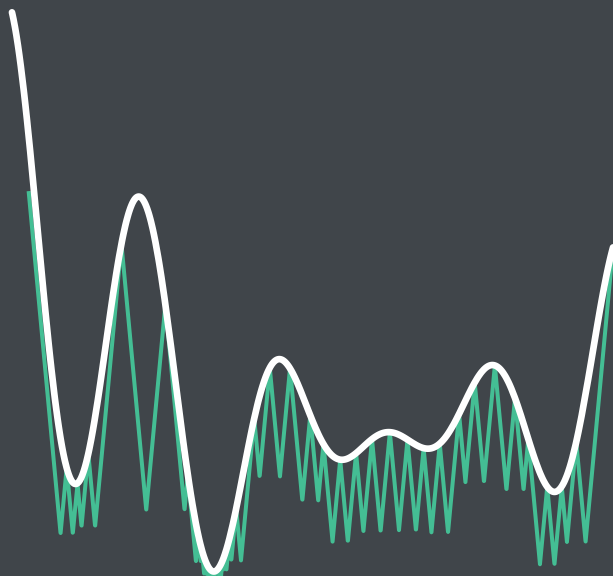


Sveučilište Josipa Jurja Strossmayera u Osijeku
Odjel za matematiku



GLOBALNA OPTIMIZACIJA

Rudolf Scitovski Kristian Sabo Danijel Grahovac

Sveučilište Josipa Jurja Strossmayera u Osijeku
Odjel za matematiku

Rudolf Scitovski, Kristian Sabo, Danijel Grahovac
GLOBALNA OPTIMIZACIJA

Osijek, 2017.

Prof. dr. sc. Rudolf Scitovski
Prof. dr. sc. Kristian Sabo
Doc. dr. sc. Danijel Grahovac
Sveučilište Josipa Jurja Strossmayera u Osijeku – Odjel za matematiku
Trg Lj. Gaja 6
HR-31 000 Osijek

Izdavač:

Sveučilište Josipa Jurja Strossmayera u Osijeku – Odjel za matematiku

Recenzenti:

Prof. dr. sc. Dragan Jukić

Odjel za matematiku Sveučilišta Josipa Jurja Strossmayera u Osijeku

Prof. dr. sc. Miljenko Marušić

Matematički odsjek Prirodoslovno-matematičkog fakulteta Sveučilišta u Zagrebu

Lektorica:

Ivanka Ferčec

Fakultet elektrotehnike, računarstva i informacijskih tehnologija, Sveučilište Josipa Jurja Strossmayera u Osijeku

CIP zapis dostupan je u računalnom katalogu Gradske
i sveučilišne knjižnice Osijek pod brojem 140804087.

ISBN 978-953-8154-06-5

Ovaj udžbenik objavljuje se uz suglasnost Senata Sveučilišta Josipa Jurja Strossmayera u Osijeku pod brojem 22/17.

Ovaj udžbenik objavljuje se uz financijsku pomoć Ministarstva znanosti i obrazovanja Republike Hrvatske.

© Rudolf Scitovski, Kristian Sabo, Danijel Grahovac 2017.

Tisak: STUDIO HS Internet d.o.o. Osijek

PREDGOVOR

Sadržaj ovog udžbenika izvodio se posljednjih nekoliko godina u okviru predmeta **Matematički praktikum** u 3. semestru svih smjerova sveučilišnog diplomskog studija matematike i u 9. semestru integriranog nastavničkog studija matematike i informatike na Odjelu za matematiku Sveučilišta u Osijeku (30 sati predavanja i 30 sati seminara, 6 ECTS bodova).

Tekst je pisan tako da podrazumijeva poznavanje osnova matematičke analize, linearne algebre, numeričke matematike i programiranja, a namijenjen je studentima diplomskih studijskih programa u STEM područjima.

Brojni ilustrativni primjeri doprinose razumijevanju izložene materije. Osim toga, u okviru svakog poglavlja nalaze se brojni zadaci: od sasvim jednostavnih, do onih koji mogu poslužiti kao teme seminarskih i sličnih radova. Izrada većine zadataka povezana je s mogućnošću korištenja programskih sustava *Mathematica* i *Matlab*, za koje Sveučilište u Osijeku već godinama raspolaže odgovarajućim brojem licenci.

Sadržaj naveden u ovom udžbeniku, kao i priloženi *Mathematica*-programi, mogu poslužiti i u nekim praktičnim istraživanjima. Za sve metode navedene u udžbeniku izrađeni su odgovarajući moduli, čiji je opis dan na kraju udžbenika, a *Mathematica*-kodovi slobodno su dostupni na adresi: <http://www.mathos.unios.hr/GOP-knjiga/GOPmoduli.rar>.

Opsežna recentna literatura, navedena na kraju udžbenika, koja obuhvaća brojne knjige i odgovarajuće članke iz renomiranih međunarodnih znanstvenih časopisa, daje neophodan pregled najvažnijih spoznaja, a također može poslužiti i za nastavak samostalnog rada u ovom znanstvenom području.

Zbog specifičnih zahtjeva matematičkog teksta i sličnih zahtjeva u programskim sustavima *Mathematica*, *Matlab*, *FORTRAN*, cijeli dio od decimalnog dijela decimalnog broja odvajan je decimalnom točkom (.), a ne decimalnim zarezom (,).

Svi teoremi, leme, definicije, slike, tablice, primjedbe, primjeri i zadaci u tekstu imaju svoju jedinstvenu oznaku i na taj način pozivaju se u cijelom tekstu. Zbog toga su pisani velikim početnim slovom¹.

Zahvaljujemo recenzentima *prof. dr. sc. Draganu Jukiću* i *prof. dr. sc. Miljenku Marušiću* te lektorici *Ivanki Ferčec*, koji su svojim primjedbama i prijedlozima značajno pomogli da ovaj tekst bude bolji.

Osijek, 1. rujna 2017.

Rudolf Scitovski
Kristian Sabo
Danijel Grahovac

¹Za pregledavanje ovog udžbenika Adobe Readerom možete koristiti sljedeće pogodnosti:

- klikom na naslov nekog poglavlja u Sadržaju dolazite na to poglavlje. Povratak (na isto mjesto odakle ste krenuli) je držanjem tipke **Alt** pa nakon toga pritisnuti tipku **<**;
- klikom na oznaku nekog teorema, leme, definicije, slike, tablice, primjedbe, primjera ili zadatka u tekstu odlazite na taj objekt. Povratak je na prethodno opisani način;
- klikom na oznaku neke reference u tekstu odlazite na tu referencu u Literaturi na kraju knjige. Povratak je na prethodno opisani način;
- klikom na oznaku stranice u Indeksu na kraju knjige odlazite na taj pojam u knjizi. Povratak je na prethodno opisani način.

Sadržaj

1	Uvod i motivacija	1
1.1	Optimizacijski problem	2
1.2	Ilustrativni primjeri	3
1.3	Konveksne funkcije	24
1.4	Kvazikonveksne funkcije	27
1.4.1	Unimodalne funkcije	32
1.5	Diferencijabilne funkcije	33
2	Klasične metode optimizacije	35
2.1	Metode spusta za konveksne funkcije	35
2.1.1	Koordinatna relaksacija	36
2.1.2	Gradijentna metoda	37
2.1.3	Newtonova metoda za minimizaciju bez ograničenja	37
2.1.4	Kvazi-Newtonova metoda	41
2.2	Nelder-Meadova metoda	41
2.2.1	Nelder-Meadova metoda za funkcije dviju varijabli	41
2.3	Kvadratne interpolacijske metode za jednodimenzionalnu minimizaciju	46
2.3.1	Newtonova metoda – metoda jedne točke	46
2.3.2	Metoda dvije točke	47
2.3.3	Metoda tri točke	49
2.3.4	Numerički primjer	51
3	Jednodimenzionalna minimizacija strogo kvazikonveksnih funkcija	53
3.1	Metoda tangenti za konveksnu derivabilnu funkciju	53
3.2	Metoda polovljenja za kvazikonveksnu derivabilnu funkciju	57
3.3	Metoda polovljenja za nederivabilnu strogo kvazikonveksnu funkciju	60

3.4	Metoda zlatnog reza	63
4	Jednodimenzionalna globalna optimizacija	69
4.1	Lipschitz-neprekidne funkcije	69
4.2	Pijavskijeva metoda slomljenih pravaca	74
4.3	Shubertova metoda	84
4.4	Optimizacijski algoritam DIRECT	86
4.4.1	Optimizacija algoritma DIRECT	89
4.4.2	Traženje potencijalno optimalnih intervala	91
5	Višedimenzionalna globalna optimizacija	101
5.1	Transformacija problema na hiperkocku	102
5.2	Optimizacijski algoritam DIRECT za funkciju dviju varijabli	104
5.2.1	Dijeljenje kvadrata $[0, 1]^2$	104
5.2.2	Nekoliko primjera	107
5.3	Optimizacijski algoritam DIRECT za funkciju više varijabli	109
5.4	Optimizacijski algoritam DIRECT za simetričnu funkciju	111
5.5	Neke druge mogućnosti	116
5.5.1	Kratki povijesni pregled	116
5.5.2	Metoda grananja i ograđivanja	117
6	Mathematica programi i moduli	121
6.1	Uvod	121
6.2	Klasične metode optimizacije	122
6.2.1	Metode spusta za konveksne funkcije	122
6.2.2	Nelder-Meadova metoda	126
6.2.3	Kvadratne interpolacijske metode za jednodimenzionalnu minimizaciju	128
6.3	Jednodimenzionalna minimizacija kvazikonveksnih funkcija	130
6.3.1	Mathematica-moduli za traženje globalnog minimuma konveksne i kvazikonveksne derivabilne funkcije	131
6.3.2	Mathematica-moduli za traženje globalnog minimuma kvazikonveksne nederivabilne funkcije	134
6.3.3	Numerički eksperimenti	135
6.4	Jednodimenzionalna globalna optimizacija	138
6.4.1	Pijavskijeva metoda slomljenih pravaca	138
6.4.2	Shubertova metoda	139
6.4.3	Optimizacijski algoritam DIRECT – poznata Lipschit- zova konstanta	139

6.4.4	Optimizacijski algoritam DIRECT – nepoznata Lipschitzova konstanta	142
6.5	Višedimenzionalna globalna optimizacija	143
6.5.1	Optimizacijski algoritam DIRECT za funkciju više varijabli	143
6.5.2	Optimizacijski algoritam SymDIRECT za funkciju više varijabli	144
	Literatura	146
	Indeks	154

Poglavlje 1

Uvod i motivacija

Često se u mnogim primijenjenim istraživanjima problem opisuje nekom funkcijom cilja uz koju se mogu dodati odgovarajuća ograničenja. Zato razlikujemo optimizacijske probleme bez ograničenja i optimizacijske probleme s ograničenjima. Problem može imati više lokalno optimalnih, ali samo jedno globalno optimalno rješenje koje se može postići u jednoj ili više točaka. Ako su pri tome funkcija cilja i ograničenja kojima je definiran problem linearnog tipa, problem se obično svodi na rješavanje različitih problema linearnog programiranja [7, 25, 35].

Ako je funkcija cilja ili ograničenja kojima je definiran problem nelinearnog tipa, govorimo o nelinearnom problemu lokalne ili globalne optimizacije. Ovakvi slučajevi često se javljaju u različitim problemima klaster analize [11, 32, 43, 59], problemima prepoznavanja oblika [15, 73], različitim fizikalnim problemima, problemima mehanike kontinuuma, itd.

Ako se optimizacijski problem postavlja kao problem minimizacije, odgovarajuća funkcija cilja može imati jedan ili više lokalnih minimuma. U tom slučaju posebnu važnost dobiva traženje globalnog minimuma, pri čemu funkcija cilja može imati veliki broj nezavisnih varijabli, može imati veliki broj lokalnih minimuma, ne mora biti diferencijabilna (pa čak ni neprekidna), ne mora biti konveksna, a područje na kome je definirana ne mora biti konveksno, pa čak ni povezano. Više ovakvih primjera može se vidjeti kod [20, 37, 50, 70].

Ne postoji jedna opća metoda kojom bi se efikasno rješavali svi ovakvi problemi. Zbog toga se u literaturi mogu pronaći različite specijalizirane metode optimizacije. Neke od njih, koje se najčešće koriste, opisat ćemo u ovom udžbeniku.

U ovom poglavlju najprije ćemo definirati pojmove lokalnog i globalnog

minimuma, a zatim navesti veći broj tipičnih primjera čije rješavanje vodi na probleme globalne optimizacije. Ovi primjeri ilustriraju probleme koji se rješavaju metodama i algoritmima navedenim u ovom udžbeniku.

1.1 Optimizacijski problem

Definicija 1.1. [46] Kažemo da funkcija $f: \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$ u točki $x^* \in \mathcal{D}$ postiže lokalni minimum ako postoji okolina $\mathcal{O}(x^*)$ takva da je $f(x) \geq f(x^*)$ za sve $x \in \mathcal{O}(x^*) \cap \mathcal{D}$. Točku x^* zovemo točka lokalnog minimuma ili lokalni minimizator funkcije f .

Kažemo da je x^* točka strogo lokalnog minimuma ili strogi lokalni minimizator funkcije f ako postoji okolina $\mathcal{O}(x^*)$ takva da je $f(x) > f(x^*)$ za sve $x \in \mathcal{O}(x^*) \cap \mathcal{D} \setminus \{x^*\}$.

Definicija 1.2. Kažemo da funkcija $f: \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$ u točki $x^* \in \mathcal{D}$ postiže globalni minimum na \mathcal{D} ako je

$$f(x^*) \leq f(x), \quad \forall x \in \mathcal{D}.$$

Točku $x^* \in \mathcal{D}$ zovemo točka globalnog minimuma ili globalni minimizator funkcije f na \mathcal{D} . Skup svih točaka globalnog minimuma funkcije f označavamo s

$$\operatorname{argmin}_{x \in \mathcal{D}} f(x),$$

vrijednost

$$f(\hat{x}), \quad \text{za } \hat{x} \in \operatorname{argmin}_{x \in \mathcal{D}} f(x),$$

zovemo globalni minimum funkcije f na \mathcal{D} , a funkciju f nazivamo minimizirajuća funkcija.

Primjedba 1.1. Analogno definiciji globalnog minimuma može se definirati pojam globalnog maksimuma i točka globalnog maksimuma funkcije f , ali kako je

$$\begin{aligned} (i) \quad & \max_{x \in \mathcal{D}} f(x) = - \min_{x \in \mathcal{D}} (-f(x)), \\ (ii) \quad & \operatorname{argmax}_{x \in \mathcal{D}} f(x) = \operatorname{argmin}_{x \in \mathcal{D}} (-f(x)), \end{aligned}$$

dovoljno je proučavati samo problem globalnog minimuma.

1.2 Ilustrativni primjeri

Na početku navedimo nekoliko praktičnih primjera u kojima se pojavljuje problem traženja globalnog minimuma funkcije jedne ili više varijabli.

Primjer 1.1. (*Udaljenost točke do pravca [27]*)

Treba izračunati ℓ_p , $1 \leq p \leq \infty$, udaljenost točke $T_0 = (x_0, y_0)$ do pravca π koji je zadan jednadžbom $y = kx + l$, $k, l \in \mathbb{R}$.

Rješavanje ovog problema svodi se na relativno jednostavni problem globalne optimizacije općenito nederivabilne funkcije jedne varijable za koji ćemo pronaći eksplicitno rješenje.

Prisjetimo se kako ℓ_p -udaljenost $d_p(T_0, \pi)$ točke T_0 do pravca π definiramo kao

$$d_p(T_0, \pi) = \min_{x \in \mathbb{R}} (|kx + l - y_0|^p + |x - x_0|^p)^{1/p}, \quad 1 \leq p < \infty,$$

te

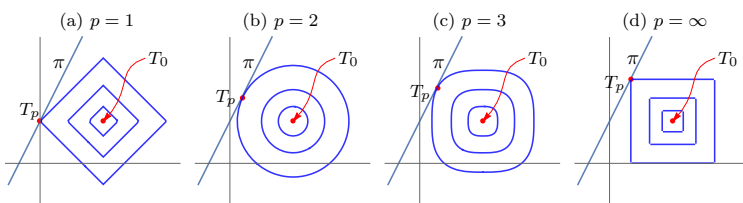
$$d_\infty(T_0, \pi) = \min_{x \in \mathbb{R}} \max\{|kx + l - y_0|, |x - x_0|\}.$$

Geometrijski gledano, udaljenost točke T_0 do pravca π možemo interpretirati na sljedeći način. Neka je

$$S_p(T_0, \varepsilon) = \{T \in \mathbb{R}^2 : d_p(T, T_0) = \varepsilon\}$$

ℓ_p -kružnica, $1 \leq p \leq \infty$, sa središtem u točki T_0 malenog polumjera $\varepsilon > 0$. Napuhujemo li kružnicu $S_p(T_0, \varepsilon)$, ona će u jednom trenutku za neki $\varepsilon_p > 0$ dotaknuti pravac π . Polumjer tako dobivene kružnice $S_p(T_0, \varepsilon_p)$ očigledno predstavlja ℓ_p -udaljenost točke T_0 do pravca π (vidi Sliku 1.1). Pri tome točku $T_p = (\xi_p, \eta_p)$ na pravcu π koja je u smislu ℓ_p -udaljenosti najbliža točki T_0 zovemo ℓ_p -projekcija točke T_0 na pravac π . Sa Slike 1.1 odmah je jasno da ℓ_1 i ℓ_∞ -projekcije točke na pravac općenito ne moraju biti jedinstvene.

Određivanje apscise ℓ_p -projekcije točke T_0 problem je globalnog minimuma, pri čemu je apscisa ℓ_p -projekcije odgovarajući minimizator. Ovaj problem globalnog minimuma moguće je eksplicitno riješiti.



Slika 1.1: Točke T_0, T_p , pravac π i kružnice $S_p(T_0, \varepsilon)$, $p = 1, 2, 3, \infty$

Iz tehničkih ćemo razloga posebno razmatrati slučaj ℓ_1 -udaljenosti (vidi Sliku 1.1a). Pri tome u izvodu razlikujemo dvije različite mogućnosti: $k = 0$ te $k \neq 0$.

