM$, FSM , FGD) odlično rade u pronalaženju minimuma na 30 test funkcija. Generalno, $FMSM$ metod ima najbolje performanse kada je u pitanju broj iteracije, MSM metod ima najbolje performanse kada je u pitanju broj evaluacija funkcija, a $FMSM$ metod ima najbolje performanse kada je u pitanju potrošnja CPU vremena. U nastavku korišćena je sledeća notacija $\mathcal{M}_i \prec \mathcal{M}_j$ da označi da je metod $\mathcal{M}_i$



Slika 8.10: Rang potrošnje CPU vremena za GD , SM , MSM , FGD , FSM i $FMSM$ metod na 30 test funkcija i njihov prosećan rang potrošnje CPU vremena.

bolje rangiran od $\mathcal{M}_j$ metoda prilikom analize dobijenih podataka na Slikama 8.8b, 8.9b i 8.10b:

- Slika 8.8b navodi na zaključak: $FMSM \prec FSM \prec MSM \prec SM \prec GD \prec FGD$;
- Slika 8.9b navodi na zaključak: $MSM \prec FSM \prec SM \prec FMSM \prec GD \prec FGD$;
- Slika 8.10b navodi na zaključak: $FMSM \prec SM \prec MSM \prec FSM \prec GD \prec FGD$.

Na osnovu prethodne analize može se uočiti interesantan zaključak da je $GD \prec FGD$ i da se nalaze na poslednjim pozicijama po svim kriterijumima. Posebno interesantno zapažanje je da predloženi fazi parametar ν_k poboljšava SM i MSM metode, ali nije pogodan za GD metodu. Na osnovu ovoga sledi zaključak da fazi parametar ν_k nije poželjno koristiti u ulozi izolovanog parametra, ali je poželjno koristiti ga u kombinaciji sa drugim parametrima.

8.5.3 Primena fazi metoda optimizacije u regresionoj analizi

Regresiona analiza je važan statistički alat koji se koristi u ekonomiji, menadžmentu, fizici, finansijama i mnogim drugim oblastima. Ovaj alat koristi se za proučavanje interakcije između nezavisnih i zavisnih promenljivih različitih skupova podataka. Opšta funkcija regresione analize je oblika

$$y = f(x_1, x_2, \dots, x_k + \epsilon), \quad (8.66)$$

gde su $x_i, i = 1, 2, \dots, k, k > 0$ nezavisne promenljive, y je zavisna promenljiva i ϵ je greška. Linearna regresiona funkcija dobija se pravolinijskim odnosom između x i y

$$y = a_0 + a_1x_1 + a_2x_2 + \dots + a_kx_k + \epsilon, \quad (8.67)$$

gde su $a_0, a_1, \dots, a_k$ parametri regresije. Glavni cilj regresione analize je da proceni parametre $a_0, a_1, \dots, a_k$ tako da greška ϵ bude minimalna. Međutim, linearna regresiona šema se

retko koristi. Stoga se češće koristi nelinearna regresiona šema. U ovom odeljku razmatra se model kvadratne regresije. Metod najmanjih kvadrata je najpopularniji pristup uklapanja linija regresije i definisan je pomoću

$$y = a_0 + a_1x + a_2x^2. \quad (8.68)$$

Greške za skup podataka (x_i, y_i) , $i = 1, 2, \dots, n$ su definisane na sledeći način

$$E_i(a) = y_i - (a_0 + a_1x_i + a_2x_i^2), \quad a = (a_0, a_1, a_2). \quad (8.69)$$

Glavni cilj je da se “najbolja” linija uklopi u podatke kako bi se minimizirao zbir kvadrata zaostalih grešaka za sve dostupne podatke

$$\min_{a \in \mathbb{R}^3} \sum_{i=1}^n E_i^2(a), \quad a = (a_0, a_1, a_2). \quad (8.70)$$

Skup podataka u Tabeli 8.8 predstavlja broj poginulih osoba u saobraćajnim nezgodama u Srbiji od 2012. do 2021. godine. Ovaj skup podataka je definisan na osnovu godišnjih izveštaja Agencije za bezbednost saobraćaja Republike Srbije. Redni broj godine prikupljanja podataka označava se promenljivom x , a broj poginulih osoba u saobraćajnim nezgodama u Srbiji promenljivom y . Štaviše, podaci od 2012. do 2020. godine uzimaju se u obzir za uklapanje podataka, dok se podaci iz 2021. godine koristili za analizu greške.

Tabela 8.8: Broj poginulih osoba u saobraćajnim nezgodama u Srbiji od 2012. do 2021. godine.

Godina	Redni broj podatka (x)	Broj poginulih osoba (y)
2012	1	688
2013	2	650
2014	3	536
2015	4	599
2016	5	607
2017	6	579
2018	7	548
2019	8	534
2020	9	492
2021	10	521

Metod najmanjih kvadrata, FMSM, FSM i FGD metode koriste se za podešavanje regresionih modela za prikupljene podatke. Metod najmanjih kvadrata se često koristi za rešavanje linearnih sistema kod kojih je obično broj jednačina veći od broja nepoznatih [56]. Pri čemu, metod najmanjih kvadrata uključuje određivanje najbolje aproksimativne linije upoređivanjem ukupne greške najmanjih kvadrata.

Aproksimativna funkcija za nelinearni metod najmanjih kvadrata dobijena korišćenjem podataka u Tabeli 8.8 je definisana na sledeći način:

$$f(x) = 0.5303030303031x^2 - 24.1030303030320x + 685.1666666666750. \quad (8.71)$$

Za više detalja o tome kako se izračunava aproksimativna funkcija (8.71) pogledati u [56]. Neka je x_i redni broj godine, a y_i broj poginulih osoba u saobraćajnim nezgodama u toj godini. Onda se metod najmanjih kvadrata (8.68) transformiše u sledeći bezuslovni problem minimizacije:

$$\min_{a \in \mathbb{R}^3} f(a) = \min_{a \in \mathbb{R}^3} \sum_{i=1}^n E_i^2(a) = \sum_{i=1}^n (y_i - (a_0 + a_1 x_i + a_2 x_i^2))^2, \quad a = (a_0, a_1, a_2). \quad (8.72)$$

gde je $n = 9$, tj. i ima vrednosti od 1 do 9, što odgovara godinama od 2012. do 2020. godine. Podaci od 2012. do 2020. godine se koriste za definisanje nelinearnog kvadratnog modela za metod najmanjih kvadrata i odgovarajuće test funkcije bezuslovnog problema optimizacije. Međutim, podaci za 2021. su isključeni iz funkcije bezuslovne optimizacije kako bi se kasnije koristili za izračunavanje relativnih grešaka prognoziranih podataka. Relativna greška se izračunava korišćenjem formule za merenje preciznosti regresionog modela:

$$\text{Relativna greška} = \frac{|\text{Tačna vrednost} - \text{Aproksimativna vrednost}|}{|\text{Tačna vrednost}|}. \quad (8.73)$$

Regresioni model sa najmanjom relativnom greškom smatra se najboljim.

Primena konjugovano gradijentnih metoda u regresionoj analizi na probleme optimizacije u pronalaženju parametara regresije $a_0, a_1, \dots, a_k$ razmatrana je u [118, 157, 158, 159]. Da bi prevazišli poteškoće u izračunavanju vrednosti parametara a_0, a_1 i a_2 pomoću inverzne matrice korišćene su predložene FMSM, FSM i FGD metode za određivanje minimuma test funkcije (8.72) i rezultati su predstavljeni u Tabeli 8.9.

Statistika poginulih osoba u saobraćajnim nezgodama u Srbiji procenjuje se primenom predloženih metoda: FMSM, FSM, FGD, najmanjih kvadrata i trend linija (*eng. trend line*). Linija trenda je iscertana na osnovu stvarnih podataka dobijenih iz Tabele 8.8 koristeći Microsoft Excel i prikazana je na Slici 8.11. Jednačina za liniju trenda je u obliku nelinearne kvadratne jednačine

$$y = 0.5303x^2 - 24.103x + 685.17. \quad (8.74)$$

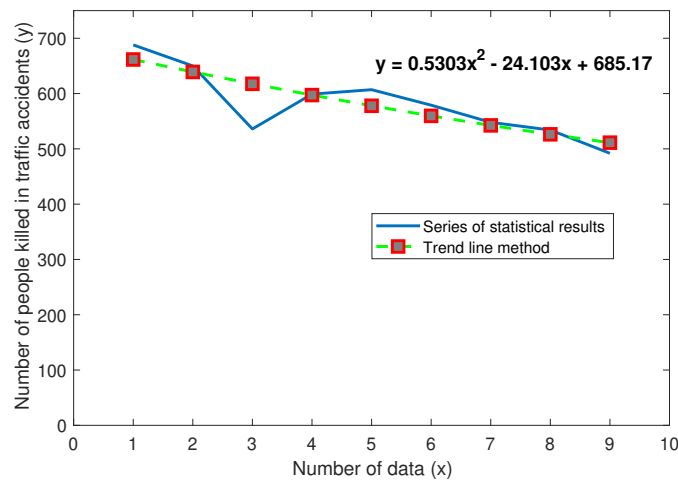
Ako uporedimo aproksimativne funkcije (8.71) i (8.74), kao i parametre regresije iz Tabele 8.9 dobijeni primenom FMSM, FSM i FGD metoda, možemo uočiti da su razlike vrlo male u vrednostima parametara a_0, a_1 i a_2 .

Funkcije linije trenda (8.74) i metoda najmanjih kvadrata (8.71) upoređuju se sa aproksimativnim funkcijama za FMSM, FSM i FGD metode dobijeni zamenom vrednosti parametara a_0, a_1 i a_2 u (8.68) za početnu tačku (1, 1, 1).

Primarni cilj regresione analize je da proceni parametre $a_0, a_1, \dots, a_k$ tako da je greška

Tabela 8.9: Numerički rezultati testiranja kvadratnog modela optimizacije pomoću FMSM, FSM i FGD metoda.

Metod	Početna tačka	NI	NFE	CPUts	Parametri regresije (a_0, a_1, a_2)		
					a_0	a_1	a_2
FMSM	(1,1,1)	28998	119898	1.484	685.1666325046	-24.10301448708	0.5303014926346
FSM	(1,1,1)	29612	120545	1.609	685.1666666295	-24.10303028897	0.5303030290905
FGD	(1,1,1)	173004	7861471	35.125	685.1617699647	-24.10091438736	0.5301142381300
FMSM	(5,5,5)	29791	126449	1.750	685.1666270050	-24.1029965382	0.5302990602898
FSM	(5,5,5)	29504	119706	1.406	685.1666666595	-24.10303030199	0.5303030302900
FGD	(5,5,5)	172876	7855584	36.812	685.1617455218	-24.10090383598	0.5301132197720
FMSM	(-1,-1,-1)	29259	120695	1.484	685.1666667610	-24.10303034254	0.5303030337903
FSM	(-1,-1,-1)	29513	119912	1.328	685.1663883598	-24.10291004492	0.5302924830427
FGD	(-1,-1,-1)	173698	7893030	37.797	685.1619870722	-24.10100820579	0.5301225799428



Slika 8.11: Nelinearna kvadratna linija trenda poginulih osoba u saobraćajnim nezgodama u Srbiji.

ϵ minimalna. U Tabeli 8.10 može se uočiti da predloženi FMSM, FSM i FGD metodi imaju slične relativne greške u poređenju sa metodom najmanjih kvadrata i linijom trenda.

Dakle, možemo zaključiti da su predloženi FMSM, FSM i FGD metodi pogodni za situacije iz stvarnog života.

8.5.4 Numeričkih rezultati za FDL metod

U ovom odeljku su analizirani numerički rezultati dobijeni FDL algoritmom i upoređeni sa numeričkim rezultatima dobijenim EDL [87] i DL [42] algoritmima. Implementacija FDL metode je zasnovana na Algoritmu 8.3.1, dok je implementacija EDL i DL metode zasnovana

Tabela 8.10: Tačka procene i relativna greška za podatak iz 2021. godine.

Metod	Tačka procene	Relativna greška
FMSM	497.16664	0.045745419
FSM	497.16667	0.045745362
FGD	497.16405	0.045750384
Najmanjeg kvadrata	497.16667	0.045745361
Linija trenda	497.17000	0.045738964

na algoritmima datim u [87] i [42].

Numerička testiranja su izvršena na 50 test funkcija preuzetih iz [7] i [26] sa dimenzijama u opsegu od 100 do 20000 promenljivih. Sva tri testirana algoritma počinju iz iste početne tačke $\mathbf{x}_0$ za svaku test funkciju. Svaka test funkcija se testira 10 puta sa progresivnim uvećanjem dimenzije (broja promenljivih) $n \in \{100, 500, 1000, 3000, 5000, 7000, 8000, 10000, 15000, 20000\}$.