Ako je $k = 0$, onda funkcija $x \mapsto |l - y_0| + |x - x_0|$ postiže globalni minimum u točki $x = x_0$ i vrijedi

$$d_1(T_0, \pi) = |l - y_0|. \quad (1.1)$$

Pretpostavimo da je $k \neq 0$. Funkcija

$$x \mapsto |kx + l - y_0| + |x - x_0|, \quad (1.2)$$

omeđena je odozdo i vrijedi

$$\lim_{x \rightarrow \infty} |kx + l - y_0| + |x - x_0| = \lim_{x \rightarrow -\infty} |kx + l - y_0| + |x - x_0| = \infty,$$

pa postoji točka iz \mathbb{R} u kojoj se postiže njezin globalni minimum. Nadalje, funkcija (1.2) je po dijelovima linearna i nije derivabilna u točkama $\frac{y_0 - l}{k}$ i x_0 tako da se prema tome globalni minimum funkcije (1.2) postiže u jednoj od tih dviju točaka. Očigledno je

$$\begin{aligned} d_1(T_0, \pi) &= \min \left\{ |kx_0 + l - y_0|, \left| \frac{y_0 - l}{k} - x_0 \right| \right\} \\ &= \min \left\{ |kx_0 + l - y_0|, \frac{|kx_0 + l - y_0|}{|k|} \right\} \\ &= \frac{|kx_0 + l - y_0|}{\max\{|k|, 1\}}. \end{aligned} \quad (1.3)$$

Konačno iz formula (1.1) i (1.3) dobivamo

$$d_1(T_0, \pi) = \frac{|kx_0 + l - y_0|}{\max\{|k|, 1\}}.$$

U nastavku razmotrimo problem jedinstvenosti globalnog minimuma. S $T_1 = (\xi_1, \eta_1)$ označimo ℓ_1 -projekciju točke T_0 na pravac π . Uočimo da se za $|k| > 1$ minimum funkcije (1.2) postiže u $x = \frac{y_0 - l}{k}$, dok se za $|k| < 1$ minimum funkcije (1.2) postiže u $x = x_0$. Očito je

$$\xi_1 = \begin{cases} \frac{y_0 - l}{k}, & |k| > 1, \\ x_0, & |k| < 1, \end{cases}, \quad \eta_1 = k\xi_1 + l = \begin{cases} y_0, & |k| > 1, \\ kx_0 + l, & |k| < 1. \end{cases} \quad (1.4)$$

Pokažimo da u slučaju $|k| = 1$ minimum funkcije (1.2) nije jedinstven i da se postiže na bilo kojoj konveksnoj kombinaciji brojeva $\frac{y_0 - l}{k}$ i x_0 , tj. za svaki

$$\xi_1(\lambda) = \lambda \frac{y_0 - l}{k} + (1 - \lambda)x_0, \quad \lambda \in [0, 1]. \quad (1.5)$$

Zaista, za $|k| = 1$ te $\lambda \in [0, 1]$ imamo

$$\begin{aligned} |k \xi_1(\lambda) + l - y_0| &+ |\xi_1(\lambda) - x_0| \\ &= \left| k \left(\lambda \frac{y_0 - l}{k} + (1 - \lambda)x_0 \right) + l - y_0 \right| + \left| \lambda \frac{y_0 - l}{k} + (1 - \lambda)x_0 - x_0 \right| \\ &= (1 - \lambda)|kx_0 + l - y_0| + \lambda |kx_0 + l - y_0| \\ &= |kx_0 + l - y_0| = d_1(T_0, \pi). \end{aligned}$$

Zaključujemo kako je ℓ_1 -projekcija točke T_0 na pravac π u slučaju $|k| \neq 1$ jedinstvena točka $T_1 = (\xi_1, \eta_1)$ zadana s (1.4), dok je ℓ_1 -projekcija točke T_0 na pravac π u slučaju $|k| = 1$ svaka točka oblika $(\xi_1(\lambda), k\xi_1(\lambda) + l - y_0)$, $\lambda \in [0, 1]$, pri čemu je $\xi_1(\lambda)$ zadan s (1.5).

Nadalje razmatramo slučaj ℓ_p -udaljenosti $1 < p < \infty$ točke T_0 do pravca π zadanog u eksplicitnom obliku (vidi Slike 1.1b-c). U općem slučaju naš problem može se svesti na minimizaciju funkcije

$$x \mapsto (|kx + l - y_0|^p + |x - x_0|^p)^{1/p}, \quad 1 < p < \infty.$$

Kako je funkcija $x \mapsto \sqrt[p]{x}$, monotono rastuća na $[0, \infty)$, naš se problem može svesti na minimizaciju funkcije zadane s

$$\varphi_p(x) = |kx + l - y_0|^p + |x - x_0|^p, \quad 1 < p < \infty.$$

Budući da je funkcija φ_p omeđena odozdo i da vrijedi

$$\lim_{x \rightarrow \infty} |kx + l - y_0|^p + |x - x_0|^p = \lim_{x \rightarrow -\infty} |kx + l - y_0|^p + |x - x_0|^p = \infty,$$

postoji točka iz \mathbb{R} u kojoj se postiže njezin globalni minimum. S ciljem određivanja te točke posebno promatramo dva slučaja: $k = 0$ te $k \neq 0$.

Ako je $k = 0$, onda je $\varphi_p(x) = |l - y_0|^p + |x - x_0|^p$, te se njezin minimum postiže u točki $x = x_0$ tako da je

$$d_p(T_0, \pi) = |l - y_0|. \quad (1.6)$$

Neka je $k \neq 0$. Primijetimo da je funkcija φ_p derivabilna u svim točkama na \mathbb{R} , osim možda u točkama $x = x_0$ i $x = \frac{y_0 - l}{k}$, pa u svrhu minimizacije funkcije φ_p možemo upotrijebiti znanja iz diferencijalnog računa funkcije jedne varijable.

Promatramo funkciju φ_p na skupu $\mathbb{R} \setminus \{x_0, \frac{y_0 - l}{k}\}$. Bez smanjenja općenitosti pretpostavimo da je $x_0 < \frac{y_0 - l}{k}$ (slučaj $x_0 \geq \frac{y_0 - l}{k}$ može se analizirati potpuno analogno).

Pokažimo najprije da funkcija φ_p nema stacionarnih točaka na intervalima $\langle -\infty, x_0 \rangle$ te $\langle \frac{y_0 - l}{k}, +\infty \rangle$. Ako je $x \in \langle -\infty, x_0 \rangle$, funkcija φ_p je derivabilna u x te je

$$\begin{aligned} \frac{d\varphi_p(x)}{dx} &= |k|^p p |x - \frac{y_0 - l}{k}|^{p-1} \text{sign}(x - \frac{y_0 - l}{k}) + p|x - x_0|^{p-1} \text{sign}(x - x_0) \\ &= -|k|^p p |x - \frac{y_0 - l}{k}|^{p-1} - p|x - x_0|^{p-1} < 0, \end{aligned}$$

gdje je $\text{sign}(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0, \end{cases}$ tzv. funkcija predznaka.

Ako je $x \in \langle \frac{y_0 - l}{k}, +\infty \rangle$, funkcija φ_p je derivabilna u x i vrijedi

$$\frac{d\varphi_p(x)}{dx} = |k|^p p |x - \frac{y_0 - l}{k}|^{p-1} + p|x - x_0|^{p-1} > 0.$$

Preostaje analizirati slučaj $x \in \langle x_0, \frac{y_0-l}{k} \rangle$. Na tom je skupu funkcija φ_p također derivabilna te je

$$\frac{d\varphi_p(x)}{dx} = -|k|^p p |x - \frac{y_0-l}{k}|^{p-1} + p|x - x_0|^{p-1}. \quad (1.7)$$

Rješenje jednadžbe $\frac{d\varphi_p(x)}{dx} = 0$ glasi

$$\xi_p = \frac{1}{|k|^{\frac{p}{p-1}} + 1} x_0 + \frac{|k|^{\frac{p}{p-1}}}{|k|^{\frac{p}{p-1}} + 1} \frac{y_0 - l}{k}. \quad (1.8)$$

Uočimo da je $\xi_p \in \langle x_0, \frac{y_0-l}{k} \rangle$ te

$$\varphi_p(\xi_p) = \left(\frac{|kx_0 + l - y_0|}{(|k|^{\frac{p}{p-1}} + 1)^{\frac{p-1}{p}}} \right)^p.$$

Kandidati za točku u kojoj se postiže globalni minimum funkcije φ_p su $x_0, \frac{y_0-l}{k}$ i ξ_p zadana s (1.8). Pokažimo da je

$$\min \left\{ \varphi_p(x_0), \varphi_p\left(\frac{y_0-l}{k}\right), \varphi_p(\xi_p) \right\} = \varphi_p(\xi_p).$$

U tu svrhu uočimo da za svaki $1 < p < \infty$ i svaki $k \in \mathbb{R}$ vrijedi

$$\left(|k|^{\frac{p}{p-1}} + 1 \right)^{\frac{p-1}{p}} > |k|, \quad (1.9)$$

te

$$\left(|k|^{\frac{p}{p-1}} + 1 \right)^{\frac{p-1}{p}} > 1. \quad (1.10)$$

Prema (1.9) i (1.10) imamo

$$\min \left\{ \left(\frac{|kx_0 + l - y_0|}{|k|} \right)^p, |kx_0 + l - y_0|^p, \left(\frac{|kx_0 + l - y_0|}{(|k|^{\frac{p}{p-1}} + 1)^{\frac{p-1}{p}}} \right)^p \right\} = \left(\frac{|kx_0 + l - y_0|}{(|k|^{\frac{p}{p-1}} + 1)^{\frac{p-1}{p}}} \right)^p.$$

Konačno zaključujemo da je $d_p(T_0, \pi) = (\varphi(\xi_p))^{1/p}$, odakle je

$$d_p(T_0, \pi) = \frac{|kx_0 + l - y_0|}{\left(|k|^{\frac{p}{p-1}} + 1 \right)^{\frac{p-1}{p}}}, \quad 1 < p < \infty. \quad (1.11)$$

Iz izvoda formule (1.11) neposredno proizlazi da je $T_p = (\xi_p, k\xi_p + l)$ pripadna l_p -projekcija točke T_0 na pravac π , gdje je ξ_p dan s (1.8).

Ako s $q \in \mathbb{R}$ označimo tzv. konjugirani eksponent od p , odnosno broj sa svojstvom da je $\frac{1}{p} + \frac{1}{q} = 1$, onda formulu (1.11) možemo zapisati u obliku

$$d_p(T_0, \pi) = \frac{|kx_0 + l - y_0|}{(|k|^q + 1)^{\frac{1}{q}}} = \frac{|kx_0 + l - y_0|}{d_q((k, 1), (0, 0))}.$$

Zadatak 1.1. Pokažite da je ℓ_∞ -udaljenost točke $T_0 \in \mathbb{R}^2$ do pravca π zadanog u eksplicitnom obliku $y = kx + l$ dana s ([41])

$$d_\infty(T_0, \pi) = \frac{|kx_0 + l - y_0|}{d_1((k, 1), (0, 0))}.$$

Primjer 1.2. (Euklidska udaljenost točke do parabole)

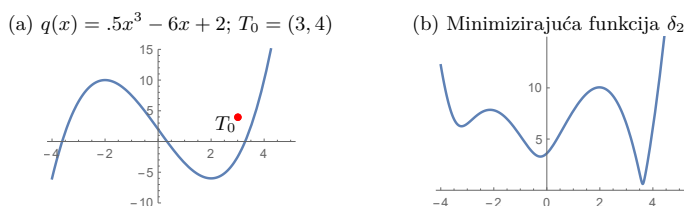
Treba izračunati ℓ_2 udaljenost točke $T_0 = (3, 4)$ do kubne parabole zadane funkcijom $q: \mathbb{R} \rightarrow \mathbb{R}$, $q(x) = .5x^3 - 6x + 2$ (vidi Sliku 1.2a).

Primijetite da je ℓ_2 udaljenost točke $T_0 = (x_0, y_0)$ do neke točke $T = (x, q(x))$ na grafu funkcije q zadana s (vidi Sliku 1.2b)

$$\delta_2(x) = d_2(T_0, T) = \sqrt{(x - x_0)^2 + (q(x) - y_0)^2},$$

a udaljenost točke T_0 do grafa funkcije q zadana je s

$$d_2(T_0, q) = \min_{x \in \mathbb{R}} d_2(T_0, (x, q(x))).$$

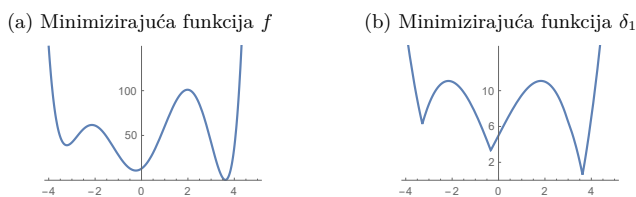


Slika 1.2: Euklidska udaljenost točke T_0 do kubne parabole q

Određivanje točke $T^* = (x^*, q(x^*))$ na kojoj se postiže minimalna udaljenost točke T_0 do grafa funkcije q jedan je problem globalne optimizacije nelinearne derivabilne funkcije jedne varijable koja ima više od jednog lokalnog minimuma. Za razliku od problema iz Primjera 1.1, ovaj problem općenito nije moguće riješiti eksplicitno.

Primijetite da je funkcija δ_2 neprekidno derivabilna na \mathbb{R} . Budući da je funkcija $x \mapsto \sqrt{x}$ monotono rastuća funkcija, naš se problem može svesti na rješavanje jednog nelinearnog problema najmanjih kvadrata (Least Squares Problem (LS)) koji se nadalje može svesti na određivanje globalnog minimuma sljedećeg polinoma 6-tog stupnja (vidi Sliku 1.3a)

$$f(x) = (x - x_0)^2 + (q(x) - y_0)^2. \tag{1.12}$$

Slika 1.3: LS i ℓ_1 udaljenost točke T_0 do parabole q **Primjer 1.3.** (ℓ_1 -udaljenost točke do parabole)

Treba odrediti ℓ_1 -udaljenost točke $T_0 = (3, 4)$ do kubne parabole zadane funkcijom $q: \mathbb{R} \rightarrow \mathbb{R}$, $q(x) = .5x^3 - 6x + 2$.

Primijetite da je ℓ_1 -udaljenost točke T_0 do neke točke $T = (x, q(x))$ na grafu funkcije q zadana s (vidi Sliku 1.3b)

$$\delta_1(x) = d_1(T_0, T) = |x - x_0| + |q(x) - y_0|,$$

pa se određivanje udaljenosti točke T_0 do grafa funkcije q može interpretirati kao problem određivanja globalnog minimuma funkcije δ_1 definirane na \mathbb{R} . Dakle, i u ovom primjeru radi se o problemu jednodimenzionalne globalne minimizacije, ali ovaj puta minimizirajuća funkcija nije derivabilna. Također, iz slike se može uočiti da ova funkcija ima tri lokalna minimuma od kojih je jedan ujedno i globalni minimum.

Zadatak 1.2. Kubna parabola zadana je funkcijom $q: \mathbb{R} \rightarrow \mathbb{R}$, $q(x) = .5x^3 - 6x + 2$.

- Gdje treba postaviti točku $T_0 = (x_0, y_0)$ tako da funkcija $f: \mathbb{R} \rightarrow \mathbb{R}$, $f(x) = (x - x_0)^2 + (q(x) - y_0)^2$ postigne svoj globalni minimum u barem dvije različite točke?
- Gdje treba postaviti točku $T_0 = (x_0, y_0)$ tako da funkcija $\delta_1: \mathbb{R} \rightarrow \mathbb{R}$, $\delta_1(x) = |x - x_0| + |q(x) - y_0|$ postigne svoj globalni minimum u barem dvije različite točke?

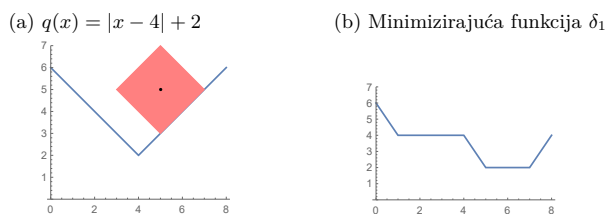
Primjer 1.4. (ℓ_1 -udaljenost točke do grafa funkcije)

Treba odrediti ℓ_1 -udaljenost točke $T_0 = (5, 5)$ do grafa funkcije $q: \mathbb{R} \rightarrow \mathbb{R}$, $q(x) = |x - 4| + 2$ (vidi Sliku 1.4a).

Slično kao u prethodnom primjeru, ℓ_1 -udaljenost točke T_0 do neke točke $T = (x, q(x))$ na grafu funkcije q zadana je s (vidi Sliku 1.4)

$$\delta_1(x) = d_1(T_0, T) = |x - x_0| + |q(x) - y_0|.$$

Određivanje udaljenosti točke T_0 do grafa funkcije q može se također interpretirati kao problem određivanja globalnog minimuma funkcije δ_1 na \mathbb{R} . Dakle, i u



Slika 1.4: ℓ_1 udaljenost točke $T_0 = (5, 5)$ do funkcije $q(x) = |x - 4| + 2$

ovom primjeru radi se o problemu jednodimenzionalne globalne minimizacije nederivabilne funkcije. Iz Slike 1.4b može se uočiti da minimizirajuća funkcija δ_1 ima beskonačno mnogo točaka globalnog minimuma, a vrijednost globalnog minimuma je $\delta_1^* = 2$, tj.

$$\delta_1(x^*) = 2, \forall x^* \in \underset{x \in \mathbb{R}}{\operatorname{argmin}} \delta_1(x) = [5, 7].$$

Zadatak 1.3. Kako će izgledati grafovi funkcije δ_1 iz prethodnog primjera ako je $T_0 = (4, 4)$, $T_0 = (2, 5)$ i $T_0 = (2, 3)$?

Zadatak 1.4. Neka su točka T_0 i funkcija q zadane kao u prethodnom primjeru. Odredite ℓ_2 i ℓ_∞ udaljenost točke T_0 do grafa funkcije q .

Zadatak 1.5. Zadana je točka $T_0 = (x_0, y_0, z_0) \in \mathbb{R}^3$ i neprekidna funkcija $q: \mathbb{R}^2 \rightarrow \mathbb{R}$. Definirajte problem određivanja udaljenosti točke T_0 do grafa funkcije q za slučaj LS-kvazimetričke funkcije $d_{LS}: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}_+$, $d_{LS}(x, y) = \|x - y\|_2^2$ i za slučaj ℓ_1 metričke funkcije $d_1: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}_+$, $d_1(x, y) = \|x - y\|_1$. Nekom metodom minimizacije riješite problem ako je $T_0 = (1, 2, 3)$ i $q(x, y) = x^2 + y^2$. Izradite odgovarajuće ilustracije primjenom programskog sustava *Mathematica*.

Primjer 1.5. (*Centroid skupa $\mathcal{A} \subset \mathbb{R}^n$*)

Zadan je skup $\mathcal{A} = \{a^i \in \mathbb{R}^n: i = 1, \dots, m\}$ s odgovarajućim težinama $w_1, \dots, w_m > 0$. Treba pronaći točku $c^* = (c_1^*, \dots, c_n^*) \in \mathbb{R}^n$, za koju je težinska suma kvadrata euklidskih udaljenosti do točaka skupa \mathcal{A} minimalna.

Tražena točka c^* točka je u kojoj se postiže globalni minimum funkcije

$$F_{LS}(x) = \sum_{i=1}^m w_i \|x - a^i\|_2^2 = \sum_{i=1}^m w_i \sum_{s=1}^n (x_s - a_s^i)^2, \quad (1.13)$$

odnosno točka $c^* \in \underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \sum_{i=1}^m w_i \|x - a^i\|_2^2$ je globalni minimizator funkcije F_{LS} .

Točku c^* zovemo centroid skupa \mathcal{A} ¹. U ovom slučaju radi se o problemu globalne optimizacije kvadratne funkcije više varijabli koji uvijek ima rješenje.

¹U fizici se ovaj problem povezuje s problemom određivanja centra mase sustava čestica, a centroid se tada naziva težište tijela ili Steinerova točka.

Zadatak 1.6. Pokažite da je $c^* = \frac{1}{W} \sum_{i=1}^m w_i a^i$, $W = \sum_{i=1}^m w_i$, jedinstvena točka globalnog minimuma funkcije $F_{LS}: \mathbb{R}^n \rightarrow \mathbb{R}^n$, $F_{LS}(x) = \sum_{i=1}^m w_i \|x - a^i\|_2^2$. Izvedite dokaz najprije za $n = 1$, a nakon toga općenito.

Zadatak 1.7. Odredite težinski centroid skupa $\mathcal{A} = \{(0, 1, 2), (2, 8, 0), (-1, 9, 4), (1, 6, 5), (-2, 8, 3)\} \subset \mathbb{R}^3$ s težinama

- (a) $w_1 = \dots = w_5 = 1$,
 (b) $w_1 = 1, w_2 = 2, w_3 = 1, w_4 = 1, w_5 = 2$.

Primjer 1.6. (Medijan skupa $\mathcal{A} \subset \mathbb{R}^n$)

Zadan je skup $\mathcal{A} = \{a^i \in \mathbb{R}^n: i = 1, \dots, m\}$ s odgovarajućim težinama $w_1, \dots, w_m > 0$. Treba pronaći točku $c^* = (c_1^*, \dots, c_n^*) \in \mathbb{R}^n$, za koju je težinska suma ℓ_1 -udaljenosti do točaka skupa \mathcal{A} minimalna.

Tražena točka c^* točka je u kojoj se postiže globalni minimum funkcije

$$F_1(x) = \sum_{i=1}^m w_i \|x - a^i\|_1 = \sum_{i=1}^m w_i \sum_{s=1}^n |c_s - a_s^i|, \quad (1.14)$$

odnosno točka $c^* \in \operatorname{argmin}_{x \in \mathbb{R}^n} \sum_{i=1}^m w_i \|x - a^i\|_1$ je globalni minimizator funkcije F_1 . U ovom slučaju radi se o problemu globalne optimizacije nediferencijabilne funkcije više varijabli koji uvijek ima rješenje.

Može se provjeriti da se globalni minimum postiže na težinskom medijanu $c^* = \operatorname{med}(w_i, a^i)$ skupa \mathcal{A} (vidi [17, 55]). Što u ovom slučaju možete reći o jedinstvenosti globalnog minimuma?

Zadatak 1.8. Neka je $\mathcal{A} = \{y_1, \dots, y_m\} \subset \mathbb{R}$ skup podataka. Pokažite da je $c^* \in \operatorname{med}_{y_i \in \mathcal{A}} y_i$ točka globalnog minimuma funkcije $F_1: \mathbb{R} \rightarrow \mathbb{R}_+$, $F_1(x) = \sum_{i=1}^m |x - y_i|$.

Zadatak 1.9. Ako su $\varphi_i \in C(\mathbb{R})$, $i = 1, \dots, n$, konveksne funkcije za koje je

$$\hat{c}_i = \operatorname{argmin}_{x \in \mathbb{R}} \varphi_i(x), \quad i = 1, \dots, n,$$

pokažite da je tada i funkcija $f: \mathbb{R}^n \rightarrow \mathbb{R}$,

$$f(x_1, \dots, x_n) = \varphi_1(x_1) + \dots + \varphi_n(x_n),$$

konveksna funkcija za koju vrijedi

$$\operatorname{argmin}_{x \in \mathbb{R}^n} f(x) = (\hat{c}_1, \dots, \hat{c}_n)^T.$$

Primijenite navedenu tvrdnju u svrhu određivanja medijana skupa točaka $\mathcal{A} = \{a^i \in \mathbb{R}^n: i = 1, \dots, m\}$.

Vrijedi li obrat navedene tvrdnje? Primjerice, ako je $f: \mathbb{R}^n \rightarrow \mathbb{R}$ konveksna (pa onda i neprekidna) nediferencijabilna funkcija i ako je $f(x_1, x_2) = \varphi_1(x_1) + \varphi_2(x_2)$, moraju li tada i funkcije φ_1, φ_2 biti konveksne (pa onda i neprekidne) funkcije? Izradite odgovarajuće primjere.

Zadatak 1.10. Odredite medijan i težinski medijan skupa točaka s težinama iz Zadatka 1.7.

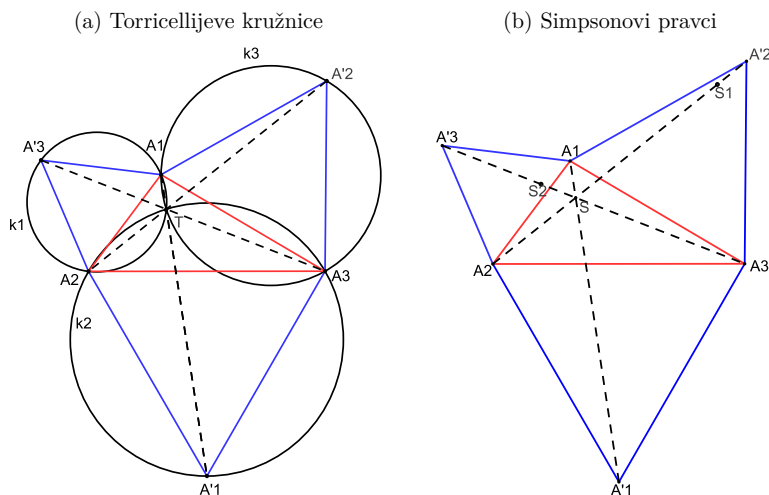
Primjer 1.7. (Geometrijski medijan skupa $\mathcal{A} \subset \mathbb{R}^n$)

Za dani skup $\mathcal{A} = \{a^i \in \mathbb{R}^n: i = 1, \dots, m\}$ s odgovarajućim težinama $w_1, \dots, w_m > 0$ treba pronaći točku $c^* = (c_1^*, \dots, c_n^*) \in \mathbb{R}^n$, za koju je težinska suma euklidskih udaljenosti do točaka skupa \mathcal{A} minimalna.

Tražena točka c^* točka je u kojoj se postiže globalni minimum funkcije

$$F_2(x) = \sum_{i=1}^m w_i d_2(x, a^i) = \sum_{i=1}^m w_i \|x - a^i\|_2 = \sum_{i=1}^m w_i \sqrt{\sum_{s=1}^n (x_s - a_s^i)^2}, \quad (1.15)$$

odnosno točka $c^* \in \operatorname{argmin}_{x \in \mathbb{R}^n} \sum_{i=1}^m w_i \|x - a^i\|_2$ je globalni minimizator funkcije F_2 .

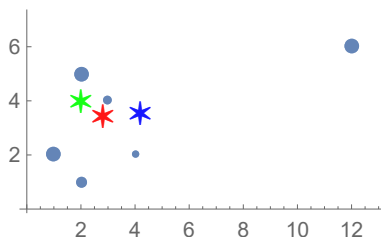


Slika 1.5: Fermat – Torricelli – Weberov problem: geometrijski medijan

Točka c^* zove se težinski geometrijski medijan skupa \mathcal{A} i općenito se ne može eksplicitno izračunati. U literaturi ovaj problem može se naći pod nazivom „Fermat–Torricelli–Weberov problem” (vidi primjerice [11]). Najpoznatiji algoritam za traženje geometrijskog medijana skupa \mathcal{A} poznati je *Weiszfeldov algoritam* iz 1936.

godine [22]. Specijalno, geometrijski medijan triju točaka u ravnini može se geometrijski dobiti [24] na presjeku tzv. Torricellijevih kružnica (vidi Sliku 1.5a) ili na presjeku tzv. Simpsonovih pravaca (vidi Sliku 1.5b).

Primjer 1.8. *Promatrajmo skup podataka $\mathcal{A} = \{a^i = (x_i, y_i) \in \mathbb{R}^2: i = 1, \dots, m\}$ s težinama $w_1, \dots, w_m > 0$. Podaci su prikazani plavim kružićima na Slici 1.6, pri čemu je veličina kružića koji prikazuje podatak a^i određena njegovom težinom w_i . Treba odrediti reprezentant skupa \mathcal{A} kao težinski centroid (1.13), težinski medijan (1.14) i kao težinski geometrijski medijan (1.15). Među podacima uočava se i jedan jako stršeci podatak (outlier) koji je smješten relativno daleko od ostalih podataka.*



Slika 1.6: Reprezentant skupa podataka \mathcal{A} : težinski centroid (plava zvjezdica), težinski medijan (zeleni zvjezdica) i težinski geometrijski medijan (crveni zvjezdica)

Korištenjem niže navedenog *Mathematica*-programa odredit ćemo tražene reprezentante skupa \mathcal{A} .

```
In[1]:= (* Težinski medijan i težinska aritmetička sredina *)
median = Median[WeightedData[podT, podW]]
mean = Mean[WeightedData[podT, podW]]
(* Geometrijski medijan *)
F[x_, y_] := Sum[podW[[i]] Norm[{x, y} - podT[[i]], {i, m}]
min = NMinimize[F[x, y], {x, y}];
GM = {xGM = x /. min[[2]], yGM = y /. min[[2]]}
```

Rezultati su vidljivi na Slici 1.6: težinski centroid podataka označen je plavom zvjezdicom, težinski medijan zelenom, a težinski geometrijski medijan crvenom zvjezdicom. Primijetite da spomenuti outlier ima značajan utjecaj na težinski centroid, dok manji utjecaj ima na težinski medijan i težinski geometrijski medijan. Vizualno, čini se da je najbolji reprezentant skupa podataka \mathcal{A} prikazanih na Slici 1.6 upravo težinski geometrijski medijan (crveni zvjezdica).

Zadatak 1.11. Neka je $\mathcal{A} = \{a^i \in \mathbb{R}^n : i = 1, \dots, m\}$ skup podataka. Problem traženja geometrijskog medijana skupa \mathcal{A} svodi se na rješavanje sljedećeg problema globalne optimizacije

$$\operatorname{argmin}_{x \in \mathbb{R}^n} F_2(x), \quad F_2(x) = \sum_{i=1}^m \|x - a^i\|_2. \quad (1.16)$$

- (a) Na što se svodi rješavanje problema (1.16) u slučaju $n = 1$?
- (b) Napišite funkciju F_2 iz (1.16) za $n = 2$. Za taj slučaj definirajte iterativni postupak (primjenom *Metode jednostavnih iteracija* ili *Newtonove metode* - vidi t.2.1.3, str.37) koji će konvergirati prema geometrijskom medijanu skupa \mathcal{A} .

Zadatak 1.12. Za skup podataka $\mathcal{A} = \{a^i \in \mathbb{R} : i = 1, \dots, m\}$ i za $p > 0$ definirajmo funkciju

$$G_p(x) = \left(\sum_{i=1}^m |x - a^i|^p \right)^{\frac{1}{p}}.$$

Što je minimum funkcije G_p za $p = 1$ i $p = 2$? Kad $p \rightarrow \infty$, G_p u limesu prelazi u funkciju

$$G_\infty(x) = \max_{i=1, \dots, m} |x - a^i|.$$

Pokažite da se minimum funkcije G_∞ postiže u točki $x^* = \frac{1}{2} \left(\min_{i=1, \dots, m} a^i + \max_{i=1, \dots, m} a^i \right)$.

Zadatak 1.13. Za skup podataka $\mathcal{A} = \{a^i \in \mathbb{R} : i = 1, \dots, m\}$ i za $p > 0$ definirajmo funkciju

$$H_p(x) = \sum_{i=1}^m |x - a^i|^p.$$

Što je minimum funkcije H_p za $p = 1$ i $p = 2$? Kad $p \rightarrow 0$, H_p u limesu prelazi u funkciju

$$H_0(x) = \max_{i=1, \dots, m} \delta(x, a^i), \quad \text{gdje je} \quad \delta(x, y) = \begin{cases} 1, & x \neq y, \\ 0, & x = y. \end{cases}$$

Pokažite da se minimum funkcije H_0 postiže za mod skupa \mathcal{A} (podatak s najvećom frekvencijom u skupu \mathcal{A}).

Zadatak 1.14. Pretpostavimo da čekamo jedan od tri lifta koji su raspoređeni duž jednog zida tako da je prvi udaljen od drugog jedan metar a drugi od trećeg tri metra. Dolazak svakog lifta jednako je vjerojatan.

- (a) Gdje treba čekati lift ako želimo da očekivana udaljenost koju ćemo prijeći do lifta bude minimalna? Što bi bio odgovor kada bi drugi i treći lift bili udaljeni 100 metara?

- (b) Gdje treba čekati lift ako želimo da udaljenost koju ćemo prijeći u najgorem slučaju bude minimalna?
- (c) Kako bismo uočili dolazak lifta, za pretpostaviti je da moramo biti na nekoj udaljenosti od zida s vratima liftova, recimo jedan metar. Gdje treba čekati lift u ovom slučaju ako želimo da očekivana udaljenost koju ćemo prijeći do lifta bude minimalna?

Obrazložite kako bi se odgovori promijenili kad bi se razlikovale vjerojatnosti dolazaka liftova.

Primjer 1.9. (*Grupiranje podataka u k klastera*)

Zadan je skup $\mathcal{A} = \{a^i \in \mathbb{R}^n : i = 1, \dots, m\}$, a svakom njegovom elementu a^i pridružena je odgovarajuća težina $w_i > 0$. Skup \mathcal{A} treba grupirati u $1 \leq k \leq m$ nepraznih disjunktih podskupova (klastera) π_1, \dots, π_k . Pri tome elementi unutar nekog klastera trebaju biti što sličniji, a različiti klasteri što bolje razdvojeni. Kažemo da klasteri trebaju biti što kompaktniji i što bolje međusobno razdvojeni.

Internu kompaktnost i međusobnu dobru razdvojenost klastera možemo definirati kao jedan od sljedeća dva problema globalne optimizacije:

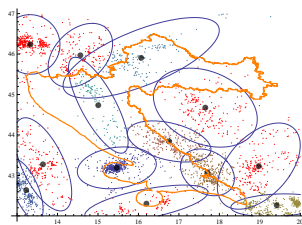
$$(i) \quad \operatorname{argmin}_{\{\pi_1, \dots, \pi_k\}} \mathcal{F}(\{\pi_1, \dots, \pi_k\}), \quad \mathcal{F}(\{\pi_1, \dots, \pi_k\}) = \sum_{j=1}^k \sum_{a^i \in \pi_j} w_i d(c(\pi_j), a^i),$$

gdje su $c(\pi_1), \dots, c(\pi_k)$ centri klastera π_1, \dots, π_k zadani su

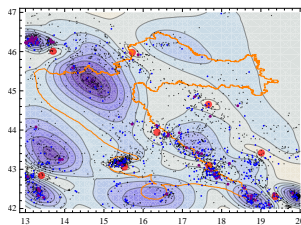
$$c(\pi_j) \in \operatorname{argmin}_{x \in \mathbb{R}^n} \sum_{a^i \in \pi_j} d(x, a^i), \text{ a } d: \mathbb{R}^k \times \mathbb{R}^k \rightarrow \mathbb{R}_+ \text{ neka kvazimetrička funkcija;}$$

$$(ii) \quad \operatorname{argmin}_{c_1, \dots, c_k} F(c_1, \dots, c_k), \quad F(c_1, \dots, c_k) = \sum_{i=1}^m w_i \min_{1 \leq j \leq k} d(c_j, a^i).$$

(a) Geografske pozicije potresa



(b) ContourPlot minimizirajuće funkcije (ii)



Slika 1.7: Seizmološka aktivnost na širem području Republike Hrvatske od 1900. godine – vidi: <http://earthquake.usgs.gov/earthquakes/eqarchives/epic/>.

Na Slici 1.7a prikazane su točke u širem području Republike Hrvatske [43] u kojima se od 1900. godine dogodio potres magnitude ≥ 3 , a na Slici 1.7b prikazan je ContourPlot odgovarajuće minimizirajuće funkcije za spomenute podatke.

Primjedba 1.2. Primijetite da je $F: \mathbb{R}^{kn} \rightarrow \mathbb{R}_+$ općenito nediferencijabilna, nekonveksna simetrična funkcija od $k \times n$ nezavisnih varijabli. Kaže se da je funkcija $f: \mathbb{R}^n \rightarrow \mathbb{R}$ simetrična ako vrijedi $f(x_1, \dots, x_n) = f(x_{i_1}, \dots, x_{i_n})$, gdje je $(x_{i_1}, \dots, x_{i_n})$ proizvoljna permutacija od (x_1, \dots, x_n) . Primjerice, funkcija $f: \mathbb{R}^2 \rightarrow \mathbb{R}$, $f(x_1, x_2) = x_1^2 - 2x_1x_2 + x_2^2$ je simetrična, ali $g: \mathbb{R}^2 \rightarrow \mathbb{R}$, $g(x_1, x_2) = x_1^2 - x_2^2$ nije simetrična funkcija. Ako prethodno navedeni problem globalne optimizacije ima rješenje, onda postoji barem $k!$ različitih točaka globalnog minimuma.

Zadatak 1.15. Pokažite da je funkcija $g: [0, 11] \times [0, 11] \rightarrow \mathbb{R}$, $g(y_1, y_2) = -\frac{1}{5}(y_1^2 + y_2^2) + 2y_1y_2 \cos y_1 \cos y_2$ [16] simetrična. Primjenom programskog sustava *Mathematica* nacrtajte njezin graf i *ContourPlot*. Ispitajte lokalne i globalne ekstreme ove funkcije.

Primjer 1.10. (*Segmentacija crno-bijele slike*)

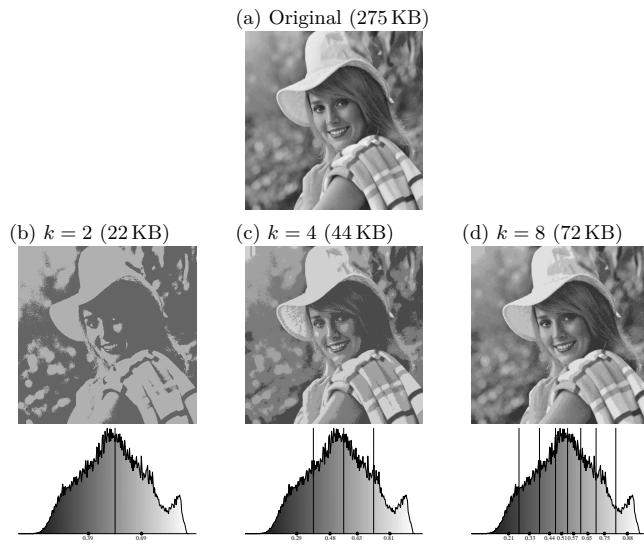
Neka je $\mathcal{A} = \{a^i \in \mathbb{R}: i = 1, \dots, 262144\}$ skup (konačni niz) realnih brojeva koji predstavljaju tonove (gray levels) točaka poznate crno-bijele slike „Elaine” veličine 512×512 (Slika 1.8a).