Zbirni numerički rezultati za DL, FDL i EDL algoritme dobijeni testiranjem 50 test funkcija dati su u Tabeli 8.11. Tabela 8.11 prikazuje zbirne numeričke rezultate za sva tri kriterijuma (broj iteracija, broj evaluacija funkcije i CPU vreme) koji odgovaraju za svaku pojedinačnu test funkciju za DL, FDL i EDL algoritme.

Da bismo vizuelno uporedili performanse posmatranih metoda, koristili smo tehniku profila performansi [58] za numeričke rezultate koji odgovaraju kriterijumima broj iteracija, broj evaluacija funkcije i CPU vreme generisanim pomoću DL, FDL i EDL metode. Gornji grafikon u profilu performansi odgovara metodi koji pokazuje najbolje performanse. Vertikalna osa svakog profila performansi na slikama označava procenat test funkcija za koje je posmatrani metod pobednik među upoređenim metodama, pri čemu desna strana odgovara procentu uspešno rešenih test funkcija.

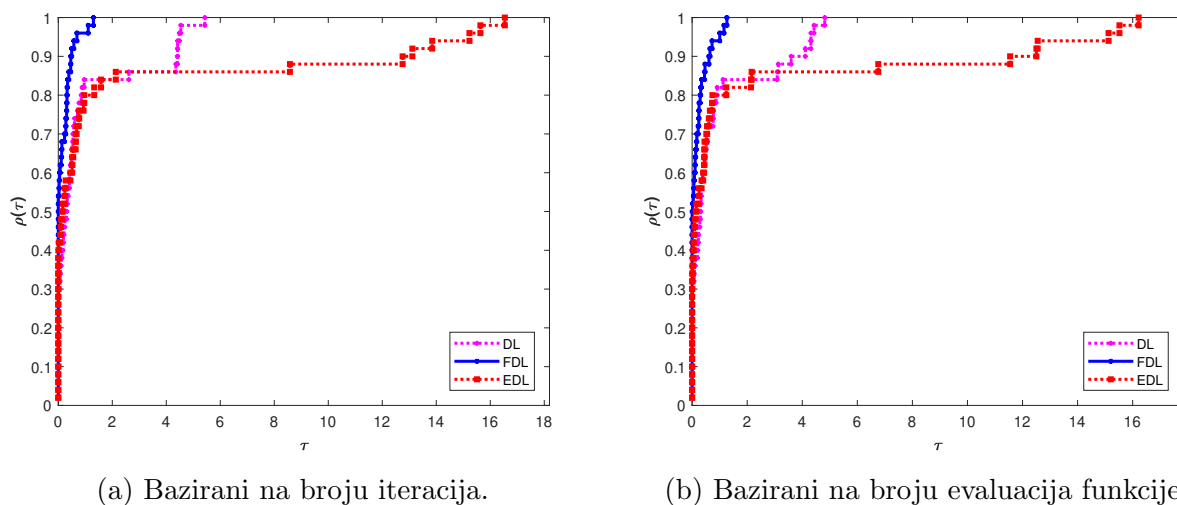
Slike 8.12 i 8.13 prikazuju profile performansi za podatke date u Tabeli 8.11. Grafikoni na Slici 8.12 ilustruju profile performansi za broj iteracija i broj evaluacija funkcije za DL, FDL i EDL iteracije na osnovu podataka iz Tabele 8.11. Na Slici 8.12a primetno je da DL, FDL i EDL metode mogu da reše sve test funkcije. Numeričko iskustvo pokazuje da FDL metod daje najbolje rezultate u 54.0% (27 od 50) test funkcija u poređenju sa DL (26.0% (13 od 50)) i EDL (38.0% (19 od 50)). Na Slici 8.12a grafik FDL metode prvi dodiruje vrh, tako da je FDL metod najbolji u odnosu na druge ispitivane metode kada je u pitanju kriterijum broj iteracija.

Slika 8.12b pokazuje da je FDL metod najefikasniji i da uspešno rešava sve test funkcije. Pored toga, dobijeni numerički rezultati potvrđuju da FDL u većini slučajeva radi dobro. Konkretno, FDL metod je najbrži jer rešava oko 48.0% (24 od 50) testiranih funkcija za najmanji broj evaluacija funkcije u poređenju sa DL i EDL metodama. Dok, DL i EDL metode rešavanju 22.0% (11 od 50) odnosno 30.0% (15 od 50) test funkcija za najmanji broj evaluacija funkcije. Dakle, FDL metod je superiorniji u poređenju sa DL i EDL metodama za date test funkcije.

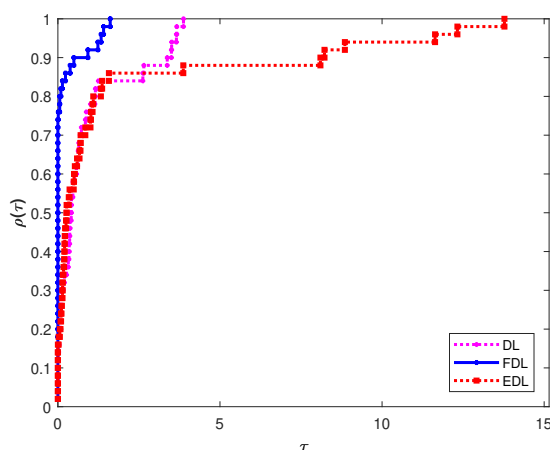
Tabela 8.11: Zbirni numerički rezultati za DL, FDL i EDL algoritam za broj iterativnih koraka (NI), broj evaluacija testirane funkcije (NFE) i CPU vreme u sekundama (CPUts).

Test funkcija	DL	FDL	EDL
	NI/NFE/CPUts	NI/NFE/CPUts	NI/NFE/CPUts
Extended Penalty	1905/77578/32.44	1610/62534/24.27	2304/82602/39.34
Perturbed Quadratic	14555/606750/379.36	10800/440213/206.5	10012/408474/248.5
Raydan 1	4337/114595/98.98	5497/122843/76.81	4194/109164/96.94
Raydan 2	1427/2864/3.19	67/144/0.28	2572540/5145090/894.45
Diagonal 1	5809/223750/245.58	5488/212491/227.78	4673/178295/219.11
Diagonal 3	5247/196745/423.77	4531/168162/307.59	4596/171636/366.2
Hager	1742/31516/103.67	1242/22799/47.06	1940/33206/98.77
Generalized Tridiagonal 1	2058/32313/49.5	2160/32033/27.5	2161/33285/44.7
Extended Tridiagonal 1	310/2932/8.39	182/2501/6.3	308/4129/12.77
Extended TET	1140/9840/11.03	619/5808/5.48	749/6362/5.97
Diagonal 5	1394/2798/6.94	60/130/0.61	3053907/6107824/3124.88
Extended Himmelblau	50/2431/1.02	51/2602/0.81	50/2413/0.94
Perturbed quadratic diagonal	1837/69156/18.45	1261/36785/13.88	2157/86977/34.8
Quadratic QF1	13895/526995/187.31	21989/846402/376.16	10199/379554/122.84
Extended quadratic penalty QP1	1080/17440/9.92	1524/23840/8.3	1157/18043/9.11
Extended quadratic penalty QP2	218/9479/11.05	112/5513/4.95	218/9194/8.91
Quadratic QF2	19211/847031/348.78	18861/816310/225.89	15555/689736/250.89
Extended quadratic exponential EP1	1254/3443/3.17	56/404/0.52	21431/43829/7.53
Extended Tridiagonal 2	22468/998473/549.48	3668/114169/87.44	10989/510713/93.61
TRIDIA (CUTE)	33278/1647913/967.23	40156/1977068/950.55	29133/1428866/675.42
ARWHEAD (CUTE)	1624/81625/44.88	1529/72379/31.59	1219/57140/28.67
Almost Perturbed Quadratic	14904/621925/259.8	19675/829784/357.36	13201/543372/188.05
LIARWHD (CUTE)	30/2705/1.28	30/2732/1.25	30/2739/1.44
POWER (CUTE)	532442/44419504/16742.67	580790/48609979/17435.61	629342/52431424/23630.78
ENGVAL1 (CUTE)	2489/33103/13.78	2400/32299/10.72	1975/27260/12.92
INDEF (CUTE)	21/1924/2.13	26/2238/2.5	30/2610/4.27
Diagonal 6	1583/3197/4.53	74/185/0.36	7052401/14105032/5037.22
DIXON3DQ (CUTE)	320921/1775846/1083.28	229757/1368033/727.17	257451/1517252/1045.33
COSINE (CUTE)	20/1600/1.89	20/1697/1.89	20/1700/2
BIGGSB1 (CUTE)	249919/1400798/832.38	259475/1549293/810.77	236612/1389720/945.67
Generalized Quartic	866/11273/3.98	1099/8951/4.06	959/10662/3.13
Diagonal 7	1453/4564/6.88	68/162/0.47	469477/940686/140.17
Diagonal 8	1371/3962/5.36	67/199/0.42	594522/1193760/195.09
Full Hessian FH3	2237/6202/7.13	52/513/0.69	767988/1537759/188.47
Diagonal 9	3312/138545/225.72	5344/217150/224.91	4520/189307/260.45
HIMMELH (CUTE)	20/1690/4.8	20/1758/4.53	20/1760/4.89
FLETCHCR (CUTE)	303212/10189775/5073.69	300227/10011849/4704.13	289670/9702961/4411.45
Extended BD1 (Block Diagonal)	1597/16783/7.63	1227/15639/5.88	1200/12605/6.63
Extended Maratos	72/3366/1.19	50/2069/0.72	40/1975/0.75
Extended Cliff	234/2992/2.08	217/6000/4.89	950/13187/6.19
Extended Hiebert	70/7215/1.94	70/7220/1.83	70/7228/1.86
NONDIA (CUTE)	33/3066/1.38	30/2829/1.27	32/3031/1.63
NONDQUAR (CUTE)	58/4652/18.05	45/3666/17.22	86/4989/19.02
DQDRTIC (CUTE)	3456/87105/26.45	2327/59047/16.41	3637/92315/34.95
Extended Freudenstein and Roth	1376/46597/10.73	3390/111830/28.52	2018/66654/16.17
Generalized Rosenbrock	282948/8410218/4125.52	280440/8335396/4088.55	281792/8373946/4055.17
Extended White and Holst	76/5794/9.22	50/3171/7.28	59/4022/11.56
Extended Beale	118/6791/14.05	72/3118/5.91	181/4748/6.75
EG2 (CUTE)	507/29388/47.55	697/48512/119.88	811/39769/122.47
EDENSCH (CUTE)	1694/23160/89.45	2089/27821/83.27	1684/22731/116.84

Slika 8.13 prikazuje profile performanse bazirani na CPU vremenu za DL, FDL i EDL metode. Primetno je da su DL, FDL i EDL u stanju da reše sve test funkcije. Dalja



Slika 8.12: Profili performansi za DL, FDL i EDL algoritme.



Slika 8.13: Profili performansi bazirani na CPU vremenu za DL, FDL i EDL algoritme.

ispitivanja dovode do zaključka da je FDL metod najbolji u 74.0% (37 od 50) test funkcija u poređenju sa DL (12.0% (6 od 50)) i EDL (16.0 % (8 od 50)) metodama. Analizirajući grafike na Slici 8.13, jasno je da FDL graf prvi dostiže vrh, odnosno vrednost 1, što potvrđuje njegovu superiornost u pogledu CPU vremena.

Opšti zaključak iz rezultata predstavljenih u Tabeli 8.11 je da je jedan od DL, FDL i EDL metoda apsolutni pobednik u 32 od 50 testiranih funkcija u odnosu na sva tri kriterijuma (broj iteracija, broj evaluacija funkcije i CPU vreme). Dalja analiza ovih podataka dovodi do sledećih zaključaka:

- DL metod je apsolutni pobednik u 5 slučajeva;
- FDL metod je apsolutni pobednik u 21 slučaju;
- EDL metod je apsolutni pobednik u 6 slučajeva.

Pored ovih činjenica, podaci dati u Tabeli 8.11 pokazuju da je FDL metod u 6 slučaja bolji od druga dva metoda u pogledu broj iteracija. Dalje, FDL metod je bolji u 3 slučaja u odnosu na DL i EDL metodu u pogledu broj evaluacija funkcije. I na kraju FDL metod postiže najbolje rezultate u 16 test slučajeva u pogledu CPU vreme, a da nije apsolutni pobednik u posmatranom test slučaju.