Svakom podatku $a^i \in \mathcal{A}$ pridružena je težina $w_i = 1$. Skup \mathcal{A} treba grupirati u 2, 4 i 8 klastera. Svakom klasteru pridružiti ćemo njegov centroid, a nakon toga svim točkama klastera gray level koji ima taj centroid. Tako dobivamo rekonstruirane slike s 2, 4 i 8 tonova (vidi Slike 1.8b-d). U podnožju ovih slika prikazani su histogrami tonova originalne slike i vrijednosti odgovarajućih centroida.

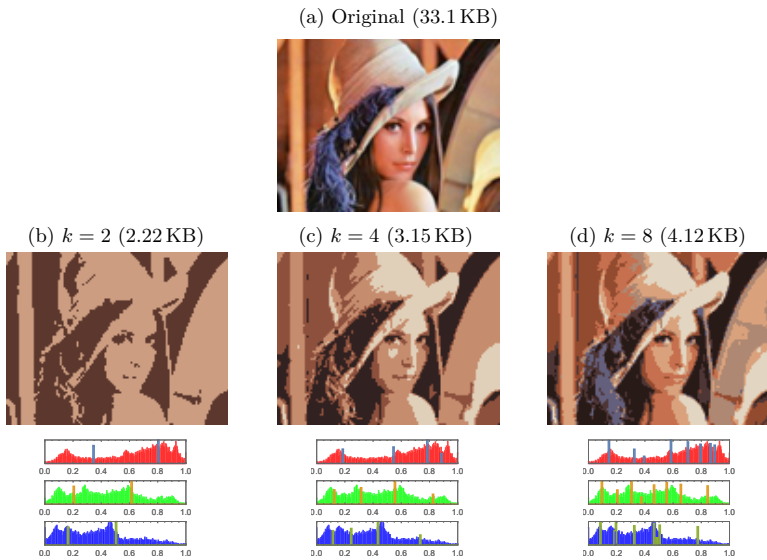
Primjer 1.11. (*Segmentacija slike u boji (RGB zapis)*)

Sliku u boji „Lena” dimenzija 120×120 , zadanu u RGB zapisu treba grupirati u 2, 4 i 8 klastera.

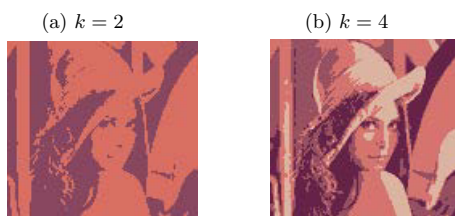
RGB zapis slike podrazumijeva da je svaki piksel a^i predstavljen uređenom trojkom podataka (R^i (crvena-red), G^i (zelena-green), B^i (plava-blue)). Skup $\mathcal{A} = \{a^i \in (R^i, G^i, B^i) \in [0, 1]^3: i = 1, \dots, 14400\}$ sastoji se od podataka s tri atributa s težinama $w_i = 1$. Rezultati dobiveni primjenom LS-kvazimetričke funkcije $d_{LS}(x, y) = \|x - y\|_2^2$ prikazani su na Slici 1.9.



Slika 1.8: Originalna slika i njena segmentacija u 2, 4 ili 8 klastera.



Slika 1.9: Originalna slika i njena segmentacija u 2, 4 ili 8 klastera.



Slika 1.10: Segmentacija “slike u boji” “Lena” dimenzija 120×120 , koja je zadana u HSI zapisu u $k = 2$ i $k = 4$ klastera po kombinaciji Intensity i Hue komponenti.

Zadatak 1.16. Odaberite proizvoljnu sliku u boji.

- (a) Koristeći mogućnosti programskog paketa *Mathematica* odredite dimenzije i RGB zapis odabrane slike.
- (b) Primjenom *Mathematica* naredbe `FindClusters[]` napravite segmentaciju slike u $k = 2, 3$ i 4 klastera, tako da svaki piksel a^i predstavite kao uređenu trojku $a^i = (R^i, G^i, B^i)$.
- (c) Pomoću veze intenziteta piksela i komponenti R, G i B , odredite odgovarajuću monokromatsku (crno-bijelu) sliku. Primjenom *Mathematica* naredbe `FindClusters[]` napravite segmentaciju slike u $k = 2, 3$ i 4 klastera, tako da svaki piksel a^i predstavite s intenzitetom piksela $\frac{1}{3}(R^i + G^i + B^i)$. Usporedite rezultate sa segmentacijama dobivenim pod (b).

Primjer 1.12. (Reprezentant podataka na jediničnoj kružnici)

Zadan je skup realnih brojeva $t_1, \dots, t_m \in [0, 2\pi]$ s odgovarajućim težinama $w_1, \dots, w_m > 0$, pomoću kojih gradimo skup točaka $\mathcal{A} = \{a^i(t_i) = (\cos t_i, \sin t_i) \in \mathbb{R}^2: t_i \in [0, 2\pi]\}$ na jediničnoj kružnici $K = \{(x, y) \in \mathbb{R}^2: x^2 + y^2 = 1\}$. Treba odrediti točku $c^*(t^*) \in K$ koja će što bolje reprezentirati skup točaka \mathcal{A} .

Udaljenost točaka na kružnici može se definirati na sljedeći način

$$d_K(a(t_1), b(t_2)) = \begin{cases} |t_1 - t_2|, & \text{ako } |t_1 - t_2| \leq \pi, \\ 2\pi - |t_1 - t_2|, & \text{ako } |t_1 - t_2| > \pi. \end{cases} \quad (1.17)$$

Zadatak 1.17. Pokažite da je funkcija $d_K: \mathcal{A} \times \mathcal{A} \rightarrow \mathbb{R}_+$ zadana s (1.17) metrika.

Najbolji reprezentant skupa \mathcal{A} s težinama $w_1, \dots, w_m > 0$ u odnosu na metričku funkciju d_K rješenje je problema globalne optimizacije

$$c^*(t^*) = \operatorname{argmin}_{t \in [0, 2\pi]} \sum_{i=1}^m w_i d_K(c(t), a^i(t_i)), \quad (1.18)$$

odnosno $c^*(t^*) \in K$ je točka globalnog minimuma funkcije $F: [0, 2\pi] \rightarrow \mathbb{R}_+$

$$F(t) = \sum_{i=1}^m w_i d_K(c(t), a^i(t_i)). \quad (1.19)$$

Na ovaj način moguće je prikazati i analizirati periodične podatke, kao primjerice (vidi [64]): vodostaj rijeke na nekom mjernom mjestu, temperatura zraka na nekom mjernom mjestu, količina oborina (kiša, snijeg, grad) na nekom mjernom mjestu, itd.

Zadatak 1.18. Na osnovi podataka iz Osijeka od 1900. godine prikažite *Burnov dijagram* [48] temperatura ispod 0°C , Burnov dijagram temperatura iznad 30°C i Burnov dijagram oborina.

Zadatak 1.19. Zadan je skup $\mathcal{A} = \{a^i \in K_1 : i = 1, \dots, m\}$ točaka na jediničnoj ℓ_1 -kružnici $K_1 = \{x \in \mathbb{R}^2 : \|x\|_1 = 1\}$.

- Definirajte udaljenost $d_{K_1}(a, b)$ dviju točaka $a, b \in K_1$ na kružnici K_1 uzimajući u obzir oblik kružnice K_1 .
- Odredite reprezentant c^* skupa točaka \mathcal{A} tako da bude

$$c^* = \operatorname{argmin}_{c \in K_1} \sum_{i=1}^m w_i d_{K_1}(c, a^i),$$

gdje su $w_1, \dots, w_m \in \mathbb{R}_+$ težine podataka.

Isti problem riješite za skup točaka $\mathcal{A} = \{a^i \in K_\infty : i = 1, \dots, m\}$ na jediničnoj ℓ_∞ -kružnici $K_\infty = \{x \in \mathbb{R}^2 : \|x\|_\infty = 1\}$.

Primjer 1.13. (*Segmentacija slike u boji (HSI zapis)*)

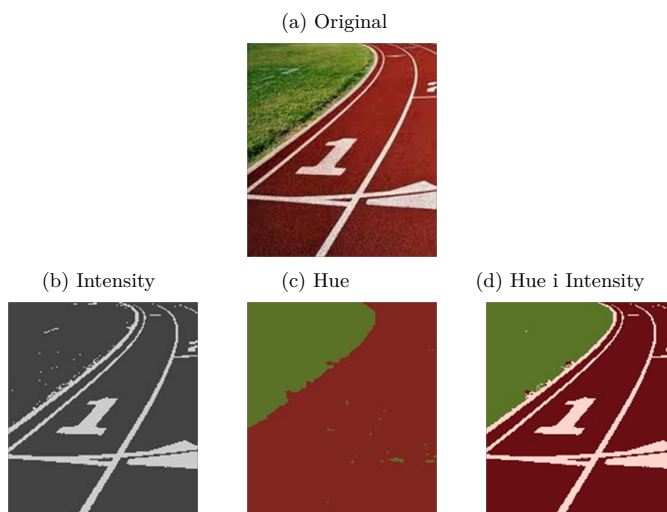
U problemima segmentacije slike u boji se umjesto RGB zapisa u primjenama često koriste i drugi zapisi kao što je primjerice HSI zapis, koji podrazumijeva da je svaki piksel a^i predstavljen uređenom trojkom (H^i (hue), S^i (saturation), I^i (intensity)), pri čemu su $S^i, I^i \in [0, 1]$, a $H^i \in [0, 2\pi]$.

Veza između RGB i HSI zapisa dana je sljedećim relacijama:

$$\begin{bmatrix} Y \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & -\frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}.$$

$$I = Y, \quad S = \sqrt{C_1^2 + C_2^2}, \quad H = \begin{cases} \arccos(\frac{C_2}{S}), & C_1 \geq 0 \\ 2\pi - \arccos(\frac{C_2}{S}), & C_1 < 0. \end{cases}$$

Ponekad je u svrhu segmentacije dovoljno koristiti samo Hue komponentu. Pri tome treba primijetiti da $H_i \in [0, 2\pi]$ ima ciklično svojstvo te je prilikom traženja optimalne particije potrebno koristiti kvazimetričku funkciju iz Primjera 1.12. U



Slika 1.11: Originalna slika i njena segmentacija u dva klastera po Intensity komponenti, Hue komponenti i kombinaciji Intensity i Hue komponente.

cilju dobivanja kvalitetnije segmentacije slike u HSI zapisu dobro je kombinirati Hue i Intensity komponente, što dobro ilustrira primjer prikazan na Slici 1.11. Pri tome je za grupiranje po Intensity komponenti korištena LS-kvazimetrička funkcija $d_{LS}(x, y) = (x - y)^2$. Na Slici 1.10 prikazana je segmentacija “slike u boji” “Lena” dimenzija 120×120 , koja je zadana u HSI zapisu u $k = 2$ i $k = 4$ klastera po kombinaciji Intensity i Hue komponenti.

Primjer 1.14. (Najbolji LS-pravac)

Za dani skup $\mathcal{A} = \{a^i = (x_i, y_i) : i = 1, \dots, m\}$ u ravnini treba odrediti pravac u eksplicitnom obliku za koji se postiže najmanja suma kvadrata vertikalnih odstupanja.

Traženi je pravac najbolji LS-pravac $y = k^*x + l^*$, takav da funkcija

$$F_{LS}(k, l) = \sum_{i=1}^m (y_i - kx_i - l)^2 \tag{1.20}$$

postiže najmanju vrijednost, odnosno točka $(k^*, l^*) \in \operatorname{argmin}_{(k, l) \in \mathbb{R}^2} \sum_{i=1}^m (y_i - kx_i - l)^2$ je globalni minimizator funkcije F_{LS} . Primijetite da se ovdje radi o problemu globalne optimizacije kvadratne funkcije dviju varijabli.

Korištenjem svojstva linearnosti aritmetičke sredine, prema Primjeru 1.5, str.9, vrijedi

$$\sum_{i=1}^m ((y_i - kx_i) - l)^2 \geq \sum_{i=1}^m ((y_i - kx_i) - (\bar{y} - k\bar{x}))^2,$$

gdje za zadani $k \in \mathbb{R}$, broj $\bar{y} - k\bar{x}$ predstavlja aritmetičku sredinu podataka $(y_i - kx_i)$, $i = 1, \dots, m$. Nadalje, rastavljanjem sume na dio u kojemu se pojavljuju apscise podataka x_i koje su jednake \bar{x} i dio u kojemu se ovakve apscise ne pojavljuju te korištenjem ponovo Primjera 1.5, str.9, dobivamo

$$\begin{aligned} \sum_{i=1}^m ((y_i - kx_i) - l)^2 &\geq \sum_{i=1}^m ((y_i - \bar{y}) - k(x_i - \bar{x}))^2 \\ &= \sum_{x_i = \bar{x}} (y_i - \bar{y})^2 + \sum_{x_i \neq \bar{x}} (x_i - \bar{x})^2 \left(\frac{y_i - \bar{y}}{x_i - \bar{x}} - k \right)^2 \\ &\geq \sum_{x_i = \bar{x}} (y_i - \bar{y})^2 + \sum_{x_i \neq \bar{x}} (x_i - \bar{x})^2 \left(\frac{y_i - \bar{y}}{x_i - \bar{x}} - k^* \right)^2, \end{aligned} \quad (1.21)$$

gdje je

$$k^* = \frac{1}{\sum_{x_k \neq \bar{x}} (x_k - \bar{x})^2} \sum_{x_k \neq \bar{x}} (x_k - \bar{x})^2 \frac{y_k - \bar{y}}{x_k - \bar{x}} = \frac{1}{\sum_{x_k \neq \bar{x}} (x_k - \bar{x})^2} \sum_{x_k \neq \bar{x}} (x_k - \bar{x})(y_k - \bar{y}).$$

Optimalna vrijednost koeficijenta l tada je $l^* = \bar{y} - k^*\bar{x}$.

Zadatak 1.20. Riješite prethodni problem uz pretpostavku da je svakom podatku a^i pridružena težina $w_i > 0$. Objasnite značenje formule (1.21).

Primjer 1.15. (Najbolji LAD-pravac)

Za dani skup $\mathcal{A} = \{a^i = (x_i, y_i) : i = 1, \dots, m\}$ s težinama $w_1, \dots, w_m > 0$ treba odrediti pravac u eksplicitnom obliku za koji se postiže najmanja suma apsolutnih vertikalnih odstupanja (Least Absolute Deviations (LAD)).

Traženi je pravac najbolji LAD-pravac $y = k^*x + l^*$, takav da funkcija

$$F_{LAD}(k, l) = \sum_{i=1}^m w_i |y_i - kx_i - l| \quad (1.22)$$

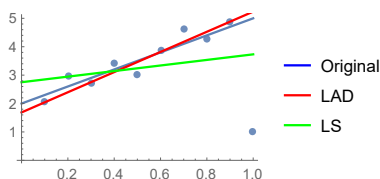
u točki (k^*, l^*) postiže najmanju vrijednost, odnosno točka $(k^*, l^*) = \operatorname{argmin}_{(k, l) \in \mathbb{R}^2} \sum_{i=1}^m w_i |y_i - kx_i - l|$ je globalni minimizator funkcije F_{LAD} . Primijetite da se ovdje radi o problemu globalne optimizacije nediferencijalne funkcije dviju varijabli.

Primjer 1.16. Promatrajmo skup $\mathcal{A} = \{(x_i, y_i) \in \mathbb{R}^2 : i = 1, \dots, m\}$. Podatke ćemo konstruirati na sljedeći način. Neka je $m = 10$, $x_i = i/10$, $y_i = f(x_i) + \varepsilon_i$, gdje je $f(x) = 3x + 2$, a ε_i pseudoslučajan broj generiran

iz normalne distribucije s očekivanjem 0 i varijancom 0.5. U skup \mathcal{A} uvest ćemo jedan jako stršeći podatak (outlier) $y_m = 1$ (vidi Sliku 1.12). Korištenjem niže navedenog Mathematica-programa potražimo najbolji LAD i najbolji LS pravac.

```
In[1]:= Am = 10; SeedRandom[7] ; f[x_] := 3 x + 2
x = Table[i/10., {i, m}];
y = f[x] + Table[Random[NormalDistribution[0, .5]], {i, m}];
y[[m]] = 1;
pod = Table[{x[[i]], y[[i]]}, {i, m}]
slpod = ListPlot[pod, PlotStyle -> {PointSize[.03], Opacity[.9]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
(* Najbolji LAD pravac *)
min = NMinimize[Total[Abs[y - a x - b]], {a, b}];
{a1, b1} = {a /. min[[2]], b /. min[[2]]}
slp1 = Plot[a1 t + b1, {t, 0, x[[m]]}, PlotStyle -> {Red}];
(* Najbolji LS pravac *)
min = NMinimize[Total[Abs[y - a x - b]^2], {a, b}];
{a1, b1} = {a /. min[[2]], b /. min[[2]]}
slp2 = Plot[a1 t + b1, {t, 0, x[[m]]}, PlotStyle -> {Green},
PlotLegends -> LineLegend[{Blue, Red, Green},
{"Original", "LAD", "LS"}]];
s12 = Show[slpod, slp, slp1, slp2, ImageSize -> Small]
```

Rezultati su vidljivi na Slici 1.12. Primijetite da najbolji LAD pravac ignorira outlier i vrlo dobro rekonstruira originalni pravac, dok suprotno tome, outlier jako utječe na najbolji LS pravac.



Slika 1.12: Rekonstrukcija pravca kao najboljeg LS i najboljeg LAD pravca

Zadatak 1.21. Pokažite da se u Primjeru 1.15 ne može postupiti kao u Primjeru 1.14. Obrazložite svoju tvrdnju!

Zadatak 1.22. U radu [55] predložena je „Two Points Method” za rješavanje problema iz Primjera 1.15. Izradite Mathematica-modul za *Two Points Method* i testirajte ga na nekoliko primjera.

Primjer 1.17. (Najbolji TLS-pravac)

Za dani skup $\mathcal{A} = \{T_i = (x_i, y_i) \in \mathbb{R}^2: i = 1, \dots, m\}$ s odgovarajućim

težinama $w_1, \dots, w_m > 0$ treba odrediti pravac p u ravnini tako da suma kvadrata euklidskih ℓ_2 udaljenosti točaka T_i do pravca p bude minimalna.

Općenito, pravac p je oblika $ax + by + c = 0$, $a^2 + b^2 \neq 0$, a euklidska udaljenost točke $T_i = (x_i, y_i)$ do pravca p zadana je s (vidi primjerice [30, 31])

$$d_2(T_i, p) = \frac{|ax_i + by_i + c|}{\sqrt{a^2 + b^2}}.$$

Pravac za koji je suma kvadrata euklidskih udaljenosti minimalna u literaturi se naziva „the best Total Least Squares Line” (TLS), a dobiva se kao rješenje sljedećeg nelinearnog problema globalne optimizacije:

$$\operatorname{argmin}_p \sum_{i=1}^m w_i d_{LS}(T_i, p) = \operatorname{argmin}_{a,b,c \in \mathbb{R}} \sum_{i=1}^m w_i \frac{(ax_i + by_i + c)^2}{a^2 + b^2}.$$

Opći pravac u ravnini možemo zapisati s

$$ax + by + c = 0, \quad a^2 + b^2 = 1,$$

pa se prethodni problem transformira na sljedeći problem uvjetnog ekstrema:

$$\operatorname{argmin}_{a^2+b^2=1} \sum_{i=1}^m w_i (ax_i + by_i + c)^2.$$

Osim toga, lako je pokazati da najbolji TLS-pravac mora proći centroidom podataka $c^* = (x_p, y_p)$ (vidi Zadatak 1.23), pa prethodni problem možemo još pojednostaviti

$$\operatorname{argmin}_{a^2+b^2=1} \sum_{i=1}^m w_i (a(x_i - x_p) + b(y_i - y_p))^2.$$

Ovdje se radi o problemu globalne optimizacije nelinearne, ali diferencijalne funkcije dviju varijabli s ograničenjima. U radu [45] ovaj problem rješava se traženjem svojstvenog vektora koji odgovara manjoj svojstvenoj vrijednosti odgovarajuće kovarijacijske matrice (vidi također [74]).

Zadatak 1.23. Pokažite da najbolji TLS pravac prolazi centroidom skupa podataka.

Primjer 1.18. (Najbolji OD-pravac)

Za dani skup podataka $\mathcal{A} = \{a^i = (x_i, y_i) \in \mathbb{R}^2 : i = 1, \dots, m\}$ s odgovarajućim težinama $w_1, \dots, w_m > 0$ treba odrediti pravac p u ravnini tako da suma euklidskih ℓ_2 udaljenosti točaka T_i do pravca p bude minimalna.

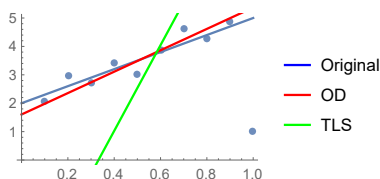
Pravac p je oblika $ax + by + c = 0$, $a^2 + b^2 = 1$. U literaturi se takav pravac naziva „the best Orthogonal Distance line” (OD), a rješenje je sljedećeg nelinearnog problema globalne optimizacije:

$$\begin{aligned} \operatorname{argmin}_p \sum_{i=1}^m w_i d_2(T_i, p) &= \operatorname{argmin}_{a,b,c \in \mathbb{R}} \sum_{i=1}^m w_i \frac{|ax_i + by_i + c|}{\sqrt{a^2 + b^2}} \\ &= \operatorname{argmin}_{a^2+b^2=1} \sum_{i=1}^m w_i |ax_i + by_i + c|. \end{aligned}$$

Primijetite da se ovdje radi o problemu globalne optimizacije nelinearne nediferencijabilne funkcije triju varijabli s ograničenjima koji se općenito ne može eksplicitno riješiti.

Primjer 1.19. Za podatke iz Primjera 1.16 potražiti ćemo najbolji TLS i najbolji OD pravac. Pri tome ćemo koristiti niže navedeni Mathematica-program.

```
In[1]:= (* Najbolji OD pravac *)
min = NMinimize[Total[Abs[k x + 1 - y]/Sqrt[k^2 + 1]], {k, 1}];
{a1, b1} = {k /. min[[2]], 1 /. min[[2]]}
slp1 = Plot[a1 t + b1, {t, 0, x[[m]]}, PlotStyle -> {Red}];
(* Najbolji TLS pravac *)
min = NMinimize[Total[Abs[k x + 1 - y]^2/(k^2 + 1)], {k, 1}];
{a1, b1} = {k /. min[[2]], 1 /. min[[2]]}
slp2 = Plot[a1 t + b1, {t, 0, x[[m]]}, PlotStyle -> {Green},
PlotLegends -> LineLegend[{Blue, Red, Green},
{"Original", "OD", "TLS"}];
Print[Show[slpod, slp, slp1, slp2, ImageSize -> Small]]
```



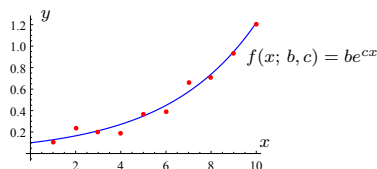
Slika 1.13: Rekonstrukcija pravca kao najboljeg OD i najboljeg TLS pravca

Rezultati su vidljivi na Slici 1.13. Primijetite da najbolji OD pravac ignorira outlier i vrlo dobro rekonstruira originalni pravac, dok suprotno tome, outlier jako utječe na najbolji TLS pravac.

Primjer 1.20. (Nelinearni problem najmanjih kvadrata)

Za dani skup $\mathcal{A} = \{a^i = (x_i, y_i) : i = 1, \dots, m\}$ treba odrediti optimalne

parametre b^* i c^* eksponencijalne model-funkcije $f(t; b, c) = be^{ct}$, tako da suma kvadrata izmjerenih od teoretskih vrijednosti bude minimalna (vidi Sliku 1.14).



Slika 1.14: Podaci i najbolja LS-eksponencijalna model-funkcija

LS-optimalni parametri b^* i c^* eksponencijalne model-funkcije $f(t; b, c) = be^{ct}$ mogu ali i ne moraju postojati (vidi [29]). Ako postoje, odredit ćemo ih tako da potražimo globalni minimizator funkcije

$$F(b, c) = \sum_{i=1}^m (be^{cx_i} - y_i)^2,$$

tj. $(b^*, c^*) \in \underset{(b, c) \in \mathbb{R}^2}{\operatorname{argmin}} F(b, c)$. Primijetite da se u ovom slučaju radi o problemu minimizacije „glatke“ (višestruko derivabilne) funkcije dviju varijabli.

Kao što smo to uradili u slučaju traženja najboljeg pravca, i u slučaju traženja „najbolje eksponencijalne funkcije“ možemo primijeniti neki drugi pristup.

Brojne druge primjere optimizacijskih problema iz praktičnih istraživanja možemo naći kod [1, 2, 37, 52].

1.3 Konveksne funkcije

Za funkciju $f: \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$, promatrajmo optimizacijski problem definiran u t.1, str.1. U nastavku navodimo definicije konveksnog skupa i konveksne funkcije te dajemo odgovarajuće karakterizacije (vidi primjerice [7, 20, 28]).

Definicija 1.3. Kažemo da je skup $\mathcal{D} \subseteq \mathbb{R}^n$ konveksan ako je za sve $x, y \in \mathcal{D}$, $\alpha x + (1 - \alpha)y \in \mathcal{D}$ za svaki $\alpha \in [0, 1]$. Za funkciju $f: \mathcal{D} \rightarrow \mathbb{R}$, definiranu na konveksnom skupu $\mathcal{D} \subseteq \mathbb{R}^n$ kažemo da je konveksna ako za sve $x, y \in \mathcal{D}$ vrijedi

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y), \quad \forall \alpha \in [0, 1]. \quad (1.23)$$

Ako u prethodnoj nejednakosti vrijedi znak „<“ za svaki $\alpha \in (0, 1)$ i $x \neq y$, onda kažemo da je funkcija f strogo konveksna.

U sljedećim lemapa navedena su neka važna svojstva konveksnih funkcija [3, 7, 20, 25].

Lema 1.1. *Neka je $f: \mathcal{D} \rightarrow \mathbb{R}$, konveksna funkcija definirana na konveksnom skupu $\mathcal{D} \subseteq \mathbb{R}^n$. Onda vrijedi:*

(i) *nivo-skup (level-set) $\mathcal{D}_\lambda(f) = \{x \in \mathcal{D}: f(x) \leq \lambda\}$ je konveksan skup za svaki $\lambda \in \mathbb{R}$;*

(ii) *ako je \mathcal{D} otvoren skup, onda je funkcija f neprekidna na \mathcal{D} .*

Lema 1.2. *Neka je $f: [a, b] \rightarrow \mathbb{R}$ konveksna derivabilna funkcija. Tada:*

(i) $f(u) \geq f(v) + f'(v)(u - v)$, $\forall u, v \in [a, b]$ (gradijentna nejednakost)

(ii) *Derivacija f' ne opada na $[a, b]$.*

Dokaz. U svrhu dokaza tvrdnje (i) nejednakost (1.23) zapišimo u obliku

$$\alpha f(u) \geq f(v + \alpha(u - v)) - (1 - \alpha)f(v),$$

što nakon dijeljenja s α možemo zapisati kao:

$$f(u) \geq f(v) + \frac{f(v + \alpha(u - v)) - f(v)}{\alpha(u - v)}(u - v).$$

Za $\alpha \rightarrow 0+$ dobivamo nejednakost (i).

Dokažimo tvrdnju (ii). U gradijentnoj nejednakosti (i) varijable u i v su ravnopravne. Zamijenimo li ih, iz (i) dobivamo $f(v) \geq f(u) + f'(u)(v - u)$. Zbrajanjem ove nejednakosti s gradijentnom nejednakosti (i) dobivamo

$$(f'(u) - f'(v))(u - v) \geq 0,$$

iz čega slijedi tražena tvrdnja (vidi također Zadatak 1.24). \square

Zadatak 1.24. Pokažite da je funkcija $f: [a, b] \rightarrow \mathbb{R}$ rastuća onda i samo onda ako vrijedi

$$(x_2 - x_1)(f(x_2) - f(x_1)) \geq 0, \quad \forall x_1, x_2 \in [a, b], x_1 < x_2,$$

a da je padajuća onda i samo onda ako vrijedi

$$(x_2 - x_1)(f(x_2) - f(x_1)) \leq 0, \quad \forall x_1, x_2 \in [a, b], x_1 < x_2.$$

Tvrdnje sljedećih zadataka u sebi sadrže nekoliko kriterija konveksnosti, ovisno o pretpostavkama koje funkcija zadovoljava.

Zadatak 1.25. Neka je $f: \mathcal{D} \rightarrow \mathbb{R}$ konveksna derivabilna funkcija definirana na otvorenom konveksnom skupu $\mathcal{D} \subseteq \mathbb{R}$. Dokažite da ako je $x^* \in \mathcal{D}$ takav da je $f'(x^*) = 0$, tada je x^* globalni minimizator funkcije f .

Zadatak 1.26. Neka je $f: \mathcal{D} \rightarrow \mathbb{R}$ dva puta derivabilna funkcija definirana na otvorenom konveksnom skupu $\mathcal{D} \subseteq \mathbb{R}$. Dokažite:

- (a) Funkcija f je konveksna ako i samo ako je $f''(x) \geq 0$ za svaki $x \in \mathcal{D}$.
- (b) Ako je $f''(x) > 0$ za svaki $x \in \mathcal{D}$, onda je f strogo konveksna.
- (c) Na primjeru funkcije $f: \mathbb{R} \rightarrow \mathbb{R}$, $f(x) = x^4$ pokažite da obrat tvrdnje pod (b) ne vrijedi.

Konveksnost minimizirajuće funkcije značajno pojednostavljuje optimizacijski problem zbog činjenice da lokalna optimalnost rješenja garantira globalnu optimalnost o čemu govori sljedeća lema.

Lema 1.3. *Neka je $f: \mathcal{D} \rightarrow \mathbb{R}$ konveksna funkcija definirana na konveksnom skupu $\mathcal{D} \subseteq \mathbb{R}^n$. Ako je x^* lokalni minimizator funkcije f , tada je x^* i globalni minimizator funkcije f .*

Dokaz. Ako je x^* lokalni minimizator funkcije f na \mathcal{D} , onda postoji okolina $U(x^*) \subseteq \mathcal{D}$ od x^* takva da je $f(z) \geq f(x^*)$ za sve $z \in U(x^*)$. Za proizvoljnu točku $y \in \mathcal{D}$ možemo odabrati $\alpha < 1$ dovoljno blizu 1 tako da je $z = \alpha x^* + (1 - \alpha)y \in U(x^*)$. Zbog konveksnosti od f

$$\alpha f(x^*) + (1 - \alpha)f(y) \geq f(z) \geq f(x^*),$$

iz čega slijedi $f(y) \geq f(x^*)$. Kako to vrijedi za proizvoljni $y \in \mathcal{D}$, x^* je globalni minimizator. \square

Zadatak 1.27. Pokažite da je funkcija $F_1: \mathbb{R}^2 \rightarrow \mathbb{R}$ definirana u Primjeru 1.15 konveksna.

Zadatak 1.28. Dokažite da je konveksna funkcija $f: [a, b] \rightarrow \mathbb{R}$ neprekidna u svim unutrašnjim točkama intervala $[a, b]$ i da ima konačne derivacije slijeva i sdesna, pri čemu vrijedi $f'(x-) \leq f'(x+)$, $x \in (a, b)$.

Uputa: Dokazati da vrijedi

$$\frac{f(u) - f(u - \tau)}{\tau} \leq \frac{f(u) - f(u - h)}{h} \leq \frac{f(u + h) - f(u)}{h} \leq \frac{f(u + \tau) - f(u)}{\tau}, \quad (1.24)$$

gdje je $0 < h < \tau$; $u, u \pm \tau, u \pm h \in [a, b]$.

Zadatak 1.29. Pokažite da za konveksnu funkciju $f: [a, b] \rightarrow \mathbb{R}$ vrijedi

$$\frac{f(u) - f(v)}{u - v} \leq \frac{f(w) - f(v)}{w - v} \leq \frac{f(w) - f(u)}{w - u}, \quad a \leq v < u < w \leq b. \quad (1.25)$$

Koji je geometrijski smisao ovih nejednakosti?

Uputa. Iskoristiti konveksnost funkcije f i zapis $u = \alpha v + (1 - \alpha)w$, gdje je $v < u < w$, $\alpha = \frac{w-u}{w-v}$, $\alpha \in (0, 1)$.

Zadatak 1.30. Dokažite da ako je $f : [a, b] \rightarrow \mathbb{R}$ konveksna funkcija, onda vrijedi

$$f(u) \geq f(v) + \gamma_n(u - v), \quad \forall u \in [a, b], \forall v \in (a, b), \quad (1.26)$$

gdje je $\gamma_n \in \mathbb{R}$ proizvoljni broj takav da je $f'(v-) \leq \gamma_n \leq f'(v+)$.

Uputa. Iskoristiti nejednakost (1.24) razmotrivši slučajeve $u > v$ i $u < v$ i nejednakost $f'(v-) \leq f'(v+)$.

Zadatak 1.31. Dokažite da je $x^* \in [a, b]$ točka minimuma konveksne funkcije $f : [a, b] \rightarrow \mathbb{R}$ onda i samo onda ako je $f'(x^*+) \geq 0$ i $f'(x^*-) \leq 0$. Specijalno, ako $f'(x^*)$ postoji i ako je $x^* \in (a, b)$, onda je $f'(x^*) = 0$.

Uputa. U (1.26) stavite $\gamma_n = 0$.

Zadatak 1.32. Dokažite da je derivabilna funkcija $f : [a, b] \rightarrow \mathbb{R}$ konveksna onda i samo onda ako njezina derivacija f' ne opada na $[a, b]$ (vidi Lemu 1.2).

Zadatak 1.33. Dokažite da ako je funkcija $f : [a, b] \rightarrow \mathbb{R}$ monotono rastuća, a $z : [c, d] \rightarrow [a, b]$ konveksna, onda je $f \circ z : [c, d] \rightarrow \mathbb{R}$ konveksna.

1.4 Kvazikonveksne funkcije

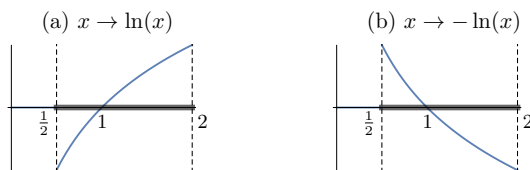
Kao što smo vidjeli iz Leme 1.3, konveksne funkcije imaju svojstvo izrazito korisno u optimizaciji, a koje glasi da je svaki lokalni minimizator funkcije (ako postoji) ujedno i globalni minimizator te funkcije. U mnogim optimizacijskim problemima pojavljuju se funkcije koje imaju slično svojstvo, a pri tome nisu konveksne. Prirodno je postaviti pitanje može li se oslabjeti zahtjev na konveksnost funkcije, a da se zadrži neki oblik ovog svojstva. S tim ciljem uvodimo sljedeću definiciju.

Definicija 1.4. Funkcija $f : \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$ je kvazikonveksna na konveksnom skupu \mathcal{D} ako za sve $x, y \in \mathcal{D}$ vrijedi

$$f(\alpha x + (1 - \alpha)y) \leq \max\{f(x), f(y)\}, \quad \forall \alpha \in [0, 1]. \quad (1.27)$$

Ako u (1.27) vrijedi stroga nejednakost za $\alpha \in (0, 1)$ i $x \neq y$, onda kažemo da je f strogo kvazikonveksna na \mathcal{D} (vidi [46]).

Primjedba 1.3. Primijetite da je svaka monotona funkcija ujedno i kvazikonveksna. Kao primjer razmotrite monotono rastuću funkciju $x \rightarrow \ln(x)$ i monotono padajuću funkciju $x \rightarrow -\ln(x)$, obje definirane na segmentu $[0.5, 2]$ (vidi Sliku 1.15).



Slika 1.15: Kvizikonveksne funkcije

Sljedeća lema daje jedan praktičan kriterij za ispitivanje kvazikonveksnosti neke funkcije.

Lema 1.4. *Funkcija $f: \mathcal{D} \rightarrow \mathbb{R}$ je kvazikonveksna na \mathcal{D} onda i samo onda ako je za svaki $\lambda \in \mathbb{R}$ nivo-skup $\mathcal{D}_\lambda = \{x \in \mathcal{D} : f(x) \leq \lambda\}$ konveksan.*

Dokaz. (Nužnost) Pretpostavimo da je funkcija $f: \mathcal{D} \rightarrow \mathbb{R}$ kvazikonveksna i dokažimo da je za proizvoljni $\lambda \in \mathbb{R}$ skup \mathcal{D}_λ konveksan.

Neka su $x, y \in \mathcal{D}_\lambda$. To znači da je $f(x) \leq \lambda$ i $f(y) \leq \lambda$. Kako je f kvazikonveksna, za proizvoljni $\alpha \in [0, 1]$ vrijedi

$$f(\alpha x + (1 - \alpha)y) \leq \max\{f(x), f(y)\} \leq \lambda,$$

iz čega zaključujemo da je $\alpha x + (1 - \alpha)y \in \mathcal{D}_\lambda$.

(Dovoljnost) Pretpostavimo da je za proizvoljni $\lambda \in \mathbb{R}$ skup \mathcal{D}_λ konveksan i dokažimo da je tada funkcija $f: \mathcal{D} \rightarrow \mathbb{R}$ kvazikonveksna.