Poređenja DL, FDL i EDL metode su prikazana na Slikama 8.12 i 8.13. Kao što se može primetiti, FDL je superiorniji u odnosu na DL i EDL metode po sva tri kriterijuma (iteracije, evaluacije funkcije i CPU vreme). Štaviše, numerički rezultati u Tabeli 8.11 pokazuju da je FDL metod apsolutni pobednik. Dakle, numerički eksperimenti ukazuju na efikasnost predloženog FDL metoda, a samim tim i na značaj upotrebe neutrosofske logike u konjugovano gradijentnim metodama.

Poglavlje 9

Zaključna razmatranja i predlozi za buduća istraživanja

Istraživanja prezentovana u ovoj doktorskoj disertaciji donela su značajni napredak u poboljšanju svojstva gradijentnih i konjugovano gradijentnih algoritama za rešavanje problema bezuslovne optimizacije i sistema nelinearnih monotonih jednačina velikih dimenzija. Neke od glavnih prednosti prezentovanih algoritama su njihovi mali zahtevi za memorijom i dobra svojstva konvergencije. Algoritmi ubrzanog opadajućeg gradijenta, hibridnog ubrzanog opadajućeg gradijenta, algoritmi bazirani na Dai–Liao konjugovano gradijentnom metodu, kao i algoritmi koji kombinuju neutrosofsku logiku sa algoritmima ubrzanog opadajućeg gradijenta i Dai–Liao konjugovano gradijentnim metodom doprineli su opštem unapređenju gradijentnih i konjugovano gradijentnih metoda. Skoro svi novodefinisani algoritmi karakterišu se uvođenjem novih parametara ili novih pravila za određivanje vrednosti postojećih parametara.

U okviru četvrtog poglavlja urađena su četiri istraživanja. Prvo istraživanje kreće od ubrzanog opadajućeg gradijenta (AGD), gde se kroz uvođenje određenih modifikacija u iteracijama dolazi do modifikovanog ubrzanog opadajućeg gradijentnog metoda (MAGD). Drugo istraživanje nadovezuje se na prvo, ali kreće od SM metode, odnosno od poboljšanog opadajućeg gradijenta i dolazi se do modifikovanog poboljšanog opadajućeg gradijentnog metoda, odnosno do modifikovane SM metode (MSM). Treće istraživanje nadovezuje se na prethodna dva istraživanja i uz pomoć Picard–Mann hibridnog iterativnog procesa dolazi se do hibridnog modifikovanog ubrzanog gradijentnog metoda (HMAGD) i hibridnog modifikovanog poboljšanog gradijentnog metoda (HMSM). I na kraju četvrto istraživanje u ovom poglavlju nadovezuje se na drugo istraživanje, gde se uz pomoć višestruke upotrebe linijskog pretraživanja unazad radi na poboljšanju efikasnosti MSM metode. U okviru četvrtog istraživanja dati su DMSM i TMSM algoritmi. Metode prezentovane u ovom poglavlju generalno imaju osobinu da u svojim formulacijama sadrže izraz $\alpha_k + \alpha_k^2 - \alpha_k^3$, parametar ubrzanja γ_k ili parametar θ_k i za hibridne metode parametar korekcije Ω_k kao multiplikativni faktor vrednosti iterativnog koraka. Određivanje ovih parametara, kako je moglo da se vidi iz predstavljenih analiza, kada su u pitanju parametri α_k i θ_k bazirano je na Algoritmima

2.2.3 i 3.5.2, određivanje parametar ubrzanja γ_k bazirano je na Tejlorovom razvoju drugog reda i adekvatnoj aproksimaciji inverza Hesijana, a za hibridne metode parametar korekcije Ω_k određuje se uzimanjem odgovarajuće vrenosti. Globalna konvergencija predloženih algoritama je dokazana, numeričko poređenje korišćenjem test funkcija pokazalo je da je predloženi algoritam za MSM metodu najefikasniji među novim predloženim metodama, kao i od svih posmatranih metoda u ovom poglavlju.

U okviru šestog poglavlja ove disertacije predložena su četiri algoritma (MLSCD, MMDL, EDL i MSMDL). Dva su hibridna konjugovano gradijentna algoritma, gde je MLSCD nastao kombinacijom klasičnih LS i CD konjugovano gradijentnih metoda i odgovarajućeg pravca pretraživanja $\mathbf{d}_k$, a MMDL algoritam je nastao kombinacijom modifikovanih Dai–Liao konjugovano gradijentnih metoda DHSDL i DLSDL sa upotrebom odgovarajućeg pravca pretraživanja $\mathbf{d}_k$. Dok EDL i MSMDL predstavljaju nove varijante Dai–Liao konjugovano gradijentnog metoda. Kada je u pitanju EDL algoritam on se bazira na novom pravilu za određivanje parametra t u Dai–Liao konjugovano gradijentnom metodu. Što se tiče MSMDL algoritma tu imamo izjednačavanje pravca pretraživanja iz MSM metode sa pravcem pretraživanja u Dai–Liao metodi sa ciljem određivanja novog pravila za izračunavanje vrednosti parametra t . Za sve algoritme dokazana je globalna konvergencija i numerička efikasnost, dok je za MSMDL metod data i primena u 2D robotskoj kontroli pokreta.

Sedmo poglavlje se nadovezuje na šesto poglavlje, ali sa ciljem da se MSMDL algoritam proširi uz pomoć projekativnog metoda u cilju rešavanja sistema nelinearnih monotoničnih jednačina velikih dimenzija. U okviru ovog poglavlja prezentovan je MSMDLPM algoritam za koji je dokazana globalna konvergencija. Numerički rezultati potvrđuju da je MSMDLPM algoritam najefikasniji od posmatranih algoritama za rešavanje sistema nelinearnih monotoničnih jednačina. Takođe, kroz primenu MSMDLPM algoritma u procesu restauracije slika samo se potvrđuje efikasnost predloženog algoritma.

U osmom poglavlju je napravljen i najznačajniji iskorak u disertaciji. Uspostavljena je veza između neutrosofske logike i gradijentnih i konjugovano gradijentnih algoritama u cilju unapređenja metoda za rešavanje problema bezuslovne optimizacije. Urađena su dva istraživanja iz kojih su nastali FGD, FSM, FMSM i FDL algoritmi. FGD, FSM i FMSM algoritmi se baziraju na korišćenju prilagodljivog neutrosofskog logičkog parametra ν_k umesto fiksnog parametra korekcije $\Omega_k + 1$ u određivanju odgovarajućih dužina koraka, dok se u FDL algoritmu neutrosofsko logički parametar ν_k koristi umesto parametra t_k u Dai–Liao metodi. Za oba ova pristupa, odnosno za FMSM i FDL algoritme dokazana je globalna konvergencija i numerička efikasnost. Takođe, data je primena FGD, FSM i FMSM metoda u regresionoj analizi.

Uvereni smo da će predložene metode poslužiti kao motivacija za dalja istraživanja u cilju poboljšanja svojstva gradijentnih i konjugovano gradijentnih algoritama za rešavanje problema bezuslovne optimizacije i sistema nelinearnih monotoničnih jednačina velikih dimenzija vođenih neutrosofskom logikom.

Buduća naučna istraživanja u ovoj oblasti mogu se nastaviti u nekoliko pravaca. Pre svega, moguće su druge strategije u neutrosofikaciji i deneutrosofikaciji, kao i drugi okviri

paralelni neutrosofskim skupovima, poznati kao fazi skupovi slika i sferni fazi skupovi, o kojima se diskutuje u [109, 166]. O njihovoj primeni se može razgovarati u budućim istraživanjima.

Prethodna istraživanja su pokazala efikasnost neutrosofskog principa u metodama opadajućeg gradijenta i Dai–Liao klasi konjugovano gradijentnih metoda. Izazov je primeniti takav princip na druge metode nelinearne optimizacije. S druge strane, postoji širok spektar mogućnosti za definisanje novih pravila neutrosofikacije i deneutrosofikacije, koja se mogu razmotriti u budućim istraživanjima. Konačno, postoji odlična prilika da se poboljša mehanizam neutrosofskog zaključivanja koji se koristi u ovom istraživanju.

Empirijsko vrednovanje pokazuje visoku osetljivost rezultata na izbor parametara koji definišu funkcije pripadnost istini $T(u)$, pripadnost neistini $F(u)$ i pripadnost neodređenosti $I(u)$. Takvo iskustvo potvrđuje pretpostavku da drugačija konfiguracija parametara, kao i poboljšanja u neutrosofskom logičkom mehanizmu, mogu dovesti do daljeg poboljšanja definisanih metoda. Mogućnost da se definišu if–then pravila na sofisticiraniji način bazirana na istoriji dobijenih vrednosti funkcije cilja $f(\mathbf{x})$ ostaje otvorena tema za buduća istraživanja.

Evidentno je da pronalaženje novih pristupa u definisanju novih pravila za određivanje vrednosti β_k parametra ažuriranja konjugovanog gradijenta su nepresušne teme za naučna istraživanja i naš pristup je samo jedan od mogućih pravaca u ovom istraživanju.

Na kraju posvetićemo se i realnoj primeni gradijentnih i konjugovano gradijentnih metoda. U poglavljima šest, sedam i osam data je primena u 2D robotskoj kontroli pokreta, procesu restauracije slika i regresionoj analizi. Međutim, ovo je samo jedan mali deo mogućih primena gradijentnih i konjugovano gradijentnih metoda u realnim problemima. U budućim istraživanjima potrebno je pored razvoja novih algoritama posvetiti se i njihovoj primeni na realnim problemima. Ukoliko pogledamo razvoj mašinskog učenja može se uočiti da se sve češće koriste gradijentni i konjugovano gradijentni metodi u optimizaciji modela mašinskog učenja. Takođe, njihova primena je sve češća i u ekonomiji i finansijama, odnosno konkretno u optimizaciji portfolija i modelovanju potražnje i predikcije cena.

Naredna oblast u kojoj bi gradijentni i konjugovano gradijentni algoritmi mogli da se primene je energetika. Kada je u pitanju energetika, primenu bi mogli da nađu u optimizaciji mreže za električnu energiju u smislu optimizacije raspodele opterećenja i minimizaciji gubitka energije. Zato sva buduća istraživanja treba usmeriti na primenu gradijentnih i konjugovano gradijentnih algoritama u realnim problemima i to ne samo u navedenim oblastima već i u mnogim drugim oblastima, jer su osnova modernih optimizacionih sistema i omogućavaju efikasno rešavanje problema gde se traži minimum ili maksimum neke funkcije.

Spisak algoritama

2.1.1 Njutnov metod	13
2.1.2 Njutnov metod sa linijskim pretraživanjem	14
2.2.1 Algoritam tačnog linijskog pretraživanja	17
2.2.2 Algoritam netačnog linijskog pretraživanja	18
2.2.3 Linijsko pretraživanje unazad (<i>backtracking</i>)	19
3.1.1 Opšti algoritam kvazi-Njutnovog metoda	23
3.5.1 SM (Stanimirović-Miladinović) metod.	27
3.5.2 Izračunavanje vrednosti parametra θ_k u (3.28).	29
4.1.1 Modifikovani ubrzani opadajući gradijentni metod (MAGD metod)	33
4.2.1 Modifikovani SM metod (MSM metod)	35
4.3.1 Hibridni modifikovani ubrzani gradijentni (HMAGD) metod	37
4.3.2 Hibridni modifikovani poboljšani gradijentni metod (HMSM metod)	39
4.4.1 Drugo linijsko pretraživanje unazad za izračunavanje l_k	40
4.4.2 Treće linijsko pretraživanje unazad za izračunavanje j_k	41
4.4.3 Trostruka upotreba linijskog pretraživanja unazad u MSM metodi (TMSM metod)	41
4.4.4 Dvostruka upotreba linijskog pretraživanja unazad u MSM metodi (DMSM metod)	42
5.1.1 Opšti algoritam konjugovano gradijentnih metoda.	78
5.5.1 Tročlani H-BFGS-CG1 algoritam.	85
6.1.1 Mešoviti LS-CD konjugovano gradijentni (MLSCD) metod.	93
6.2.1 Mešoviti DHSDL-DLSDL konjugovano gradijentni (MMDL) metod.	95
6.3.1 Efikasan Dai-Liao (EDL) konjugovano gradijentni metod.	98
6.3.2 MSMDL metod	101
7.1.1 Dai-Liao projektni metod (DLPM).	142
7.2.1 MSM Dai-Liao projektni metod (MSMDLPM).	145
8.2.1 Opšti fazi metod opadajućeg pravca FDD	172
8.2.2 Fazi modifikovani SM metod (FMSM metod).	175
8.3.1 Fazi neutrosofski Dai-Liao (FDL) konjugovano gradijentni metod.	180

Spisak tabela

4.1	Zbirni numerički rezultati MAGD, HMAGD, MSM, HMSM i HSM metoda za broj iterativnih koraka.	54
4.2	Zbirni numerički rezultati MAGD, HMAGD, MSM, HMSM i HSM metoda za broj evaluacija testirane funkcije.	55
4.3	Zbirni numerički rezultati MAGD, HMAGD, MSM, HMSM i HSM metoda za CPU vreme u sekundama.	56
4.4	Prosečni numerički rezultati za 28 test funkcije testiranih na 12 različitih numeričkih testova.	56
4.5	Zbirni numerički rezultati MSM, HMSM i HSM metoda za broj iterativnih koraka.	58
4.6	Zbirni numerički rezultati MSM, HMSM i HSM metoda za broj evaluacija testirane funkcije.	59
4.7	Zbirni numerički rezultati MSM, HMSM i HSM metoda za CPU vreme u sekundama.	60
4.8	Prosečni numerički rezultati za 30 test funkcija testiranih na 11 numeričkih testova.	62
4.9	Zbirni numerički rezultati MSM i SM metoda za broj iteracija, broj evaluacija funkcije i CPU vreme.	63
4.10	Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj iterativnih koraka.	65
4.11	Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj evaluacija funkcija.	66
4.12	Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za CPU vreme u sekundama.	67
4.13	Prosečni numerički rezultati za 24 test funkcija testiranih na 12 numeričkih testova.	67
4.14	Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj iteracija.	69
4.15	Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za broj evaluacija funkcija.	70

4.16	Zbirni numerički rezultati AGD, MAGD, MSM, SM, DMSM i TMSM metoda za CPU vreme u sekundama.	71
4.17	Prosečni numerički rezultati za 24 test funkcija testiranih na 12 nemeričkih testova.	71
5.1	Klasične konjugovano gradijentne metode.	75
5.2	Klasifikacija konjugovano gradijentnih metoda.	75
5.3	Zbirni numerički rezultati za broj iterativnih koraka.	88
5.4	Zbirni numerički rezultati za broj evaluacija testiranih funkcija.	89
5.5	Zbirni numerički rezultati za CPU vreme u sekundama.	90
5.6	Prosečni numerički rezultati za 26 test funkcija testiranih na 10 numeričkih testova.	90
6.1	Zbirni numerički rezultati DHSDL, DLSDL, MMDL i MLSCD metoda za broj iterativnih koraka.	120
6.2	Zbirni numerički rezultati DHSDL, DLSDL, MMDL i MLSCD metoda za broj evaluacija funkcija.	121
6.3	Zbirni numerički rezultati DHSDL, DLSDL, MMDL i MLSCD metoda za CPU vreme u sekundama.	123
6.4	Prosečni numerički rezultati za 33 test funkcija testiranih na 10 numeričkih testova.	124
6.5	Zbirni numerički rezultati EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metoda za broj iterativnih koraka.	125
6.6	Zbirni numerički rezultati EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metoda za broj evaluacija funkcija.	126
6.7	Zbirni numerički rezultati EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metoda za CPU vreme u sekundama.	127
6.8	Zbirni numerički rezultati MSMDL, CG-DESCENT, M1 i DK metoda za broj iterativnih koraka (NI), broj evaluacija testirane funkcije (NFE) i CPU vreme u sekundama (CPUts).	129
6.9	Statistika neuspešno rešenih test funkcija (test probelma) po svakom metodu. . .	131
6.10	Zbirni rezultati analize ponašanja parametara u MSMDL metodi.	132
6.11	Zbirni numerički rezultati EDL, MDL, MSMDL i BB1DL metoda za broj iterativnih koraka (NI), broj evaluacija testirane funkcije (NFE) i CPU vreme u sekundama (CPUts).	134
7.1	Numerički rezultati za DLPM, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUts) i $\ F(\mathbf{x})\ $ za Probleme 1, 2 i 3.	156
7.2	Numerički rezultati za DLPM, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUts) i $\ F(\mathbf{x})\ $ za Probleme 4, 5 i 6.	157

7.3	Numerički rezultati za DLPM, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUts) i $\ F(\mathbf{x})\ $ za Probleme 7, 8 i 9.	158
7.4	Numerički rezultati za DLPM, DKPM, MDLPM i MSMDLPM algoritam za broj iterativnih koraka (NI), broj evaluacija funkcije $F(\mathbf{x})$ (NFE), CPU vreme u sekundama (CPUts) i $\ F(\mathbf{x})\ $ za Probleme 10 i 11.	159
7.5	Analiza rezultata iz Tabela 7.1–7.4 koja prikazuje broj test peimera (procenat) u kojima je posmatrani algoritam postigao minimalnu vrednost za broj iteracija, broj evaluacija funkcije $F(\mathbf{x})$ i CPU vreme.	160
7.6	Performanse MSMDLPM, Algoritma 1 i CGD algoritma pri uklanjanju zamućenja na slikama.	164
8.1	Parametri u metodama opadajućeg gradijenta i neutrosofskim modifikacijama.	171
8.2	Preporučene vrednosti parametara u strukturi neutrosofskog logičkog kontrolera.	174
8.3	Preporučene vrednosti parametara u strukturi neutrosofskog logičkog kontrolera.	178
8.4	Zbirni numerički rezultati <i>MSM</i> , <i>SM</i> , <i>GD</i> , <i>FSM</i> , <i>FGD</i> i <i>FMSM</i> metoda za broj iterativnih koraka.	192
8.5	Zbirni numerički rezultati <i>MSM</i> , <i>SM</i> , <i>GD</i> , <i>FSM</i> , <i>FGD</i> i <i>FMSM</i> metoda za broj evaluacija testirane funkcije.	193
8.6	Zbirni numerički rezultati <i>MSM</i> , <i>SM</i> , <i>GD</i> , <i>FSM</i> , <i>FGD</i> i <i>FMSM</i> metoda za CPU vreme u sekundama.	194
8.7	Prosečni numerički rezultati za 30 test funkcije testiranih na 10 različitih numeričkih testova.	194
8.8	Broj poginulih osoba u saobraćajnim nezgodama u Srbiji od 2012. do 2021. godine.	199
8.9	Numerički rezultati testiranja kvadratnog modela optimizacije pomoću <i>FMSM</i> , <i>FSM</i> i <i>FGD</i> metoda.	201
8.10	Tačka procene i relativna greška za podatak iz 2021. godine.	202
8.11	Zbirni numerički rezultati za DL, FDL i EDL algoritam za broj iterativnih koraka (NI), broj evaluacija testirane funkcije (NFE) i CPU vreme u sekundama (CPUts).	203

Spisak slika

4.1	Profili performansi za MAGD, HMAGD, MSM, HMSM i HSM metod.	57
4.2	Profili performansi bazirani na CPU vremenu za MAGD, HMAGD, MSM, HMSM i HSM metod.	57
4.3	Profili performansi za MSM, HMSM i HSM metod.	61
4.4	Profili performansi bazirani na CPU vremenu za MSM, HMSM i HSM metod.	61
4.5	Profili performansi za MSM i SM metod.	64
4.6	Profili performansi bazirani na CPU vremenu za MSM i SM metod.	64
4.7	Profili performansi za AGD, MAGD, MSM, SM, DMSM i TMSM metod.	68
4.8	Profili performansi bazirani na CPU vremenu za AGD, MAGD, MSM, SM, DMSM i TMSM metod.	68
4.9	Profili performansi za AGD, MAGD, MSM, SM, DMSM i TMSM metod.	72
4.10	Profili performansi bazirani na CPU vremenu za AGD, MAGD, MSM, SM, DMSM i TMSM metod.	72
5.1	Profili performansi za DHSDL, DLSDL i H-BFGS-CG1 metod.	88
5.2	Profili performansi bazirani na CPU vremenu za DHSDL, DLSDL i H-BFGS-CG1 metod.	89
6.1	Grafik funkcije $\mathfrak{I}(\alpha_k) = 1 + \alpha_k - \alpha_k^2$ za $\alpha_k \in (0, 1]$	99
6.2	Profili performansi za DHSDL, DLSDL, MMDL i MLSCD metod.	122
6.3	Profili performansi bazirani na CPU vremenu za DHSDL, DLSDL, MMDL i MLSCD metod.	122
6.4	Profili performansi za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metod.	124
6.5	Profili performansi bazirani na CPU vremenu za EDL, MHSDL3, MHSDL4, MHSDL5 i MHSDL6 metod.	128
6.6	Profili performansi za MSMDL, M1, CG-DESCENT i DK metod.	130
6.7	Profili performansi bazirani na CPU vremenu za MSMDL, M1, CG-DESCENT i DK metod.	130
6.8	Profili performansi za EDL, MDL, MSMDL i BB1DL metod.	133
6.9	Profili performansi bazirani na CPU vremenu za EDL, MDL, MSMDL i BB1DL metod.	133
6.10	Numerički rezultati generisani MSMDL metodom za $\hat{\varphi}_k = \hat{\varphi}_{1k}$	136

6.11	Numerički rezultati generisani MSMDL metodom za $\hat{\varphi}_k = \hat{\varphi}_{2k}$	137
7.1	Profili performansi za DLPM, DKPM, MDLPM i MSMDLPM metod.	161
7.2	Profili performansi bazirani na CPU vremenu za DLPM, DKPM, MDLPM i MSMDLPM metod.	162
7.3	Originalne slike (prva kolona), zamućene slike (druga kolona), restaurirane slike MSMDLPM algoritmom (treća kolona), Algoritmom 1 (četvrta kolona) i CGD algoritmom (poslednja kolona).	164
7.4	Originalne slike (prva kolona), zamućene slike (druga kolona), restaurirane slike MSMDLPM algoritmom (treća kolona), Algoritmom 1 (četvrta kolona) i CGD algoritmom (poslednja kolona).	165
8.1	Opšta struktura fazi metoda optimizacije.	172
8.2	Struktura neutrosofskog logičkog kontrolera bazirana na neutrosofskim pravilima.	174
8.3	Neutrosofikacija i deneutrosofikacija na osnovu parametara u Tabeli 8.2. . . .	175
8.4	Opšta struktura fazi neutrosofskog Dai–Liao konjugovano gradijentnog metoda. . .	177
8.5	Neutrosofikacija i deneutrosofikacija na osnovu parametara u Tabeli 8.3. . .	179
8.6	Profili performansi za <i>MSM</i> , <i>SM</i> , <i>GD</i> , <i>FSM</i> , <i>FGD</i> i <i>FMSM</i> metod.	195
8.7	Profili performansi bazirani na CPU vremenu za <i>MSM</i> , <i>SM</i> , <i>GD</i> , <i>FSM</i> , <i>FGD</i> i <i>FMSM</i> metod.	195
8.8	Rang broja iteracija za <i>GD</i> , <i>SM</i> , <i>MSM</i> , <i>FGD</i> , <i>FSM</i> i <i>FMSM</i> metod na 30 test funkcija i njihov prosečan rang broja iteracija.	196
8.9	Rang broja evaluacija funkcija za <i>GD</i> , <i>SM</i> , <i>MSM</i> , <i>FGD</i> , <i>FSM</i> i <i>FMSM</i> metod na 30 test funkcija i njihov prosečan rang broja evaluacija funkcija. . .	197
8.10	Rang potrošnje CPU vremena za <i>GD</i> , <i>SM</i> , <i>MSM</i> , <i>FGD</i> , <i>FSM</i> i <i>FMSM</i> metod na 30 test funkcija i njihov prosečan rang potrošnje CPU vremena. . .	198
8.11	Nelinearna kvadratna linija trenda poginulih osoba u saobraćajnim nezgodama u Srbiji.	201
8.12	Profili performansi za DL, FDL i EDL algoritme.	204
8.13	Profili performansi bazirani na CPU vremenu za DL, FDL i EDL algoritme. . .	204

Literatura

- [1] A. B. Abubakar and P. Kumam, *A descent Dai–Liao conjugate gradient method for nonlinear equations*, Numerical Algorithms, 81, 197–210, 2019.
- [2] A. B. Abubakar, P. Kumam and H. Mohammad, *A note on the spectral gradient projection method for nonlinear monotone equations with applications*, Computational and Applied Mathematics, 39, Article number: 129, 2020.
- [3] A. B. Abubakar, K. Muangchoo, A. H. Ibrahim, A. B. Muhammad, L. O. Jolaoso and K. O. Aremu, *A new three-term Hestenes–Stiefel type method for nonlinear monotone operator equations and image restoration*, IEEE Access, 9, 18262–18277, 2021.
- [4] M. Ahookhosh, K. Amini and S. Bahrami, *Two derivative-free projection approaches for systems of large-scale nonlinear monotone equations*, Numer. Algor., 64(1), 21–42, 2013.
- [5] Z. Aminifard and S. Babaie-Kafaki, *An optimal parameter choice for the Dai–Liao family of conjugate gradient methods by avoiding a direction of the maximum magnification by the search direction matrix*, 4OR, 17, 317–330, 2019.
- [6] N. Andrei, *An acceleration of gradient descent algorithm with backtracking for unconstrained optimization*, Numer. Algorithms, 42, 63–73, 2006.
- [7] N. Andrei, *An unconstrained optimization test functions collection*, Adv. Model. Optim., 10, 147–161, 2008.
- [8] N. Andrei, *Open problems in nonlinear conjugate gradient algorithms for unconstrained optimization*, Bull. Malays. Math. Sci. Soc., 34(2), 319–330, 2011.
- [9] N. Andrei, *A Dai–Liao conjugate gradient algorithm with clustering of eigenvalues*, Numer. Algorithms, 77, 1273–1282, 2018.
- [10] N. Andrei, *Relaxed gradient descent and a new gradient descent methods for unconstrained optimization*, <https://camo.ici.ro/neculai/newgrad.pdf>, Visited September 27, 2024.