Za proizvoljne $x, y \in \mathcal{D}$ definirajmo $\lambda_0 = \max\{f(x), f(y)\}$. Tada su $x, y \in \mathcal{D}_{\lambda_0}$, a kako je \mathcal{D}_{λ_0} konveksan, onda je $\alpha x + (1 - \alpha)y \in \mathcal{D}_{\lambda_0}$, $\forall \alpha \in [0, 1]$, iz čega slijedi

$$f(\alpha x + (1 - \alpha)y) \leq \lambda_0 = \max\{f(x), f(y)\}. \quad \square$$

Primjedba 1.4. Primijetite da vrijedi [3, 46]:

- (i) Svaka konveksna funkcija f je kvazikonveksna, ali obrat ne vrijedi. S druge strane, kvazikonveksna funkcija ne mora biti konveksna. Primjerice, na osnovu Leme 1.1, nije teško vidjeti da su funkcije $f: (0, \infty) \rightarrow \mathbb{R}$ zadane s $f(x) = \ln x$ te $g: [-1, 1] \rightarrow \mathbb{R}$, $g(x) = \sqrt[3]{x^2}$ kvazikonveksne, ali nisu konveksne.
- (ii) Ako su funkcije f_i , $i = 1, \dots, m$, kvazikonveksne na \mathcal{D} , onda je i funkcija $f(x) = \max_{i=1, \dots, m} f_i(x)$ također kvazikonveksna na \mathcal{D} .
- (iii) Kvizikonveksna funkcija ne mora biti neprekidna. Primjerice, funkcija $f: \mathbb{R} \rightarrow \mathbb{R}$, $f(x) = \text{sign } x$ je kvazikonveksna na \mathbb{R} , ali nije neprekidna. Primijetite također da ova funkcija nije strogo kvazikonveksna.

Primjer 1.21. Promatramo sljedeće funkcije (vidi Sliku 1.16)

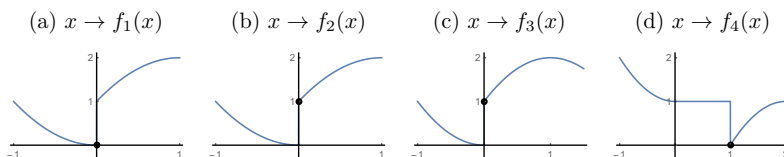
$$f_1(x) = \begin{cases} x^2, & x \in [-1, 0] \\ 2 - (x - 1)^2, & x \in (0, 1] \end{cases}, \quad f_2(x) = \begin{cases} x^2, & x \in [-1, 0) \\ 2 - (x - 1)^2, & x \in [0, 1] \end{cases},$$

$$f_3(x) = \begin{cases} x^2, & x \in [-1, 0] \\ 2 - (x - 1)^2, & x \in (0, 1.5] \end{cases}, \quad f_4(x) = \begin{cases} x^2 + 1, & x \in [-1, 0] \\ 1, & x \in (0, 1) \\ 1 - (x - 2)^2, & x \in [1, 2] \end{cases}.$$

Funkcija f_1 je strogo kvazikonveksna funkcija za koju je $\operatorname{argmin}_{x \in [-1, 1]} f_1(x) = \{0\}$.

Funkcija f_2 je strogo kvazikonveksna funkcija za koju vrijedi $\operatorname{argmin}_{x \in [-1, 1]} f_2(x) = \emptyset$.

Funkcija f_3 nije kvazikonveksna funkcija, dok je f_4 kvazikonveksna, ali ne i strogo kvazikonveksna funkcija. Nadalje, lokalni minimizator funkcije f_4 svaka je točka iz intervala $[0, 1)$, ali niti jedna od tih točaka nije globalni minimizator. Njezin globalni minimizator je $\operatorname{argmin}_{x \in [-1, 2]} f_4(x) = x^* = 1$, a globalni minimum iznosi $f_4(x^*) = 0$.



Slika 1.16: Kvazikonveksne funkcije

Primjer funkcije f_4 pokazuje kako se može dogoditi da lokalni minimizator kvazikonveksne funkcije nije ujedno globalni minimizator, odnosno da tvrdnja iz Leme 1.3 ne vrijedi općenito za kvazikonveksne funkcije. Međutim, analogna tvrdnja može se dokazati uz nešto jaču pretpostavku na skup lokalnih minimizatora funkcije f .

Lema 1.5. Neka je $f: \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$ kvazikonveksna funkcija definirana na konveksnom skupu \mathcal{D} . Ako je x^* strogi lokalni minimizator funkcije f , tada je x^* strogi globalni minimizator funkcije f .

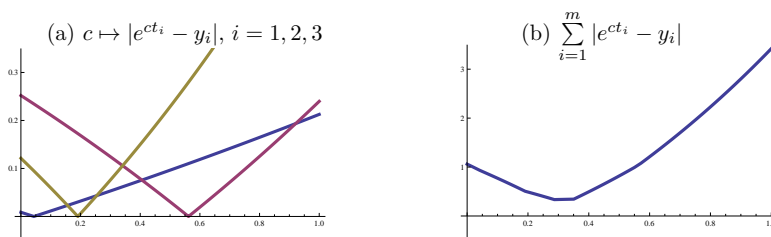
Dokaz. Ako je x^* strogi lokalni minimizator funkcije f na skupu \mathcal{D} , onda postoji otvoreni skup $U(x^*)$ oko točke x^* takav da je $f(x^*) < f(x)$, za svaki $x \in U(x^*) \cap \mathcal{D}$ i $x \neq x^*$. Pretpostavimo da x^* nije strogi globalni minimizator od f na \mathcal{D} , odnosno da postoji $\bar{x} \in \mathcal{D}$, $\bar{x} \neq x^*$ i $f(\bar{x}) < f(x^*)$. Zbog kvazikonveksnosti funkcije f vrijedi

$$f(\lambda \bar{x} + (1 - \lambda)x^*) \leq \max\{f(\bar{x}), f(x^*)\} = f(x^*)$$

za svaki $\lambda \in [0, 1]$. Za dovoljno maleni $\lambda \in [0, 1]$ dobivamo $\lambda\bar{x} + (1 - \lambda)x^* \in \mathcal{D} \cap U(x^*)$, što je u kontradikciji s pretpostavkom da je x^* strogi lokalni mimimizator funkcije f . \square

Zadatak 1.34. Je li zbroj dviju kvazikonveksnih funkcija također kvazikonveksna funkcija? Za podatke (t_i, y_i) , $i = 1, \dots, m$, funkcija $c \mapsto |e^{ct_i} - y_i|$ je kvazikonveksna. Je li i sljedeća funkcija kvazikonveksna (vidi Sliku 1.17):

$$F(c) = \sum_{i=1}^m |e^{ct_i} - y_i|?$$



Slika 1.17: Funkcija $F(c) = \sum_{i=1}^m |e^{ct_i} - y_i|$

Zadatak 1.35. Proverite koje su od ovih funkcija konveksne, a koje kvazikonveksne:

- (a) $f_1 : [a, b] \rightarrow \mathbb{R}, f_1(x) = x^2$,
- (b) $f_2 : [a, b] \rightarrow \mathbb{R}, f_2(x) = |x| + x + \text{sign } x$,
- (c) $f_3 : [1, 2] \rightarrow \mathbb{R}, f_3(x) = \ln x$,
- (d) $g_1 : [a, b] \rightarrow \mathbb{R}, g_1(x) = \begin{cases} |x|, & x \neq 0 \\ -1, & x = 0 \end{cases}$,
- (e) $g_2 : [a, b] \rightarrow \mathbb{R}, g_2(x) = \begin{cases} |x|, & x \neq 0 \\ 1, & x = 0 \end{cases}$,
- (f) $g_3 : [0, 2] \rightarrow \mathbb{R}, g_3(x) = \begin{cases} x^2, & x \in [0, 1] \\ \sqrt{x}, & x \in [1, 2] \end{cases}$.

Zadatak 1.36. Proverite svojstvo konveksnosti i kvazikonveksnosti sljedećih funkcija:

- (a) $f_1 : [a, b] \rightarrow \mathbb{R}, f_1(x) = x^4 + 3x$,
- (b) $f_2 : (0, \infty) \rightarrow \mathbb{R}, f_2(x) = -e^{-x}$,
- (c) $f_3 : [0, 3] \rightarrow \mathbb{R}, f_3(x) = \frac{-1}{x^2 + 1}$.

Zadatak 1.37. Kako treba proširiti funkciju $f(x) = \frac{1}{e^{1/x^2} + 1}$ u točki $x_0 = 0$ tako da ona bude

- (a) strogo kvazikonveksna na $[0, 1]$,
- (b) strogo kvazikonveksna na $[-1, 1]$,
- (c) strogo kvazikonveksna na $[-1, 0]$?

Zadatak 1.38. Kako treba proširiti funkciju

$$f(x) = \begin{cases} x, & x > 0 \\ -x + A, & x < 0 \end{cases}$$

u točki $x_0 = 0$ tako da ona bude

- (a) strogo kvazikonveksna na $[0, 1]$,
- (b) strogo kvazikonveksna na $[-1, 0]$,
- (c) strogo kvazikonveksna na $[-1, 1]$,
- (d) strogo kvazikonveksna na $[a, b]$ za proizvoljne $a, b \in \mathbb{R}$?

Razmotrite slučajeve: $A = 0, +1, -1$.

Zadatak 1.39. Ako su funkcije f_1, f_2 strogo kvazikonveksne na $[a, b]$, jesu li takve i funkcije $f_1 + f_2, f_1 - f_2, f_1 \cdot f_2$ te f_1/f_2 pod pretpostavkom da su dobro definirane?

Zadatak 1.40. Neka je $f: \mathcal{D} \rightarrow \mathbb{R}, \mathcal{D} \subseteq \mathbb{R}$ derivabilna funkcija na otvorenom konveksnom skupu \mathcal{D} . Pokažite da je tada f kvazikonveksna na \mathcal{D} ako i samo ako za svaki $x, y \in \mathcal{D}$ vrijedi

$$f(y) \leq f(x) \Rightarrow f'(x)(y - x) \leq 0.$$

Zadatak 1.41. Neka je $f: \mathbb{R} \rightarrow \mathbb{R}$ dva puta derivabilna funkcija. Dokažite da je f kvazikonveksna ako i samo ako za svaki $x \in \mathbb{R}$

$$f'(x) = 0 \Rightarrow f''(x) \geq 0.$$

Zadatak 1.42. Provjerite je li funkcija $f: (0, \infty)^2 \rightarrow \mathbb{R}, f(x_1, x_2) = -x_1 x_2$ kvazikonveksna.

Zadatak 1.43. Neka je funkcija $f: \mathbb{R}^n \rightarrow \mathbb{R}$ definirana s $F(x) = \max\{i: x_i \neq 0\}$ i $f(\mathbf{0}) = 0$. Pokažite da je f kvazikonveksna.

Zadatak 1.44. Provjerite jesu li sljedeće funkcije kvazikonveksne.

- a) $f: \mathcal{D} \rightarrow \mathbb{R}, \mathcal{D} = \{x \in \mathbb{R}^n: c^T x + d > 0\}$ zadana s

$$f(x) = \frac{a^T x + b}{c^T x + d},$$

gdje su $a, b, c, d \in \mathbb{R}^n$.

- b) $f: \mathcal{D} \rightarrow \mathbb{R}, \mathcal{D} = \{x \in \mathbb{R}^n: \|x - a\|_2 \leq \|x - b\|_2\}$ zadana s

$$f(x) = \frac{\|x - a\|_2}{\|x - b\|_2},$$

gdje su $a, b \in \mathbb{R}^n$.

1.4.1 Unimodalne funkcije

U slučaju funkcija jedne varijable pojam „strove kvazikonveksnosti” u literaturi se može pronaći i pod nazivom „unimodalne funkcije” (vidi primjerice [25, 46]).

Definicija 1.5. [25] *Kažemo da je funkcija $f : [a, b] \rightarrow \mathbb{R}$ unimodalna ako postoji $x^* \in [a, b]$ takav da je $f(x^*) = \min_{x \in [a, b]} f(x)$, a za proizvoljne $a \leq x_1 < x_2 \leq b$ vrijedi*

$$\begin{aligned} x_2 \leq x^* &\implies f(x_1) > f(x_2), \\ x^* \leq x_1 &\implies f(x_2) > f(x_1). \end{aligned}$$

Primjedba 1.5. [25]

- (i) Uočimo da svojstvo unimodalnosti povlači da je funkcija f strogo padajuća na $[a, x^*]$ i strogo rastuća na $[x^*, b]$ dok u točki x^* postiže globalni minimum.
- (ii) Svaka je strogo monotono rastuća (padajuća) funkcija $f : [a, b] \rightarrow \mathbb{R}$ unimodalna.
- (iii) Strogo konveksna funkcija je unimodalna.
- (iv) Unimodalna funkcija ne mora biti neprekidna.
- (v) Ako je $f : [a, b] \rightarrow \mathbb{R}$ unimodalna funkcija i $[a_1, b_1] \subset [a, b]$, onda je $f|_{[a_1, b_1]}$ unimodalna funkcija na $[a_1, b_1]$.
- (vi) Ako je $f : [a, b] \rightarrow \mathbb{R}$ unimodalna funkcija, $x^* = \operatorname{argmin}_{x \in [a, b]} f(x)$ i ako je $a < x_1 < x_2 < b$, onda vrijedi

$$\begin{aligned} f(x_1) \leq f(x_2) &\implies x^* \in [a, x_2], \\ f(x_1) > f(x_2) &\implies x^* \in [x_1, b]. \end{aligned}$$

Zadatak 1.45. Neka za funkciju $f : [a, b] \rightarrow \mathbb{R}$ postoji $x^* = \operatorname{argmin}_{x \in [a, b]} f(x)$. Pokažite da je tada funkcija f strogo kvazikonveksna na $[a, b]$ onda i samo onda ako je unimodalna.

1.5 Diferencijabilne funkcije

Ako se može pretpostaviti da je promatrana funkcija diferencijabilna, pristup rješavanju optimizacijskog problema može se preciznije postaviti. Svojstvo diferencijabilnosti funkcije ima posebno veliki značaj u karakterizaciji lokalnog minimizatora funkcije te kod konstrukcije iterativne numeričke metode za rješavanje optimizacijskog problema. U nastavku navodimo definiciju i neka važna svojstva diferencijabilnih funkcija.

Definicija 1.6. [46] *Kažemo da funkcija $f : \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$ u točki $x \in \text{int } \mathcal{D}$ diferencijabilna ako postoji $a \in \mathbb{R}^n$ takav da za svaki $h \in \mathbb{R}^n$ vrijedi*

$$\lim_{t \rightarrow 0} \frac{f(x + th) - f(x) - ta^T h}{t} = 0. \quad (1.28)$$

Ako za h redom uzmemo j -ti koordinatni vektor e^j , dobivamo

$$\lim_{t \rightarrow 0} \frac{f(x + te^j) - f(x) - ta_j}{t} = 0.$$

Dakle, komponente vektora a parcijalne su derivacije $a_j = \frac{\partial f(x)}{\partial x_j} = \partial_j f(x)$.

Ako je funkcija f diferencijabilna u točki $x \in \text{int } \mathcal{D}$, onda ćemo derivaciju funkcije f u točki $x = (x_1, \dots, x_n)^T \in \text{int } \mathcal{D}$ označavati s

$$f'(x) = \left(\frac{\partial f(x)}{\partial x_1}, \dots, \frac{\partial f(x)}{\partial x_n} \right) = (\partial_1 f(x), \dots, \partial_n f(x)),$$

a gradijent funkcije f u točki x s

$$\text{grad } f(x) = \nabla f(x) = f'(x)^T.$$

Nadalje, drugu derivaciju (Hessian) funkcije f u točki x (ako postoji) označavat ćemo s

$$f''(x) = H_f(x) = \nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \cdots & \cdots & \cdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix} = \left(\frac{\partial^2 f(x)}{\partial x_i \partial x_j} \right).$$

Prisjetimo se kako skup svih p -puta neprekidno diferencijabilnih funkcija $f : \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$ označavamo s $C^p(\mathcal{D})$.

Ako je $f \in C^p(\mathcal{D})$ te $x + \alpha h \in \mathcal{D}$ za $\alpha \in [0, 1]$, onda za nju vrijedi Taylorova formula (vidi primjerice [46, 71]). Specijalno, za $p = 1, 2$ vrijedi

$$\begin{aligned} f(x + h) - f(x) &= (\nabla f(x))^T h + o(\|h\|), & p = 1, \\ f(x + h) - f(x) &= (\nabla f(x))^T h + \frac{1}{2} h^T \nabla^2 f(x) h + o(\|h\|^2), & p = 2. \end{aligned}$$

Definicija 1.7. *Kažemo da je $x^* \in \mathcal{D} \subseteq \mathbb{R}^n$ stacionarna točka funkcije $f: \mathcal{D} \rightarrow \mathbb{R}$ ako je $\nabla f(x^*) = 0$.*

Sljedeći teorem daje vezu između stacionarne točke i lokalnog minimizatora neke funkcije (vidi primjerice [25, 46, 69, 71]).

Teorem 1.1. [25] *Neka je $\mathcal{D} \subseteq \mathbb{R}^n$ otvoren skup. Ako je $x^* \in \mathcal{D}$ lokalni minimizator funkcija $f \in C^1(\mathcal{D})$, onda je x^* stacionarna točka funkcije f . Ako je dodatno $f \in C^2(\mathcal{D})$, onda je Hessian pozitivno semidefinitan (s oznakom $H_f(x^*) \geq 0$).*

Ako je $x^ \in \mathcal{D}$ stacionarna točka funkcije $f \in C^2(\mathcal{D})$, a Hessian $H_f(x^*)$ pozitivno definitan, onda je x^* strogi lokalni minimumizator funkcije f na \mathcal{D} .*

Poglavlje 2

Klasične metode optimizacije

Primjeri navedeni u prethodnom poglavlju ilustriraju da je teško očekivati postojanje jedne optimizacijske metode koja bi jednako uspješno rješavala optimizacijski problem u svim uvjetima. Iz tog razloga, pri izboru optimizacijske metode ključno je iskoristiti sve dostupne informacije o optimizacijskoj funkciji i strukturi njene domene.

U ovom poglavlju navest ćemo neke klasične optimizacijske metode koje se najčešće sreću u literaturi [5, 7, 20, 25, 46, 62, 65, 71]. Promatrat ćemo samo problem minimizacije jer, kao što smo spomenuli u uvodu (vidi t.1.1, str.2), problem maksimizacije lako se svodi na problem minimizacije. Kod tih metoda posebno važnu ulogu ima svojstvo konveksnosti i diferencijabilnosti minimizirajuće funkcije.

2.1 Metode spusta za konveksne funkcije

Minimizacijski problem vrlo često ne možemo riješiti egzaktno te ga u tom slučaju nastojimo riješiti nekom iterativnom numeričkom metodom. U iterativnim metodama obično polazimo od neke aproksimacije x^k za koju želimo odrediti vektor $p^k \neq 0$ te $\alpha_k > 0$ tako da vrijedi

$$f(x^k + \alpha_k p^k) < f(x^k), \quad k = 0, 1, \dots$$

Na taj način dobivamo niz aproksimacija

$$x^{k+1} = x^k + \alpha_k p^k, \quad (2.1)$$

gdje $p^k \in \mathbb{R}^n$ zovemo vektor smjera, a $\alpha_k > 0$ duljina koraka u smjeru p^k . Ako u nekoj iteraciji nismo u mogućnosti pronaći vektor smjera i duljinu

koraka s kojim možemo smanjiti vrijednost funkcije f , za očekivati je da je iterativni postupak pronašao stacionarnu točku funkcije f . Ako je uz to funkcija f konveksna, točka koju smo na taj način odredili ujedno je globalni minimizator funkcije f (vidi Lemu 1.3, str.26).