- [11] A. Q. Ansari, *From fuzzy logic to neutrosophic logic: A paradigm shift and logics*, In Proceedings of the 2017 International Conference on Intelligent Communication and Computational Techniques (ICCT), Jaipur, India, 22–23 December 2017, pp. 11–15.
- [12] L. Armijo, *Minimization of functions having Lipschitz continuous first partial derivatives*, Pacific J. Math., 16, 1–3, 1966.
- [13] K. T. Atanassov, *Intuitionistic fuzzy sets*, Fuzzy Sets Syst., 20, 87–96, 1986.
- [14] A. M. Awwal, P. Kumam, L. Wang, S. Huang and W. Kumam, *Inertial-based derivative-free method for system of monotone nonlinear equations and application*, IEEE Access, 8, 226921–226930, 2020.
- [15] A. M. Awwal, I. M. Sulaiman, M. Malik, M. Mamat, P. Kumam and K. Sitthithakerngkiet, *A spectral RMIL+ conjugate gradient method for unconstrained optimization with applications in portfolio selection and motion control*, IEEE Access, 9, 75398–75414, 2021.
- [16] A. M. Awwal, L. Wang, P. Kumam and H. Mohammad, *A two-step spectral gradient projection method for system of nonlinear monotone equations and image deblurring problems*, Symmetry, 12(6), Article number: 874, 2020.
- [17] S. Babaie-Kafaki, *A survey on the Dai–Liao family of nonlinear conjugate gradient methods*, RAIRO-Oper. Res., 57, 43–58, 2023.
- [18] S. Babaie-Kafaki and R. Gambari, *A descent family of Dai-Liao conjugate gradient methods*, Optim. Meth. Soft., 29(3), 583–591, 2014.
- [19] S. Babaie-Kafaki and R. Ghanbari, *The Dai-Liao nonlinear conjugate gradient method with optimal parameter choices*, European J. Oper. Res., 234, 625–630, 2014.
- [20] B. Baluch, Z. Salleh, A. Alhawarat and U. A. M. Roslan, *A new modified three-term conjugate gradient method with sufficient descent property and its global convergence*, J. Math., 2017, Article ID 2715854, 12 pages, 2017.
- [21] M. R. Banham and A. K. Katsaggelos, *Digital image restoration*, IEEE Signal Processing Magazine, 14(2), 24–41, 1997.
- [22] J. Barzilai and J. M. Borwein, *Two-point step-size gradient method*, IMA J. Numer. Anal., 8, 141–148, 1988.
- [23] M. Bastani and D. K. Salkuyeh, *On the GSOR iteration method for image restoration*, Numerical Algebra, Control and Optimization, 2020.
- [24] Y. Bing and G. Lin, *An efficient implementation of merrill’s method for sparse or partially separable systems of nonlinear equations*, SIAM J. Optim., 1(2), 206–221, 1991.

-
- [25] A. E. J. Bogaers, S. Kok, B. D. Reddy and T. Franz, *An evaluation of quasi-Newton methods for application to FSI problems involving free surface flow and solid body contact*, Computers & Structures, 173, 71–83, 2016.
- [26] I. Bongartz, A.R. Conn, N.I.M. Gould and Ph.L. Toint, *CUTer: constrained and unconstrained testing environments*, ACM Trans. Math. Softw., 21, 123–160, 1995.
- [27] A. C. Bovik, *Handbook of Image and Video Processing*, New York, NY, USA: Academic, 2010.
- [28] C. Brezinski, *A classification of quasi-Newton methods*, Numer. Algorithms, 33, 123–135, 2003.
- [29] R.H. Byrd and J. Nocedal, *A tool for the analysis of quasi-Newton methods with application to unconstrained minimization*, SIAM J. Numer. Anal., 26(3), 727–739, 1989.
- [30] J. Cao and J. Wu, *A conjugate gradient algorithm and its applications in image restoration*, Appl. Numer. Math., 152, 243–252, 2020.
- [31] C. L. Chan, A. K. Katsaggelos and A. V. Sahakian, *Image sequence filtering in quantum-limited noise with applications to low-dose fluoroscopy*, IEEE Transactions on Medical Imaging, 12(3), 610–621, 1993.
- [32] W. Cheng, *A two-term PRP-based descent method*, Numer. Funct. Anal. Optim., 28(11–12), 1217–1230, 2007.
- [33] Y. Cheng, Q. Mou, X. Pan and S. Yao, *A sufficient descent conjugate gradient method and its global convergence*, Optim. Methods Softw., 31, 577–590, 2016.
- [34] V. Christianto and F. Smarandache, *A Review of Seven Applications of Neutrosophic Logic: In Cultural Psychology, Economics Theorizing, Conflict Resolution, Philosophy of Science, etc.*, Multidisciplinary Scientific Journal, 2, 128–137, 2019.
- [35] A. I. Cohen, *Stepsize analysis for descent methods*, J. Optim. Theory Appl., 33, 187–205, 1981.
- [36] A. R. Conn, N. I. M. Gould and Ph. L. Toint, *Convergence of quasi-Newton matrices generated by the symmetric rank one update*, Math. Programming, 50, 177–195, 1991.
- [37] W. L. Cruz, J. M. Martinez and M. Raydan, *Spectral residual method without gradient information for solving large-scale nonlinear systems of equations: theory and experiments*, Math. Comp., 75(225), 1429–1448, 2006.
- [38] Y.-H. Dai, *Alternate Step Gradient Method*, Report AMSS–2001–041, Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, 2001.

- [39] Y. Dai, *A nonmonotone conjugate gradient algorithm for unconstrained optimization*, J. Syst. Sci. Complex., 15, 139–145, 2002.
- [40] Y.-H. Dai, *Nonlinear Conjugate Gradient Methods*, Wiley Encyclopedia of Operations Research and Management Science, 2011.
- [41] Y. -H. Dai and C. -X. Kou, *A nonlinear conjugate gradient algorithm with an optimal property and an improved wolfe line search*, SIAM. J. Optim., 23(1), 296—320, 2013.
- [42] Y.-H. Dai and L.-Z. Liao, *New conjugacy conditions and related nonlinear conjugate gradient methods*, Appl. Math. Optim., 43, 87–101, 2001.
- [43] Y.-H. Dai and Q. Ni, *Testing different conjugate gradient methods for large-scale unconstrained optimization*, J. Comput. Math., 21, 311–320, 2003.
- [44] Z. Dai and F. Wen, *Another improved Wei–Yao–Liu nonlinear conjugate gradient method with sufficient descent property*, Appl. Math. Comput., 218, 7421–7430, 2012.
- [45] Y. H. Dai and Y. Yuan, *A class of Globally Convergent Conjugate Gradient Methods*, Research report ICM-98-030, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, 1998.
- [46] Y.-H. Dai and Y. Yuan, *A nonlinear conjugate gradient method with a strong global convergence property*, SIAM J. Optim., 10, 177–182, 1999.
- [47] Y.-H. Dai and Y. Yuan, *An efficient hybrid conjugate gradient method for unconstrained optimization*, Ann. Oper. Res., 103, 33–47, 2001.
- [48] Y. Dai and Y. Yuan, *A class of globally convergent conjugate gradient methods*, Sci. China Ser. A, 46, 251–261, 2003.
- [49] Y.-H. Dai and Y.-X. Yuan, *Alternate minimization gradient method*, IMA J. Numer. Anal., 23, 377–393, 2003.
- [50] Y.-H. Dai and Y.-X. Yuan, *Analysis of monotone gradient methods*, J. Ind. Manag. Optim., 1, 181–192, 2005.
- [51] Y. Dai, J. Yuan and Y.-X. Yuan, *Modified two-point step-size gradient methods for unconstrained optimization*, Comput. Optim. Appl., 22, 103–109, 2002.
- [52] Y.-H. Dai and H. Zhang, *An Adaptive Two-Point Step-size gradient method*, Research report, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, 2001.
- [53] J. W. Daniel, *The conjugate gradient method for linear and nonlinear operator equations*, SIAM J. Numer. Anal., 4, 10–26, 1967.

-
- [54] K. U. Danmalam, H. Mohammad and M. Y. Waziri, *Structured diagonal Gauss–Newton method for nonlinear least squares*, Comp. Appl. Math., 41, Article number: 68, 2022.
- [55] W. C. Davidon, *Variable metric methods for minimization*, Argonne National Labs Report, ANL–5990, 1959.
- [56] M. Dawahdeh, M. Mamat, M. Rivaie, I. M. Sulaiman, *Application of conjugate gradient method for solution of regression models*, Int. J. Adv. Sci. Technol., 29, 1754–1763, 2020.
- [57] S. Delladji, M. Belloufi and B. Sellami, *Behavior of the combination of PRP and HZ methods for unconstrained optimization*, Numerical Algebra, Control and Optimization, 11(3), 377–389, 2021.
- [58] E.D. Dolan and J.J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., 91(2), 201–213, 2002.
- [59] S. S. Djordjević, *Two modifications of the method of the multiplicative parameters in descent gradient methods*, Appl. Math, Comput., 218, 8672–8683, 2012.
- [60] S. S. Djordjević, *Unconstrained Optimization Methods: Conjugate Gradient Methods and Trust-Region Methods*, Book Chapter, 2019.
- [61] M. S. Engelman, G. Strang and K.-J. Bathe, *The application of quasi-Newton methods in fluid mechanics*, Internat. J. Numer. Methods Engrg., 17, 707–718, 1981.
- [62] D. K. Faddeev and I. S. Sominskiĭ, *Collection of Problems on Higher Algebra*. Gostekhizdat, 2nd edition, Moscow, 1949.
- [63] A. G. Farizawani, M. Puteh, Y. Marina and A. Rivaie, *A review of artificial neural network learning rule based on multiple variant of conjugate gradient approaches*, Journal of Physics: Conference Series, 1529, 2020.
- [64] R. Fletcher, *Practical Methods of Optimization, Unconstrained Optimization*, 1st edition, Wiley, New York, 1987.
- [65] R. Fletcher and M. J. D. Powell, *A rapid convergent descent method for minimization*, Computer Journal, 6, 163–168, 1963.
- [66] R. Fletcher and C. M. Reeves, *Function minimization by conjugate gradients*, Comput. J., 7, 149–154, 1964.
- [67] J. C. Gilbert and J. Nocedal, *Global convergence properties of conjugate gradient methods for optimization*, SIAM J. Optim., 2, 21–42, 1992.
- [68] A. A. Goldstein, *On steepest descent*, J. SIAM Control Ser. A, 3, 147–151, 1965.

- [69] L. Grippo, F. Lampariello and S. Lucidi, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. ANAL., 23, 707–716, 1986.
- [70] L. Grippo, F. Lampariello and S. Lucidi, *A truncated Newton method with nonmonotone line search for unconstrained optimization*, J. Optim. Theory Appl., 60, 401–419, 1989.
- [71] L. Grippo, F. Lampariello and S. Lucidi, *A class of nonmonotone stability methods in unconstrained optimization*, Numer. Math., 59, 779–805, 1991.
- [72] Y. Guo, H. D. Cheng and Y. Zhang, *A new neutrosophic approach to image denoising*, New Math. Nat. Comput., 5, 653–662, 2009.
- [73] W. W. Hager and H. Zhang, *A new conjugate gradient method with guaranteed descent and an efficient line search*, SIAM J. Optim., 16, 170–192, 2005.
- [74] W. W. Hager and H. Zhang, *A survey of nonlinear conjugate gradient methods*, Pac. J. Optim., 2, 35–58, 2006.
- [75] W. W. Hager and H. Zhang, *Algorithm 851: CG_DESCENT, a conjugate gradient method with guaranteed descent*, ACM Trans. Math. Software, 32, 113–137, 2006.
- [76] L. Han and M. Neumann, *Combining quasi-Newton and Cauchy directions*, Int. J. Appl. Math., 12, 167–191, 2003.
- [77] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49, 409–436, 1952.
- [78] Y. F. Hu and C. Storey, *Global convergence result for conjugate gradient methods*, J. Optim. Theory Appl., 71, 399–405, 1991.
- [79] A. H. Ibrahim, P. Kumam, A. B. Abubakar, W. Jirakitpuwapat and J. Abubakar, *A hybrid conjugate gradient algorithm for constrained monotone equations with application in compressive sensing*, Heliyon, 6(3), e03466, 2020.
- [80] A. H. Ibrahim, P. Kumam and W. Kumam, *A family of derivative-free conjugate gradient methods for constrained nonlinear equations and image restoration*, IEEE Access, 8, 162714–162729, 2020.
- [81] M.A.H. Ibrahim, M. Mamat and W.J. Leong, *The hybrid BFGS-CG method in solving unconstrained optimization problems*, Hindawi Publishing Corporation, Abstract and Applied Analysis, 2014, Article ID 507102, 6 pages, 2014.
- [82] S. Ishikawa, *Fixed points by a new iteration method*, Proc. Am. Math. Soc., 44, 147–150, 1974.

-
- [83] B. Ivanov, G. V. Milovanović, P. S. Stanimirović, A. M. Awwal, L. A. Kazakovtsev and V. N. Krutikov, *A Modified Dai–Liao Conjugate Gradient Method Based on a Scalar Matrix Approximation of Hessian and Its Application*, Journal of Mathematics, vol. 2023, Article ID 9945581, 20 pages, 2023.
 - [84] B. Ivanov, G. V. Milovanović and P. S. Stanimirović, *Accelerated Dai–Liao projection method for solving systems of monotone nonlinear equations with application to image deblurring*, Journal of Global Optimization, 85, 377–420, 2023.
 - [85] B. Ivanov, B. I. Shaini and P. S. Stanimirović, *Multiple use of backtracking line search in unconstrained optimization*, Facta Universitatis (Niš): Ser. Math. Inform., 35(5), 1417–1438, 2020.
 - [86] B. Ivanov, P. S. Stanimirović, G. V. Milovanović, S. Djordjević and I. Brajević, *Accelerated multiple step-size methods for solving unconstrained optimization problems*, Optimization Methods and Software, 36, 998–1029, 2021.
 - [87] B. Ivanov, P. S. Stanimirović, B. I. Shaini, H. Ahmad and M.-K. Wang, *A Novel Value for the Parameter in the Dai–Liao-Type Conjugate Gradient Method*, Journal of Function Spaces, vol. 2021, Article ID 6693401, 10 pages, 2021.
 - [88] Z. Jia, *Applications of the conjugate gradient method in optimal surface parameterizations*, Int. J. Comput. Math., 87, 1032–1039, 2010.
 - [89] A. M. Khalil, D. Cao, A. Azzam, F. Smarandache and W. R. Alharbi, *Combination of the single-valued neutrosophic fuzzy set and the soft set with applications in decision-making*, Symmetry, 12, 1361, 2020.
 - [90] S. H. Khan, *A Picard-Mann hybrid iterative process*, Fixed Point Theory Appl., 2013, Article number: 69, 10 pp., 2013.
 - [91] Z. Khanaiah and G. Hmod, *Novel hybrid algorithm in solving unconstrained optimizations problems*, International Journal of Novel Research in Physics Chemistry & Mathematics, 4, 36–42, 2017.
 - [92] N. Kontrec and M. Petrović, *Implementation of gradient methods for optimization of underage costs in aviation industry*, University Thought, Publication in Natural Sciences, 6, 71–74, 2016.
 - [93] M. S. Koorapetse and P. Kaelo, *Globally convergent three-term conjugate gradient projection methods for solving nonlinear monotone equations*, Arab. J. Math. (Springer), 7, 289–301, 2018.
 - [94] M. Koorapetse and P. Kaelo, *A new three-term conjugate gradient-based projection method for solving large-scale nonlinear monotone equations*, Math. Model. Anal., 24(4), 550–563, 2019.

- [95] M. Koorapetse and P. Kaelo, *Self adaptive spectral conjugate gradient method for solving nonlinear monotone equations*, J. Egyptian Math. Soc., 28(1), Paper No. 4, 21 pp, 2020.
- [96] M. Li, *A family of three-term nonlinear conjugate gradient methods close to the memoryless BFGS method*, Optim. Lett., 12, 1911–1927, 2018.
- [97] Q. Li and D. H. Li, *A class of derivative-free methods for large-scale nonlinear monotone equations*, IMA Journal of Numerical Analysis, 31, 1625–1635, 2011.
- [98] X. Li and Q. Ruan, *A modified PRP conjugate gradient algorithm with trust region for optimization problems*, Numer. Funct. Anal. Optim., 32(5), 496–506, 2011.
- [99] X. Li, C. Shen and L.-H. Zhang, *A projected preconditioned conjugate gradient method for the linear response eigenvalue problem*, Numer. Algebra Control Optim., 8, 389–412, 2018.
- [100] D.-H. Li and X.-L. Wang, *A modified Fletcher-Reeves-type derivative-free method for symmetric nonlinear equations*, Numer. Algebra Control Optim., 1, 71–82, 2011.
- [101] K. H. Lim, H. H. Tan and H. G. Harno, *Approximate greatest descent in neural network optimization*, Numer. Algebra Control Optim., 8, 327–336, 2018.
- [102] J. K. Liu and S. J. Li, *A projection method for convex constrained monotone nonlinear equations with applications*, Comput. Math. Appl., 70, 2442–2453, 2015.
- [103] J. K. Liu and S. J. Li, *A three-term derivative-free projection method for nonlinear monotone system of equations*, Calcolo, 53, 427–450, 2016.
- [104] J. Liu and S. Li, *Multivariate spectral DY-type projection method for convex constrained nonlinear monotone equations*, Journal of Industrial & Management Optimization, 13(1), 283–295, 2017.
- [105] Y. Liu and C. Storey, *Efficient generalized conjugate gradient algorithms, part 1: Theory*, J. Optim. Theory Appl., 69, 129–137, 1991.
- [106] I. E. Livieris and P. Pintelas, *A descent Dai-Liao conjugate gradient method based on a modified secant equation and its global convergence*, ISRN Computational Mathematics, 2012, Article ID 435495, 2012.
- [107] M. Lotfi and S. M. Hosseini, *An efficient Dai-Liao type conjugate gradient method by reformulating the CG parameter in the search direction equation*, J. Comput. Appl. Math., 371, 112708, 2020.
- [108] Y.-Z. Luo, G.-J. Tang and L.-N. Zhou, *Hybrid approach for solving systems of nonlinear equations using chaos optimization and quasi-Newton method*, Applied Soft Computing, 8, 1068–1073, 2008.

-
- [109] T. Mahmood, K. Ullah, Q. Khan, N. Jan, *An approach toward decision-making and medical diagnosis problems using the concept of spherical fuzzy sets*, Neural Comput. Appl., 31, 7041–7053, 2019.
 - [110] W. R. Mann, *Mean value methods in iterations*, Proc. Amer. Math. Soc., 4, 506–510, 1953.
 - [111] A. Miele, T. Wang and S. Mancuso, *Optimization of missions to Mars for robotic and manned spacecraft*, Nonlinear Analysis: Theory, Methods and Applications, 47(3), 1425–1443, 2001.
 - [112] M. Miladinović, P. S. Stanimirović, *Nelinearna optimizacija*, Prirodno–matematički fakultet, Niš, 2015.
 - [113] M. Miladinović, P. Stanimirović and S. Miljković, *Scalar correction method for solving large scale unconstrained minimization problems*, J. Optim. Theory Appl., 151, 304–320, 2011.
 - [114] S. K. Mishra and B. Ram, *Introduction to Unconstrained Optimization with R*, 1st edition, Springer Singapore, Springer Nature Singapore Pte Ltd., 2019.
 - [115] K. Mishra, I. Kandasamy, W. B. V. Kandasamy and F. Smarandache, *A novel framework using neutrosophy for integrated speech and text sentiment analysis*, Symmetry, 12, 1715, 2020.
 - [116] I. S. Mohammed, M. Mamat, I. Abdulkarim and F. S. Bt. Mohamad, *A survey on recent modifications of conjugate gradient methods*, Proceedings of the UniSZA Research Conference 2015 (URC 5), Universiti Sultan Zainal Abidin, 14–16 April 2015.
 - [117] H. Mohammad, M. Y. Waziri and S. A. Santos, *A brief survey of methods for solving nonlinear least-squares problems*, Numer. Algebra Control Optim., 9, 1–13, 2019.
 - [118] A. U. Moyi, W. J. Leong, I. Saidu, *On the application of three-term conjugate gradient method in regression analysis*, Int. J. Comput. Appl., 102, 1–4, 2014.
 - [119] Y. Narushima and H. Yabe, *A survey of sufficient descent conjugate gradient methods for unconstrained optimization*, SUT J. Math., 50, 167–203, 2014.
 - [120] J. L. Nazareth, *Conjugate-Gradient Methods*, In: C. Floudas and P. Pardalos (eds), Encyclopedia of Optimization, 2nd edition, Springer, Boston, 2009.
 - [121] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, 1999.
 - [122] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equation in Several Variables*, Academic Press, New York, London, 1970.

- [123] W. F. H. W. Osman, M. A. H. Ibrahim and M. Mamat, *Hybrid DFP-CG method for solving unconstrained optimization problems*, Journal of Physics: Conf. Series, 890, 012033, (2017).
- [124] S. Panić, M. J. Petrović and M. Mihajlov-Carević, *Initial improvement of the hybrid accelerated gradient descent process*, Bull. Aust. Math. Soc., 98, 331–338, 2018.
- [125] A. Perry, *A modified conjugate gradient algorithm*, Oper. Res., 26, 1073–1078, 1978.
- [126] M. J. Petrović, *An accelerated double step-size method in unconstrained optimization*, Applied Math. Comput., 250, 309–319, 2015.
- [127] M. J. Petrović, *Hybridization rule applied on accelerated double step size optimization scheme*, Filomat, 33, 655–665, 2019.
- [128] M. J. Petrović, N. Kontrec and S. Panić, *Determination of accelerated factors in gradient descent iterations based on Taylor's series*, University Thought, Publication in Natural Sciences, 7, 41–45, 2017.
- [129] M. J. Petrović, V. Rakočević, N. Kontrec, S. Panić and D. Ilić, *Hybridization of accelerated gradient descent method*, Numer. Algorithms, 79, 769–786, 2018.
- [130] M. J. Petrović, V. Rakočević, D. Valjarević and D. Ilić, *A note on hybridization process applied on transformed double step size model*, Numerical Algorithms, 85, 449–465, 2020.
- [131] M. J. Petrović and P. S. Stanimirović, *Accelerated double direction method for solving unconstrained optimization problems*, Math. Probl. Eng., 2014, Article ID 965104, 8 pages, 2014.
- [132] M. J. Petrović, P. S. Stanimirović, N. Kontrec and J. Mladenović, *Hybrid modification of accelerated double direction method*, Math. Probl. Eng., 2018, Article ID 1523267, 8 pages, 2018.
- [133] M. R. Peyghami, H. Ahmadzadeh and A. Fazli, *A new class of efficient and globally convergent conjugate gradient methods in the Dai-Liao family*, Optim. Methods Softw., 30, 843–863, 2015.
- [134] E. Picard, *Memoire sur la theorie des equations aux derivees partielles et la methode des approximations successives*, J. Math. Pures Appl., 6, 145–210, 1890.
- [135] E. Polak and G. Ribière, *Note sur la convergence des méthodes de directions conjuguées*, Rev. Française Informat Recherche Opérationnelle, 3, 35–43, 1969.
- [136] B. T. Polyak, *The conjugate gradient method in extreme problems*, USSR Comput. Math. and Math. Phys., 9, 94–112, 1969.