Primijetite da ako vektor smjera normiramo $\|p^k\| = 1$, onda α_k predstavlja udaljenost točaka x^{k+1} i x^k . Sljedeća lema pokazuje kako treba birati vektor smjera p^k i duljinu koraka α_k kako bismo postigli sniženje vrijednosti minimizirajuće funkcije.

Lema 2.1. [46] *Neka je funkcija $f: \mathcal{D} \rightarrow \mathbb{R}$ diferencijabilna u nekoj točki $x \in \text{int}(\mathcal{D})$ konveksnog skupa $\mathcal{D} \subseteq \mathbb{R}^n$ i neka je $(\nabla f(x))^T p < 0$ za neki $p \in \mathbb{R}^n$. Tada postoji $\delta > 0$ takav da je*

$$f(x + \alpha p) < f(x), \quad \forall \alpha \in (0, \delta).$$

Dokaz. Definiramo pomoćnu funkciju $\varphi: [0, +\infty) \rightarrow \mathbb{R}$, $\varphi(\alpha) = f(x + \alpha p)$, za koju je

$$\varphi'(0) = \lim_{\alpha \rightarrow 0} \frac{\varphi(\alpha) - \varphi(0)}{\alpha} = \lim_{\alpha \rightarrow 0} \frac{f(x + \alpha p) - f(x)}{\alpha} = (\nabla f(x))^T p.$$

Kako je $(\nabla f(x))^T p < 0$, postoji realan broj $\delta > 0$ takav da je $x + \alpha p \in \mathcal{D}$ i $\frac{1}{\alpha}(f(x + \alpha p) - f(x)) < 0$ za sve $\alpha \in (0, \delta)$. \square

Prema Lemi 2.1, ako je vektor smjera iz točke x^k u točku x^{k+1} dobro definiran ($(\nabla f(x^k))^T p^k < 0$), onda se u iterativnom procesu (2.1)

$$x^{k+1} = x^k + \alpha_k p^k$$

uvijek može odabrati duljina koraka $\alpha_k > 0$ tako da $f(x^{k+1}) < f(x^k)$. Za to postoje brojne metode (vidi [5, 20, 46, 65]).

Različite strategije izbora vektora smjera p_k vode ka različitim optimizacijskim metodama o kojima ćemo nešto reći u nastavku.

2.1.1 Koordinatna relaksacija

Metoda započinje izborom onog baznog vektora $e^i \in \{e^1, \dots, e^n\}$ za koji je

$$|(\nabla f(x^k))^T e^i| = \max_{j=1, \dots, n} |(\nabla f(x^k))^T e^j|.$$

Primijetite da je $(\nabla f(x^k))^T e^i = (f'(x^k))_i$. Sukladno Lemi 2.1, vektor smjera p^k izabrat ćemo između vektora $\{e^i, -e^i\}$ za koji je

$$(\nabla f(x^k))^T p^k < 0.$$

Umjesto ortonormiranih baznih vektora $\{e^1, \dots, e^n\}$, u metodi možemo koristiti neke druge bazne vektore iz \mathbb{R}^n . Metoda ima nisku brzinu konvergencije i koristi se za određivanje početne aproksimacije za neku drugu iterativnu metodu [46].

2.1.2 Gradijentna metoda

Za vektor $p^k := -\nabla f(x^k)$ vrijedi

$$(\nabla f(x^k))^T p^k = -(\nabla f(x^k))^T \nabla f(x^k) = -\|\nabla f(x^k)\|^2 < 0.$$

Zato sukladno Lemi 2.1, kao vektor smjera ima smisla izabrati ovakav vektor p^k koji određuje **gradijentnu metodu**. U literaturi (vidi primjerice [10, 20, 46]) ova metoda naziva se još i *Metoda najbržeg spusta*. *Gradijentna metoda* sporo konvergira – ima linearnu brzinu konvergencije [7, 10, 20, 46].

2.1.3 Newtonova metoda za minimizaciju bez ograničenja

Za vektor $p^k := -[\nabla^2 f(x^k)]^{-1} \nabla f(x^k)$ vrijedi

$$(\nabla f(x^k))^T p^k = -(\nabla f(x^k))^T [\nabla^2 f(x^k)]^{-1} \nabla f(x^k),$$

što je negativno ako je $\nabla^2 f(x^k)$ pozitivno definitna matrica (u tom slučaju je i $[\nabla^2 f(x^k)]^{-1}$ pozitivno definitna). Zato sukladno Lemi 2.1, kao vektor smjera ima smisla izabrati ovakav vektor p^k .

U cilju izbjegavanja numerički nestabilnog izračunavanja inverzne matrice $[\nabla^2 f(x^k)]^{-1}$, iterativni postupak uvijek se definira na sljedeći način:

$$x^{k+1} = x^k + s^k, \quad x^0 \in \mathbb{R}^n,$$

gdje je s^k rješenje jednadžbe

$$\nabla^2 f(x^k) s = -\nabla f(x^k).$$

Ovako definirana metoda poznata je kao **Newtonova metoda za minimizaciju bez ograničenja** [7, 10, 20, 46]. Primijetite da se ovaj iterativni postupak potpuno podudara s *Newtonovom metodom* za rješavanje jednadžbe $\nabla f(x) = 0$ (vidi primjerice [62, 65]).

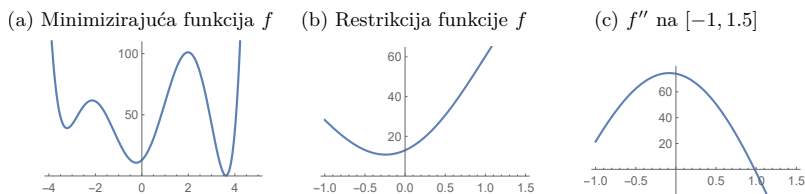
Algoritam 1 (Newtonova metoda)**Input:** $f: \mathbb{R}^n \rightarrow \mathbb{R}$;**Input:** $x^0 \in \mathbb{R}^n$; $\epsilon > 0$; It , $k = 0$;

{Učitati funkciju, početnu aproksimaciju, točnost i maksimalno dozvoljeni broj iteracija;}

1: **while** $\|\nabla f(x^k)\| > \epsilon$ and $k < It$ **do**2: Riješi sustav $\nabla^2 f(x^k)s = -\nabla f(x^k)$ i rješenje označi sa s^k ;3: Stavi $x^{k+1} = x^k + s^k$; $k = k + 1$;4: **end while****Output:** $\{k, x^{k+1}\}$

Primjer 2.1. Promatramo funkciju $f: \mathbb{R} \rightarrow \mathbb{R}$ zadanu s (1.12) iz Primjera 1.2, str. 7, čiji je graf prikazan na Slici 2.1a. Na Slici 2.1b prikazana je restrikcija minimizirajuće funkcije f na interval $[-1, 1.5]$, a na Slici 2.1c vidi se da ta restrikcija nije konveksna. Zato će trebati još suziti interval promatranja.

Newtonovom metodom minimizacije potražiti ćemo lokalni minimum ove funkcije na intervalu $[-1, 1.5]$. Uz odabir početne aproksimacije $x_0 = 0.6$, dobivamo niz aproksimacija (vidi Tablicu 2.1) u kojima funkcija f prima padajući niz vrijednosti. Vrijednost derivacije $f'(x_3) = 0.09$ pokazuje da smo se već u trećoj iteraciji približili točnosti na jednu decimalu, a u četvrtoj iteraciji točnosti na 4 decimale. Vrijednost derivacije funkcije u nekoj aproksimaciji može samo sugerirati da smo bliže ili dalje od tražene točke minimuma, ali ne može to i garantirati. Drugi indikator točnosti aproksimacije je međusobna udaljenost susjednih aproksimacija, ali naravno, ni to nije garancija točnosti aproksimacije. Egzaktni kriteriji točnosti aproksimacije obično su toliko složeni da se u praktičnim primjenama rijetko koriste (vidi primjerice [10]).

Slika 2.1: LS udaljenost točke T_0 do parabole q

Uz početnu aproksimaciju $x_0 = .8$, i $f(x_0) = 47.66$ u sljedećoj aproksimaciji dobivamo: $x_1 = -2.078$, i $f(x_1) = 61.564$, dakle vrijednost funkcije ne opada.

Zato u Algoritam 1 uključujemo izračunavanje duljine koraka:

k	x_k	$f(x_k)$	$f'(x_k)$
0	.6	35.922	55.173
1	-0.718	18.339	29.641
2	-0.094	11.638	11.022
3	-0.242	10.819	0.093
4	-0.243	10.819	0.00002

Tablica 2.1: *Newtonova metoda minimizacije*

Algoritam 2 (Izbor duljine koraka)

Input: $f: \mathbb{R}^n \rightarrow \mathbb{R}; s \in \mathbb{R}^n$;

Input: $x^k \in \mathbb{R}^n; \tau \in (0, \frac{1}{2}); \nu \in (0, 1); \lambda = 1$;

{Učitati funkciju, k -tu aproksimaciju i parametre τ, ν, λ_{min} };

1: Riješiti sustav: $\nabla^2 f(x^k)s = -\nabla f(x^k)$ i rješenje označiti sa s^k ;

2: **while** $f(x^k + \lambda s^k) - f(x^k) > \tau \lambda (\nabla f(x^k))^T \cdot s^k$ and $\lambda > \lambda_{min}$, **do**

3: $\lambda = \lambda \cdot \nu$

4: **end while**

Output: λ ,

Nakon toga, u Algoritmu 1 Korak 3: zamijenimo s

$$3: \quad x^{k+1} = x^k + \lambda s^k, \quad k = k + 1.$$

Ovako redefinirani Algoritam 1 u literaturi se naziva *Newtonova metoda s regulacijom koraka*.

U tom slučaju iterativni proces uz početnu aproksimaciju $x_0 = .8$ daje niz aproksimacija u kojima funkcija f postiže opadajuće vrijednosti, a već u trećoj iteraciji postiže točnost na jednu decimalu (vidi Tablicu 2.2).

k	x_k	$f(x_k)$	$f'(x_k)$	λ_k
0	.8	47.664	61.564	.5
1	-0.639	16.143	25.630	1
2	-0.160	11.071	6.099	1
3	-0.242	10.819	0.050	-

Tablica 2.2: *Newtonova metoda s regulacijom koraka*

Primjer 2.2. Promatramo funkciju [61] $f: \mathbb{R}^2 \rightarrow \mathbb{R}, f(x_1, x_2) = 2x_1^3 + x_1x_2^2 + 5x_1^2 + x_2^2$ [61], čiji su gradijent i Hessian

$$\nabla f(x_1, x_2) = \begin{bmatrix} 10x_1 + 6x_1^2 + x_2^2 \\ 2x_2 + 2x_1x_2 \end{bmatrix}; \quad \nabla^2 f(x_1, x_2) = \begin{bmatrix} 10 + 12x_1 & 2x_2 \\ 2x_2 & 2 + 2x_1 \end{bmatrix}. \quad (2.2)$$

Može se pokazati da ova funkcija ima četiri stacionarne točke: $S_1 = (-\frac{5}{3}, 0)$, $S_2 = (-1, -2)$, $S_3 = (-1, 2)$, $S_4 = (0, 0)$, i da u točki S_1 postiže lokalni maksimum, u točki S_4 lokalni minimum, a S_2 i S_3 su sedlaste točke.

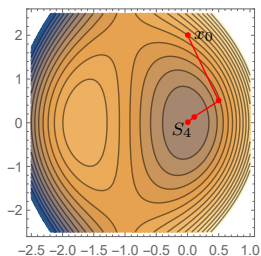
U t.6.2.1, str.123 opisani su *Mathematica*-moduli *Newton* i *NewtonKorak* koji implementiraju *Newtonovu metodu* (Algoritam 1) te *Newtonovu metodu uz izbor duljine koraka* (Algoritam 2). Obje varijante Newtonove metode zahtijevaju poznavanje gradijenta i Hessiana minimizirajuće funkcije. Programski sustav *Mathematica* omogućava simboličko računanje derivacija funkcija korištenjem naredbe *D*. Tako primjerice izraze (2.2) možemo jednostavno dobiti sljedećim naredbama:

```
In[1]:= Clear[x1, x2];
        f[x1_, x2_] := 2*x1^3 + x1*x2^2 + 5*x1^2 + x2^2;
        D[f[x1, x2], {x1, x2}]
Out[1]:= {10 x1 + 6 x1^2 + x2^2, 2 x2 + 2 x1 x2}

In[2]:= D[f[x1, x2], {{x1, x2}}, {{x1, x2}}]
Out[2]:= {{10 + 12 x1, 2 x2}, {2 x2, 2 + 2 x1}}
```

Dobivene izraze moguće je direktno iskoristiti za definiranje gradijenta i Hessiana kao funkcija argumenta (x_1, x_2) , što je i napravljeno u modulima opisanim u t.6.2.1, str.123.

Newtonov iterativni postupak (Algoritam 1) uz izbor duljine koraka (Algoritam 2) i početnu aproksimaciju $x_0 = (0, 2)$ prikazan je na Slici 2.2. Već nakon tri iteracije dobivamo aproksimaciju stacionarne točke S_4 , u kojoj funkcija f postiže lokalni (ujedno i globalni) minimum.



Slika 2.2: Traženje točke minimuma *Newtonovom metodom*

Zadatak 2.1. *Newtonovom metodom* odredite točku maksimuma funkcije iz Primjera 2.2.

Zadatak 2.2. Odredite prve tri aproksimacije minimuma funkcije $f(x) = x^3 - 4x + 5$ na intervalu $[0, 2]$ uz početnu aproksimaciju $x^0 = 2$ korištenjem *Newtonove metode*.

Zadatak 2.3. Odredite prve tri aproksimacije minimuma funkcije $f(x) = x^3 - 4x + 5$ na intervalu $[0, 2]$ uz početnu aproksimaciju $x^0 = 0.5$ korištenjem *Newtonove metode* s izborom duljine koraka (za parametre uzmite $\tau = 0.25$ i $\nu = 0.5$).

Zadatak 2.4. *Newtonovom metodom* odredite prve tri aproksimacije minimuma funkcije $F(x, y) = x^4 + y^4 + y^2$ uz početnu aproksimaciju $(x^0, y^0)^T = (0.5, -0.5)^T$.

Zadatak 2.5. *Newtonovom metodom s izborom duljine koraka* odredite prve tri aproksimacije minimuma funkcije $F(x, y) = e^x + x^4 + y^4$ uz početnu aproksimaciju $(x^0, y^0)^T = (0.5, -0.5)^T$ (za parametre uzmite $\tau = 0.25$ i $\nu = 0.5$).

2.1.4 Kvazi-Newtonova metoda

Za vektor $p^k := -B^{-1}\nabla f(x^k)$, gdje je $B > 0$ pozitivno definitna matrica koja „nalikuje” na $\nabla^2 f(x^k)$, vrijedi

$$(\nabla f(x^k))^T p^k = -(\nabla f(x^k))^T B^{-1} \nabla f(x^k) < 0.$$

Zato sukladno Lemi 2.1, kao vektor smjera možemo izabrati ovakav vektor smjera p^k .

U ovisnosti o izboru matrice B dobivamo poznate kvazi-Newtonove metode: Broyden Method, Davidon–Fletcher–Powell Method (DFP), Broyden–Fletcher–Goldfarb–Shano Method (BFGS), itd (vidi [20, 62, 65]).

2.2 Nelder-Meadova metoda

Analizirat ćemo također jednu lokalnu metodu za traženje minimuma realne funkcije $f: \mathbb{R}^n \rightarrow \mathbb{R}$ koja je u literaturi poznata kao *Nelder-Meadova metoda*. Ime je dobila po svojim autorima R. Nelderu i J. A. Meadu koji su metodu objavili 1965. godine u radu [44]. Metoda je jednostavna i u mnogim slučajevima vrlo efikasna, posebno u malim dimenzijama. *Nelder-Meadovu metodu* moguće je primijeniti na funkciju za koju je dovoljno poznavati isključivo vrijednosti u svim točkama domene. Nažalost, u velikim dimenzijama metoda se ne pokazuje efikasnom (vidi primjerice [36, 52, 67]), osim toga, općenito ne postoji dokaz konvergencije ove metode. U radu [36] dani su uvjeti konvergencije za specijalni tip funkcije u jednoj i dvije dimenzije. Iz metodičkih razloga posebno ćemo analizirati samo metodu za funkcije dvije varijable.

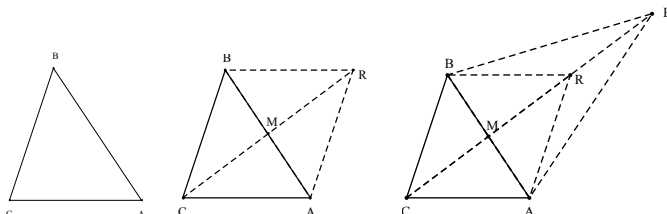
2.2.1 Nelder-Meadova metoda za funkcije dviju varijabli

Pretpostavimo da je zadana funkcija $f: \mathbb{R}^2 \rightarrow \mathbb{R}$. Promatrajmo trokut $\triangle ABC \subset \mathbb{R}^2$ s vrhovima $A = (x_1, y_1)$, $B = (x_2, y_2)$ i $C = (x_3, y_3)$. Bez

smanjenja općenitosti možemo pretpostaviti da je

$$f(A) \leq f(B) \leq f(C).$$

Pri tome ćemo točku A zvati najbolji vrh, točku C najlošiji vrh, dok ćemo točku B zvati drugi najlošiji vrh.



Slika 2.3: Trokut $\triangle ABC$ (lijevo) i njegova refleksija (sredina) i ekspanzija (desno)

Osnovna ideja jedne iteracije *Nelder-Meadove metode* sastoji se u tome da trokut $\triangle ABC$ zamijenimo novim trokutom u čijim vrhovima funkcija f postiže manju vrijednost od one koja se postiže u vrhovima trokuta $\triangle ABC$. U tu svrhu centroid vrhova A i B označimo s $M = \frac{1}{2}(A+B)$. Potom najlošiji vrh C preslikajmo centralno-simetrično u odnosu na točku M . Na taj način dobivamo vrh

$$R = M + (M - C) = 2M - C = A + B - C.$$

Transformaciju koja trokut $\triangle ABC$ preslikava u trokut $\triangle ABR$ zvat ćemo refleksija trokuta (vidi Sliku 2.3). Pri tome nas zanima je li $f(R) < f(A)$, odnosno je li vrh R bolji od najboljeg vrha A ?

- Ako je odgovor pozitivan, naslućujemo da se krećemo u pravom smjeru te se u istom smjeru pomaknemo do točke

$$E = R + (R - M) = 2R - M = 2A + 2B - 2C - \frac{1}{2}A - \frac{1}{2}B = \frac{3}{2}A + \frac{3}{2}B - 2C.$$

Transformaciju koja trokut $\triangle ABC$ preslikava u trokut $\triangle ABE$ zovemo ekspanzija trokuta (vidi Sliku 2.3). Točku C zamijenit ćemo točkom E ako je $f(E) < f(R)$, odnosno točkom R ako je $f(E) \geq f(R)$.