-
- [137] M. J. D. Powell, *Some global convergence properties of a variable-metric algorithm for minimization without exact line searches*, SIAM–AMS Proc., Philadelphia, 9, 53–72, 1976.
 - [138] Y. Qiang, F. Jing, J. Zeng and Z. Hou, *Dynamic modeling and vibration mode analysis for an industrial robot with rigid links and flexible joints*, in Proceedings of the 2012 24th Chinese Control and Decision Conference (CCDC), 3317–3321, IEEE, Taiyuan, China, May 2012.
 - [139] V. Rakočević and M.J. Petrović, *Comparative analysis of accelerated models for solving unconstrained optimization problems with application of Khan’s hybrid rule*, Mathematics, 10(23), 4411, 2022.
 - [140] M. Raydan, *On the Barzilai and Borwein choice of steplength for the gradient method*, IMA J. Numer. Anal., 13, 321–326, 1993.
 - [141] M. Raydan, *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*, SIAM J. Optim., 7, 26–33, 1997.
 - [142] M. Raydan and B. F. Svaiter, *Relaxed steepest descent and Cauchy-Barzilai-Borwein method*, Comput. Optim. Appl., 21, 155–167, 2002.
 - [143] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, 1970.
 - [144] J. Sabi’u, A. Shah, P. S. Stanimirović, B. Ivanov and M. Y. Waziri, *Modified optimal Perry conjugate gradient method for solving system of monotone equations with applications*, Applied Numerical Mathematics, 184, 431–445, 2023.
 - [145] N. Shapiee, M. Rivaie, M. Mamat and P. L. Ghazali, *A new family of conjugate gradient coefficient with application*, International Journal of Engineering & Technology, 7, 36–43, 2018.
 - [146] Z.-J. Shi, *Convergence of line search methods for unconstrained optimization*, Appl. Math. Comput., 157, 393–405, 2004.
 - [147] S. Shoid, N. Shapiee, N. Zull, N. H. A. Ghani, N. S. Mohamed, M. Rivaie and M. Mamat, *The application of new conjugate gradient methods in estimating data*, International Journal of Engineering & Technology, 7, 25–27, 2018.
 - [148] F. Smarandache, *A Unifying Field in Logics, Neutrosophy: Neutrosophic Probability, Set and Logic*, American Research Press: Rehoboth, NM, USA, 1999.
 - [149] F. Smarandache, *Neutrosophic Logic—A Generalization of the Intuitionistic Fuzzy Logic*, 25 January 2016. Available online: <https://ssrn.com/abstract=2721587> (accessed on 1 September 2021).

- [150] M. V. Solodov and B. F. Svaiter, *A globally convergent inexact Newton method for systems of monotone equations*, Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, Applied Optimization, 22, 355–369, 1998.
- [151] P. S. Stanimirović, B. Ivanov, S. Djordjević and I. Brajević, *New hybrid conjugate gradient and Broyden-Fletcher-Goldfarb-Shanno conjugate gradient methods*, J. Optim. Theory Appl., 178, 860–884, 2018.
- [152] P. S. Stanimirović, B. Ivanov, H. Ma and D. Mosić, *A survey of gradient methods for solving nonlinear optimization*, Electronic Research Archive, 28(4), 1573–1624, 2020.
- [153] P. S. Stanimirović, B. Ivanov, D. Stanujkić, V. N. Katsikis, S. D. Mourtas, L. A. Kazakovtsev, S. A. Edalatpanah, *Improvement of Unconstrained Optimization Methods Based on Symmetry Involved in Neutrosophy*, Symmetry, 15(1), 250, 2023.
- [154] P. S. Stanimirović, B. D. Ivanov, D. Stanujkić, L. A. Kazakovtsev, V. N. Krutikov, D. Karabašević, *Fuzzy Adaptive Parameter in the Dai-Liao Optimization Method Based on Neutrosophy*, Symmetry, 15(6), 1217, 2023.
- [155] P. S. Stanimirović and M. B. Miladinović, *Accelerated gradient descent methods with line search*, Numer. Algor., 54, 503–520, 2010.
- [156] P. S. Stanimirović, G. V. Milovanović, M. J. Petrović and N. Z. Kontrec, *A transformation of accelerated double step-size method for unconstrained optimization*, Math. Probl. Eng., 2015, Article ID 283679, 8 pages, 2015.
- [157] I. M. Sulaiman, N. A. Bakar, M. Mamat, B. A. Hassan, M. Malik, A. M. Ahmed, *A new hybrid conjugate gradient algorithm for optimization models and its application to regression analysis*, Indones. J. Electr. Eng. Comput. Sci., 23, 1100–1109, 2021.
- [158] I. M. Sulaiman, M. Malik, A. M. Awwal, P. Kumam, M. Mamat, S. Al-Ahmad, *On three-term conjugate gradient method for optimization problems with applications on COVID-19 model and robotic motion control*, Adv. Contin. Discret. Model., 2022, 1, 2022.
- [159] I. M. Sulaiman, M. Mamat, *A new conjugate gradient method with descent properties and its application to regression analysis*, J. Numer. Anal. Ind. Appl. Math., 14, 25–39, 2020.
- [160] W. Sun, J. Han and J. Sun, *Global convergence of nonmonotone descent methods for unconstrained optimization problems*, J. Comp. Appl. Math., 146, 89–98, 2002.
- [161] M. Sun, J. Liu and Y. Wang, *Two improved conjugate gradient methods with application in compressive sensing and motion control*, Mathematical Problems in Engineering, vol. 2020, Article ID 9175496, 11 pages, 2020.

-
- [162] W. Sun and Y.-X. Yuan, *Optimization Theory and Methods: Nonlinear Programming*, 1st edition, Springer, New York, 2006.
- [163] G. Y. Tang, L. Sun, C. Li and M.Q. Fan, *Successive approximation procedure of optimal tracking control for nonlinear similar composite systems*, Nonlinear Analysis: Theory, Methods and Applications, 70(2), 631–641, 2009.
- [164] Ph. L. Toint, *Non-monotone trust-region algorithm for nonlinear optimization subject to convex constraints*, Math. Prog., 77, 69–94, 1997.
- [165] D. Touati-Ahmed and C. Storey, *Efficient hybrid conjugate gradient techniques*, J. Optim. Theory Appl., 64, 379–397, 1990.
- [166] K. Ullah, *Picture fuzzy maclaurin symmetric mean operators and their applications in solving multiattribute decision-making problems*, Math. Probl. Eng., 2021, 1098631, 2021.
- [167] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, *Image quality assessment: From error visibility to structural similarity*, IEEE Transactions on Image Processing, 13(4), 600–612, 2004.
- [168] H. Wang, F. Smarandache, Y. Q. Zhang and R. Sunderraman, *Single valued neutrosophic sets*, Multispace Multistructure, 4, 410–413, 2010.
- [169] Z. Wei, G. Li and L. Qi, *New nonlinear conjugate gradient formulas for large-scale unconstrained optimization problems*, Appl. Math. Comput., 179, 407–430, 2006.
- [170] Z. Wei, S. Yao and L. Liu, *The convergence properties of some new conjugate gradient methods*, Appl. Math. Comput., 183, 1341–1350, 2006.
- [171] P. Wolfe, *Convergence conditions for ascent methods*, SIAM Rev., 11, 226–235, 1969.
- [172] P. Wolfe, *Convergence conditions for ascent methods. II: Some corrections*, SIAM Rev., 13, 185–188, 1971.
- [173] Y. H. Xiao, Q. Y. Wang and Q. J. Hu, *Non-smooth equations based method for ℓ_1 -norm problems with applications to compressed sensing*, Nonlinear Anal., 74(11), 3570–3577, 2011.
- [174] Y. Xiao and H. Zhu, *A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing*, J. Math. Anal. Appl., 405(1), 310–319, 2013.
- [175] H. Yabe and M. Takano, *Global convergence properties of nonlinear conjugate gradient methods with modified secant condition*, Comput. Optim. Appl., 28, 203–225, 2004.

- [176] M. M. Yahaya, P. Kumam, A. M. Awwal and S. Aji, *A structured quasi-Newton algorithm with nonmonotone search strategy for structured NLS problems and its application in robotic motion control*, J. Comput. Appl. Math., 395(5), Article ID 113582, 2021.
- [177] X. Yang, Z. Luo and X. Dai, *A global convergence of LS-CD hybrid conjugate gradient method*, Adv. Numer. Anal., 2013, Article ID 517452, 5 pages, 2013.
- [178] S. Yao, X. Lu and Z. Wei, *A conjugate gradient method with global convergence for large-scale unconstrained optimization problems*, J. Appl. Math., 2013, Article ID 730454, 9 pages, 2013.
- [179] S. Yao and B. Qin, *A hybrid of DL and WYL nonlinear conjugate gradient methods*, Abstr. Appl. Anal., 2014, Article ID 279891, 2014.
- [180] S. Yao, Z. Wei and H. Huang, *A notes about WYL's conjugate gradient method and its applications*, Appl. Math. Comput., 191, 381–388, 2007.
- [181] L. E. Yeganova, J. E. Falk and Y. V. Dandurova, *Robust separation of multiple sets*, Nonlinear Analysis: Teory, Methods and Applications, 47(3), 1845–1856, 2001.
- [182] G. Yu, S. Niu, J. Ma and Y. Song, *An adaptive prediction–correction method for solving large–scale nonlinear systems of monotone equations with applications*, Abstract and Applied Analysis ,2013, Article ID 619123, 2013.
- [183] Y. Yuan, *A new stepsize for the steepest descent method*, J. Comput. Math., 24, 149–156, 2006.
- [184] G. Yuan, T. Li and W. Hu, *A conjugate gradient algorithm and its application in large-scale optimization problems and image restoration*, J. Inequal. Appl., 2019, Article number: 247, 25 pp, 2019.
- [185] G. Yuan, T. Li and W. Hu, *A conjugate gradient algorithm for large-scale nonlinear equations and image restoration problems*, Appl. Numer. Math., 147, 129–141, 2020.
- [186] L. A. Zadeh, *Fuzzy sets*, Inf. Control, 8, 338–353, 1965.
- [187] L. Zhang, *An improved Wei-Yao-Liu nonlinear conjugate gradient method for optimization computation*, Appl. Math. Comput., 215, 2269–2274, 2009.
- [188] Y. Zhang, L. He, C. Hu, J. Guo, J. Li and Y. Shi, *General four-step discrete-time zeroing and derivative dynamics applied to time-varying nonlinear optimization*, Journal of Computational and Applied Mathematics, 347, 314–329, 2019.

- [189] Y. Zhang, W. Li, B. Qiu, Y. Ding and D. Zhang, *Three-state space reformulation and control of MD-included one-link robot system using direct-derivative and zhang-dynamics methods*, in Proceedings of the 2017 29th Chinese Control And Decision Conference (CCDC), 3724–3729, IEEE, Chongqing, China, May 2017.
- [190] Y. Zheng and B. Zheng, *Two new Dai-Liao-type conjugate gradient methods for unconstrained optimization problems*, J. Optim. Theory Appl., 175, 502–509, 2017.
- [191] H. Zhong, G. Chen and X. Guo, *Semi-local convergence of the Newton-HSS method under the center Lipschitz condition*, Numer. Algebra Control Optim., 9, 85–99, 2019.
- [192] W. J. Zhou and D. H. Li, *A globally convergent bfgs method for nonlinear monotone equations without any merit functions*, Math. Comput., 77(264), 2231–2240, 2008.
- [193] W. Zhou and L. Zhang, *A nonlinear conjugate gradient method based on the MBFGS secant condition*, Optim. Methods Softw., 21, 707–714, 2006.
- [194] G. Zoutendijk, *Nonlinear Programming, Computational Methods*. In: J. Abadie (eds.): *Integer and Nonlinear Programming*, North-Holland, 37–86, Amsterdam, 1970.
- [195] N. Zull, N. 'Aini, M. Rivaie and M. Mamat, *A new gradient method for solving linear regression model*, International Journal of Recent Technology and Engineering, 7, 624–630, 2019.

Biografija autora

Branislav D. Ivanov je rođen 29.11.1983. godine u Bosilegradu, gde je završio osnovnu i srednju školu sa odličnim uspehom. Osnovne studije na Prirodno-matematičkom fakultetu u Nišu na Odseku za matematiku i informatiku, smer matematika ekonomije je završio 2009. godine sa prosečnom ocenom 8.48 i stekao diplomu diplomiranog matematičara za matematiku ekonomije-master.

Doktorske akademske studije iz oblasti računarskih nauka upisao je na departmanu za računarske nauke Prirodno-matematičkog fakulteta Univerziteta u Nišu oktobra 2018. godine, pri čemu je položio sve ispite predviđene studijskim programom sa najvišom ocenom.

Od 16. oktobra 2009. godine do 31. decembra 2015. godine bio je zaposlen u zvanju nastavnik matematike u OŠ “Milić Rakić Mirko” u Prokuplju, a od 01. januara 2016. godine do 31. marta 2017. godine obavljao je i posao pomoćnika direktora škole. Za vreme rada u školi položio je ispit za licencu Ministarstva prosvete, nauke i tehnološkog razvoja i steko pravo da samostalno obavlja obrazovno-vaspitni rad u okviru svoje struke. Od 01. aprila 2017. godine do 23. februara 2023. godine obavljao je poslove saradnika u nastavi u zvanju asistenta na Tehničkom fakultetu u Boru Univerziteta u Beogradu, gde je bio angažovan na izvođenju numeričkih i softversko - aplikativnih vežbi na predmetima: Informatika 1, Informatika 2, Programski jezici, Programiranje, Napredne informacione tehnologije, Algoritmi i strukture podataka, Poslovni veb dizajn i Poslovna informatika na osnovnim akademskim studijama. Od 24. februara 2023. godine angažovan je kao asistent za naučnu oblast Računarske nauke na Departmanu za računarske nauke Prirodno-matematičkog fakulteta Univerziteta u Nišu, gde izvodi vežbe na predmetima: Uvod u računarstvo, Uvod u baze podataka, Upravljanje projektima u IT, Numerički metodi 2 i Numerička optimizacija.

U periodu od 3. do 10. oktobra 2010. godine boravio je u Ohridu, Severna Makedonija u okviru DAAD projekta “Center of Excellence for Applications of Mathematics” i stekao sertifikat “Mathematical Approaches to Learning Processes”.

Od strane Prirodno-matematičkog fakulteta Univerziteta u Nišu septembra 2022. godine je dobio povelju i nagradu za izuzetan doprinos nauci postignut kroz publikacije u 2021. godini. Školske 2023/24. godine je dobio zahvalnicu za doprinos realizaciji “Škole prirodno-matematičkih nauka” organizovane od strane Prirodno-matematičkog fakulteta Univerziteta u Nišu i Regionalnog centra za talente Niš.

U periodu od 2018. do 2022. godine bio je član Organizacionog odbora međunarodne konferencije “International May Conference on Strategic Management” – IMKSM.

Do sada je objavio 15 radova, i to: 1 rad u časopisu kategorije M21a, 6 rada u časopisu kategorije M21, 2 rada u časopisu kategorije M22, po 1 rad u časopisima kategorije M51 i M52 i 4 rada u zbornicima međunarodnih konferencija kategorije M33.

Koautor je univerzitetskog udžbenika "Osnove programskog jezika Python", ISBN: 978-86-84531-47-8.

Bibliografija

Naučni radovi objavljeni u naučnim časopisima međunarodnog značaja (M20)

Naučni radovi u izuzetnim međunarodnim časopisima (M21a)

1. **B. Ivanov**, P. S. Stanimirović, B. I. Shaini, H. Ahmad, M.-K. Wang: *A Novel Value for the Parameter in the Dai-Liao-Type Conjugate Gradient Method*, Journal of Function Spaces, ISSN 2314–8896, Vol. 2021, Article ID 6693401, 10 pages, (2021).

Naučni radovi u vrhunskim međunarodnim časopisima (M21)

1. **B. Ivanov**, G. V. Milovanović, P. S. Stanimirović, A. M. Awwal, L. A. Kazakovtsev, V. N. Krutikov: *A modified Dai-Liao conjugate gradient method based on a scalar matrix approximation of hessian and its application*, Journal of Mathematics, ISSN 2314–4629, Vol. 2023, Article ID 9945581, 20 pages, (2023).
2. J. Sabi'u, A. Shah, P. S. Stanimirović, **B. Ivanov**, M. Y. Waziri: *Modified optimal Perry conjugate gradient method for solving system of monotone equations with applications*, Applied Numerical Mathematics, ISSN 0168–9274, Vol. 184, 431–445, (2023).
3. **B. Ivanov**, G.V. Milovanović, P.S. Stanimirović: *Accelerated Dai-Liao projection method for solving systems of monotone nonlinear equations with application to image deblurring*, Journal of Global Optimization, ISSN 0925–5001, Vol. 85, No. 2, 377–420, (2023).
4. **B. Ivanov**, P. S. Stanimirović, G. Milovanović, S. Đorđević, I. Brajević: *Accelerated multiple step-size methods for solving unconstrained optimization problems*, Optimization Methods and Software, ISSN 1055–6788, Vol. 36, No. 5, 998–1029, (2021).
5. P. S. Stanimirović, **B. Ivanov**, H. Ma, D. Mosić: *A survey of gradient methods for solving nonlinear optimization*, Electronic Research Archive, ISSN 2688–1594, Vol. 28, No. 4, 1573–1624, (2020).
6. P. S. Stanimirović, **B. Ivanov**, S. Djordjević, I. Brajević: *New Hybrid Conjugate Gradient and Broyden–Fletcher–Goldfarb–Shanno Conjugate Gradient Methods*, Journal

of Optimization Theory and Applications, ISSN 0022–3239, Vol. 178, No. 3, 860–884, (2018).

Naučni radovi u istaknutim međunarodnim časopisima (M22)

1. P. S. Stanimirović, **B. Ivanov**, D. Stanujkić, V. N. Katsikis, S. D. Mourtas, L. A. Kazakovtsev, S. A. Edalatpanah: *Improvement of Unconstrained Optimization Methods Based on Symmetry Involved in Neutrosophy*, Symmetry, ISSN 2073–8994, Vol. 15, No. 1, Article 250, (2023).
2. P. S. Stanimirović, **B. D. Ivanov**, D. Stanujkić, L. A. Kazakovtsev, V. N. Krutikov, D. Karabašević: *Fuzzy Adaptive Parameter in the Dai–Liao Optimization Method Based on Neutrosophy*, Symmetry, ISSN 2073–8994, Vol. 15, No. 6, Article 1217, (2023).

Radovi uvršteni u zbornike međunarodnih naučnih skupova (M30)

Radovi saopšteni na skupovima međunarodnog značaja, štampani u celini (M33)

1. P. S. Stanimirović, **B. Ivanov**, V. N. Katsikis, S. D. Mourtas: *Neutrosophy in Unconstrained Nonlinear Optimization*, In P. Stanimirovic, A. A. Stupina, E. Semkin, and I. V. Kovalev (Eds.), Hybrid Methods of Modeling and Optimization in Complex Systems, Vol. 1, European Proceedings of Computers and Technology (pp. 131-139), European Publisher, (2023).
2. **B. Ivanov**, D. Stanujkić: *Evaluation of electric vehicles using the simplified WISP method*, In Proc. of International Scientific Conference UNITECH 2022, Gabrovo, 18 - 19 November, 2022, Gabrovo, Bulgaria, pp. 255–260, ISSN: 2603-378X.
3. D. Stanujkić, M. Jevtić, **B. Ivanov**: *An approach for laptop computers evaluation using multiple-criteria decision analysis*, In Proc. of International Scientific Conference UNITECH 2018, Gabrovo, 16 - 17 November, 2018, Gabrovo, Bulgaria, pp. II-83-87, ISSN: 1313-230X.
4. D. Brodić, **B. Ivanov**, M. Jevtić: *The statistical correlation of demographic factor influence to solving facial expression CAPTCHAs*, In Proc. of International Scientific Conference UNITECH 2017, Gabrovo, 17 - 18 November, 2017, Gabrovo, Bulgaria, pp. II-247-251, ISSN: 1313-230X.

Naučni radovi objavljeni u naučnim časopisima nacionalnog značaja (M50)

Naučni radovi u časopisima nacionalnog značaja (M51)

1. **B. Ivanov**, B. I. Shaini, P. S. Stanimirović: *Multiple Use of Backtracking Line Search in Unconstrained Optimization*, Facta Universitatis, Series: Mathematics and Infor-

matics, ISSN 0352–9665, Vol. 35, No. 5, 1417–1438, (2020).

Naučni radovi u časopisima nacionalnog značaja (M52)

1. **B. Ivanov**, M. Jevtić, D. Stanujkić, D. Karabašević, F. Smarandache: *Evaluation of Websites of It Companies From the Perspective of It Beginners*, BizInfo - Journal of Economics, Management and Informatics, ISSN 2217–2769, Vol. 9, No. 2, 1-9, (2018).

Udžbenici, zbirke zadataka, praktikumi i skripta

1. D. Stanujkić, **B. Ivanov**, D. Karabašević, *Osnove programskog jezika Python*, Univerzitetski udžbenik, Univerzitet Privredna akademija u Novom Sadu, Fakultet za primenjeni menadžment, ekonomiju i finansije u Beogradu, Službeni glasnik, Beograd, ISBN: 978-86-84531-47-8, 2020.

Naučni radovi objavljeni u međunarodnim naučnim časopisima

1. P. S. Stanimirović, B. I. Shaini, J. Sabi'u, A. Shah, M. J. Petrović, **B. Ivanov**, X. Cao, A. Stupina, S. Li: *Improved Gradient Descent Iterations for Solving Systems of Nonlinear Equations*, Algorithms, ISSN 1999–4893, Vol. 16, No. 2, Article 64, (2023).

ИЗЈАВА О АУТОРСТВУ

Изјављујем да је докторска дисертација, под насловом

НАПРЕДНИ ГРАДИЈЕНТНИ АЛГОРИТМИ ЗА РЕШАВАЊЕ ПРОБЛЕМА БЕЗУСЛОВНЕ ОПТИМИЗАЦИЈЕ И НЕЛИНЕАРНИХ МОНОТОНИХ СИСТЕМА ЈЕДНАЧИНА ВЕЛИКИХ ДИМЕНЗИЈА

која је одбрањена на Природно-математичком факултету Универзитета у Нишу:

- резултат сопственог истраживачког рада;
- да ову дисертацију, ни у целини, нити у деловима, нисам пријављивао/ла на другим факултетима, нити универзитетима;
- да нисам повредио/ла ауторска права, нити злоупотребио/ла интелектуалну својину других лица.

Дозвољавам да се објаве моји лични подаци, који су у вези са ауторством и добијањем академског звања доктора наука, као што су име и презиме, година и место рођења и датум одбране рада, и то у каталогу Библиотеке, Дигиталном репозиторијуму Универзитета у Нишу, као и у публикацијама Универзитета у Нишу.

У Нишу, _____.

Потпис аутора дисертације:

Бранислав Д. Иванов

**ИЗЈАВА О ИСТОВЕТНОСТИ ШТАМПАНОГ И ЕЛЕКТРОНСКОГ ОБЛИКА
ДОКТОРСKE ДИСЕРТАЦИЈЕ**

Наслов дисертације:

**НАПРЕДНИ ГРАДИЈЕНТНИ АЛГОРИТМИ ЗА РЕШАВАЊЕ ПРОБЛЕМА
БЕЗУСЛОВНЕ ОПТИМИЗАЦИЈЕ И НЕЛИНЕАРНИХ МОНОТОНИХ
СИСТЕМА ЈЕДНАЧИНА ВЕЛИКИХ ДИМЕНЗИЈА**

Изјављујем да је електронски облик моје докторске дисертације, коју сам предао/ла за уношење у **Дигитални репозиторијум Универзитета у Нишу**, истоветан штампаном облику.

У Нишу, _____.

Потпис аутора дисертације:

Бранислав Д. Иванов

ИЗЈАВА О КОРИШЋЕЊУ

Овлашћујем Универзитетску библиотеку „Никола Тесла“ да у Дигитални репозиторијум Универзитета у Нишу унесе моју докторску дисертацију, под насловом:

НАПРЕДНИ ГРАДИЈЕНТНИ АЛГОРИТМИ ЗА РЕШАВАЊЕ ПРОБЛЕМА БЕЗУСЛОВНЕ ОПТИМИЗАЦИЈЕ И НЕЛИНЕАРНИХ МОНОТОНИХ СИСТЕМА ЈЕДНАЧИНА ВЕЛИКИХ ДИМЕНЗИЈА

Дисертацију са свим прилозима предао/ла сам у електронском облику, погодном за трајно архивирање.

Моју докторску дисертацију, унету у Дигитални репозиторијум Универзитета у Нишу, могу користити сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons), за коју сам се одлучио/ла.

1. Ауторство (CC BY)

2. Ауторство – некомерцијално (CC BY-NC)

3. Ауторство – некомерцијално – без прераде (CC BY-NC-ND)

4. Ауторство – некомерцијално – делити под истим условима (CC BY-NC-SA)

5. Ауторство – без прераде (CC BY-ND)

6. Ауторство – делити под истим условима (CC BY-SA)

У Нишу, _____.

Потпис аутора дисертације:

Бранислав Д. Иванов