- Ako je odgovor negativan, odnosno ako je $f(R) \geq f(A)$, točku R uspoređujemo s drugom najlošijom točkom B , odnosno postavljamo sljedeće pitanje: je li $f(R) < f(B)$?

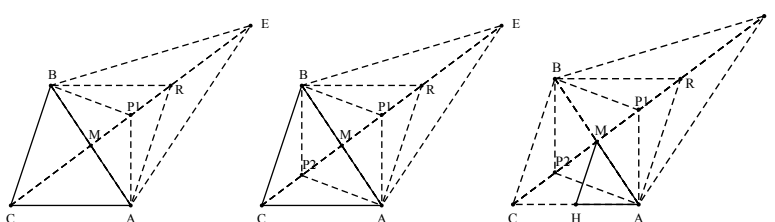
- (1) Ako je odgovor pozitivan, točku C zamjenjujemo točkom R .
- (2) Ako je odgovor negativan, postavljamo novo pitanje: je li $f(R) < f(C)$?
 - (2a) Ako je odgovor pozitivan, točku C zamjenjujemo točkom R te definiramo novu točku

$$P = M + \frac{1}{2}(R - M) = \frac{1}{2}M + \frac{1}{2}R = \frac{3}{4}A + \frac{3}{4}B - \frac{1}{2}C.$$

- (2b) Ako je odgovor negativan, definiramo novu točku

$$P = M - \frac{1}{2}(M - C) = \frac{1}{2}M + \frac{1}{2}C = \frac{1}{4}A + \frac{1}{4}B + \frac{1}{2}C.$$

Transformaciju koja trokut $\triangle ABC$ preslikava u trokut $\triangle ABP$ zovemo kontrakcijom trokuta (vidi Sliku 2.4).

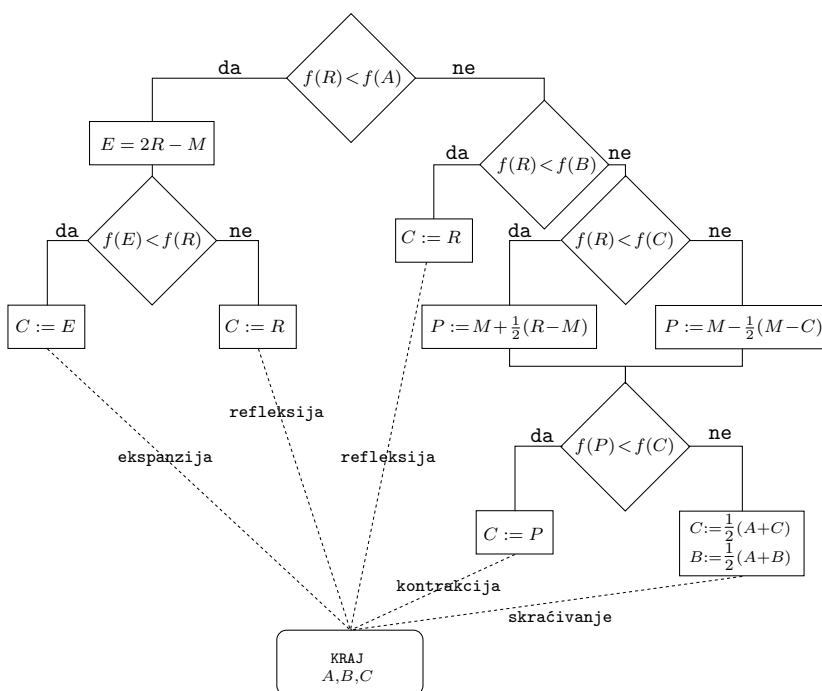


Slika 2.4: Kontrakcija trokuta (lijevo i u sredini) i njegovo skraćivanje (desno)

Konačno, pitamo se je li $f(P) < f(C)$.

- (2c) Ako je odgovor pozitivan, točku C zamjenjujemo točkom P .
- (2d) Ako je odgovor negativan, definiramo $H = \frac{1}{2}(A + C)$ te točku B zamijenimo s M , a točku C zamijenimo s H .

Transformaciju koja trokut $\triangle ABC$ preslikava u trokut $\triangle AMH$ zovemo skraćivanjem trokuta (vidi Sliku 2.4).

Slika 2.5: Dijagram toka za *Nelder-Meadovu metodu* u \mathbb{R}^2

Iteracije *Nelder-Meadovog algoritma* ponavljaju se dok se ne zadovolji neki kriterij zaustavljanja. Jedna mogućnost kriterija zaustavljanja može biti udaljenost najbolje točke do težišta trokuta $\triangle ABC$, odnosno algoritam se ponavlja dok god je $d\left(A, \frac{1}{3}(A + B + C)\right) \geq \varepsilon$, gdje su $d : \mathbb{R}^2 \rightarrow [0, \infty)$ neka funkcija udaljenosti te $\varepsilon > 0$ unaprijed zadana točnost.

U t.6.2.2, str.126 opisana je uporaba *Mathematica*-modula `NelderMead` koji implementira *Nelder-Meadovu metodu* za funkcije dviju varijabli na osnovu Algoritma 3. Modul ima i mogućnost grafičkog prikaza transformacija trokuta koje se primjenjuju u iteracijama algoritma (vidjeti Primjer 6.3, str.126).

Algoritam 3 (Nelder-Meadov algoritam za funkcije dviju varijabli)

Input: f , $A = (x_1, y_1)$, $B = (x_2, y_2)$, $C = (x_3, y_3)$

- 1: poredaj A, B, C tako da je $f(A) \leq f(B) \leq f(C)$
- 2: $M = \frac{1}{2}(A + B)$ //centroid dobre strane
- 3: $R = M + (M - C) = 2M - C$ //refleksija
- 4: **if** $f(R) < f(A)$ **then**
- 5: $E = R + (R - M) = 2R - M$ //ekspanzija
- 6: **if** $f(E) < f(R)$ **then**
- 7: $C = E$
- 8: **else**
- 9: $C = R$
- 10: **end if**
- 11: **else**
- 12: **if** $f(R) < f(B)$ **then**
- 13: $C = R$
- 14: **else**
- 15: **if** $f(R) < f(C)$ **then**
- 16: $P = M + \frac{1}{2}(R - M)$ //kontrakcija
- 17: **else**
- 18: $P = M - \frac{1}{2}(M - C)$ //kontrakcija
- 19: **end if**
- 20: **end if**
- 21: **if** $f(P) < f(C)$ **then**
- 22: $C = P$
- 23: **else**
- 24: $B = \frac{1}{2}(A + B)$ //skraćivanje
- 25: $C = \frac{1}{2}(A + C)$ //skraćivanje
- 26: **end if**
- 27: **end if**

Output: A, B, C

Zadatak 2.6. Zadan je trokut $\triangle ABC$ s vrhovima $A = (5, 1)$, $B = (3, 4)$ i $C = (1, 2)$. Odredite vrhove trokuta nastalih refleksijom, ekspanzijom, unutarnjom i vanjskom kontrakcijom te skraćivanjem trokuta $\triangle ABC$.

Zadatak 2.7. Za funkciju $f: \mathbb{R}^2 \rightarrow \mathbb{R}$, $f(x_1, x_2) = \max\{|x_1 + 2x_2 - 4|, |x_1 - x_2|\}$ provedite prve dvije iteracije *Nelder-Meadove metode* tako da krenete od trokuta s vrhovima

- (a) $A = (3, 0), B = (0, 3), C = (0, 0)$,
- (b) $A = (3, 2), B = (1, 3), C = (3, 4)$.

U svakoj iteraciji odredite aproksimaciju minimuma kao vrijednost funkcije u centroidu trenutnog trokuta.

2.3 Kvadratne interpolacijske metode za jednodimenzionalnu minimizaciju

Pretpostavimo da je zadana funkcija $f: \mathbb{R} \rightarrow \mathbb{R}$ koja u točki $x^* \in \mathbb{R}$ postiže lokalni minimum. U nastavku ćemo opisati jednu klasu metoda za numeričko određivanje točke lokalnog minimuma poznatu kao kvadratne interpolacijske metode. Zajedničko kvadratnim interpolacijskim metodama je interpolacija funkcije f kvadratnom funkcijom u jednoj, dvjema ili trima točkama. Tada se umjesto traženja točke lokalnog minimuma funkcije f traži točka globalnog minimuma kvadratne funkcije te se postupak uz određena pravila iterativno nastavlja. U slučaju interpolacije u jednoj točki dobivamo najčešće korištenu *Newtonovu metodu* ili *Metodu jedne točke*, dok u slučajevima interpolacije u dvjema odnosno trima točkama, dobivamo *Metodu dvije točke*, odnosno *Metodu tri točke*. Ako je funkcija f konveksna i ako ima točku lokalnog minimuma, onda spomenute metode pronalaze globalni minimum.

2.3.1 Newtonova metoda – metoda jedne točke

Ovdje ćemo pretpostaviti da je funkcija f dva puta neprekidno derivabilna. Neka je zadana točka $x_0 \in \mathbb{R}$, za koju pretpostavljamo da leži dovoljno blizu točke minimuma x^* koju želimo odrediti. Funkciju f u okolini točke x_0 aproksimirat ćemo kvadratnom funkcijom $g: \mathbb{R} \rightarrow \mathbb{R}$, $g(x) = \alpha x^2 + \beta x + \gamma$, pri čemu koeficijente α, β, γ određujemo iz sljedećih uvjeta:

$$\begin{aligned} g(x_0) &= \alpha x_0^2 + \beta x_0 + \gamma = f(x_0) =: f_0, \\ g'(x_0) &= 2\alpha x_0 + \beta = f'(x_0) =: f'_0, \\ g''(x_0) &= 2\alpha = f''(x_0) =: f''_0. \end{aligned} \quad (2.3)$$

Rješavanjem sustava (2.3) dobivamo:

$$\alpha = \frac{1}{2}f''_0, \quad \beta = f'_0 - f''_0 x_0, \quad \gamma = f_0 - \alpha x_0^2 - \beta x_0.$$

Umjesto traženja točke lokalnog minimuma funkcije f potražiti ćemo točku u kojoj se postiže globalni minimum njezine aproksimacije g . Kako je g kvadratna funkcija, njezin globalni minimum postiže se u točki

$$\bar{x} = -\frac{\beta}{2\alpha} = x_0 - \frac{f'_0}{f''_0},$$

odakle dobivamo rekurzivnu formulu na kojoj se temelji poznata *Newtonova metoda*:

$$x_{k+1} = x_k - \frac{f'_k}{f''_k}, \quad k = 0, 1, \dots, \quad (2.4)$$

pri čemu su $f'_k := f'(x_k)$ te $f''_k := f''(x_k)$, $k = 0, 1, \dots$

U udžbenicima iz numeričke analize mogu se pronaći teoremi o konvergenciji *Newtonove metode* te njezinoj brzini konvergencije. S obzirom na to da je riječ o standardnim rezultatima, sljedeći teorem navodimo bez dokaza. Iskaz i dokaz ovog teorema s nešto blažim uvjetima na funkciju f možemo naći u [62].

Teorem 2.1. *Neka je $f: \mathbb{R} \rightarrow \mathbb{R}$ tri puta neprekidno diferencijabilna funkcija. Ako je $x^* \in \mathbb{R}$ stacionarna točka takva da je $f''(x^*) \neq 0$, onda postoji $\delta > 0$ takav da za svaki $x_0 \in (x^* - \delta, x^* + \delta)$ niz (x_k) zadan rekursivno sa (2.4) konvergira prema x^* kvadratnom brzinom konvergencije.*

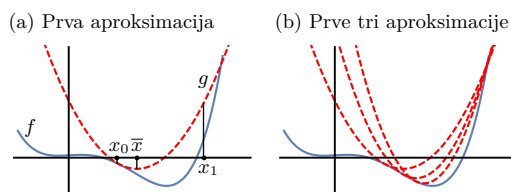
Primijetimo da je za primjenu rekursivne formule (2.4) dovoljno da funkcija ima drugu derivaciju. Međutim, kako bi se dokazao prethodno navedeni teorem o konvergenciji te brzini konvergencije, postavljen je zahtjev da je funkcija tri puta neprekidno derivabilna. Situacija u kojoj su za primjenu metode potrebni blaži uvjeti, dok su za bilo kakav teorijski rezultat potrebni puno jači uvjeti, vrlo je česta u numeričkoj analizi.

U sljedećim pododjeljcima analizirat ćemo metode za koje nije potrebno izračunavanje druge derivacije, već samo vrijednost prve derivaciju ili samo vrijednost funkcije. Smanjenje zahtjeva na funkciju očekivano će rezultirati smanjenjem brzine konvergencije odgovarajuće metode.

2.3.2 Metoda dvije točke

Metoda I

Neka su $x_0, x_1 \in \mathbb{R}$, $x_0 \neq x_1$ dvije točke za koje su poznate vrijednosti $f(x_0)$, $f'(x_0)$ i $f'(x_1)$, pri čemu dodatno pretpostavljamo da je $f'(x_0) \neq f'(x_1)$.



Slika 2.6: Graf funkcije f i kvadratnih interpolacijskih funkcija

U tom slučaju možemo potražiti kvadratnu interpolacijsku funkciju $g: \mathbb{R} \rightarrow \mathbb{R}$ oblika $g(x) = \alpha x^2 + \beta x + \gamma$, $\alpha, \beta, \gamma \in \mathbb{R}$, sa sljedećim svojstvima (vidi

Sliku 2.6):

$$\begin{aligned} g(x_0) &= \alpha x_0^2 + \beta x_0 + \gamma = f(x_0) =: f_0, \\ g'(x_0) &= 2\alpha x_0 + \beta = f'(x_0) =: f'_0, \\ g'(x_1) &= 2\alpha x_1 + \beta = f'(x_1) =: f'_1. \end{aligned} \quad (2.5)$$

Rješavanjem sustava linearnih jednadžbi (2.5) dobivamo

$$\alpha = \frac{f'_1 - f'_0}{2(x_1 - x_0)}, \quad \beta = \frac{x_1 f'_0 - x_0 f'_1}{x_1 - x_0}, \quad \gamma = f_0 - \alpha x_0^2 - \beta x_0.$$

Kvadratna funkcija g postiže globalni minimum u točki

$$\bar{x} = -\frac{\beta}{2\alpha} = x_0 - \frac{x_1 - x_0}{f'_1 - f'_0} f'_0.$$

Općenito, dobivamo rekurzivnu formulu

$$x_{k+1} = x_{k-1} - \frac{x_k - x_{k-1}}{f'_k - f'_{k-1}} f'_{k-1}, \quad k = 1, 2, \dots, \quad (2.6)$$

pri čemu su $f'_k := f'(x_k)$, $k = 0, 1, \dots$

Algoritam 4 (Metoda dvije točke)

Input: f, x_0, x_1, ε

- 1: $f'_0 = f'(x_0), f'_1 = f'(x_1)$
- 2: $\bar{x} = x_0 - \frac{x_0 - x_1}{f'_0 - f'_1} f'_0$
- 3: **while** $|f'(\bar{x})| < \varepsilon$ (ili dok nije zadovoljen neki drugi kriterij zaustavljanja) **do**
- 4: $x_0 = x_1, x_1 = \bar{x}$
- 5: $f'_0 = f'(x_0), f'_1 = f'(x_1)$
- 6: $\bar{x} = x_0 - \frac{x_0 - x_1}{f'_0 - f'_1} f'_0$
- 7: **end while**

Output: \bar{x}

U sljedećem teoremu odgovaramo na pitanje konvergencije i brzine konvergencije iterativnog postupka Algoritma 4 (vidi [68, 72]).

Teorem 2.2. *Neka je $f: \mathbb{R} \rightarrow \mathbb{R}$ tri puta neprekidno diferencijabilna funkcija. Ako je $x^* \in \mathbb{R}$ stacionarna točka takva da je $f''(x^*) \neq 0$ i $f'''(x^*) \neq 0$, onda postoji $\delta > 0$ takav da za svaki $x_0, x_1 \in \langle x^* - \delta, x^* + \delta \rangle$, $x_0 \neq x_1$, niz (x_k) zadan rekurzivno s (2.6) konvergira prema x^* , pri čemu je brzina konvergencije metode $r = \frac{1+\sqrt{5}}{2} \approx 1.618$.*

Iz iterativne procedure (2.6) proizlazi Algoritam 4. Uočimo da smo u Algoritmu 4 naveli kriterij zaustavljanja, prema kojemu algoritam staje kada je derivacija funkcije f u trenutnoj aproksimaciji manja od unaprijed zadanog broja $\varepsilon > 0$. U literaturi postoje i drugi kriteriji zaustavljanja, ali budući da zahtijevaju posebnu analizu i obrazloženja, ovdje ih nećemo razmatrati.

Metoda II

U ovom pododjeljku kratko ćemo opisati još jednu *Metodu dvije točke*. Slično kao i u prvoj metodi, pretpostavimo da su $x_0, x_1 \in \mathbb{R}$, $x_0 < x_1$ dvije različite točke za koje su poznate vrijednosti $f(x_0)$, $f(x_1)$ te $f'(x_0)$, pri čemu je $f(x_0) \neq f(x_1)$. U tom slučaju tražimo kvadratnu interpolacijsku funkciju $g: \mathbb{R} \rightarrow \mathbb{R}$, $g(x) = \alpha x^2 + \beta x + \gamma$ sa sljedećim svojstvima:

$$\begin{aligned} g(x_0) &= \alpha x_0^2 + \beta x_0 + \gamma = f(x_0) =: f_0, \\ g(x_1) &= \alpha x_1^2 + \beta x_1 + \gamma = f(x_1) =: f_1, \\ g'(x_0) &= 2\alpha x_0 + \beta = f'(x_0) =: f'_0. \end{aligned} \quad (2.7)$$

Rješavanjem sustava (2.7) dobivamo

$$\begin{aligned} \alpha &= \frac{f_0 - f_1 - f'_0(x_0 - x_1)}{-(x_0 - x_1)^2}, \\ \beta &= f'_0 + 2x_0 \frac{f_0 - f_1 - f'_0(x_0 - x_1)}{(x_0 - x_1)^2}, \\ \gamma &= f(x_0) - \alpha x_0^2 - \beta x_0, \end{aligned}$$

odakle slijedi rekurzivna formula:

$$x_{k+1} = x_{k-1} - \frac{1}{2} \frac{(x_{k-1} - x_k) f'_{k-1}}{f'_{k-1} - \frac{f_k - f_{k-1}}{x_k - x_{k-1}}}. \quad (2.8)$$

Za metodu koja se zasniva na iterativnoj proceduri (2.8) po uzoru na dokaz Teorema 2.2 može se pokazati analogna tvrdnja.

Zadatak 2.8. *Metodom dvije točke* odredite aproksimaciju minimuma funkcije $f(x) = x^3 - 4x + 5$ korištenjem kriterija zaustavljanja iz Algoritma 4 uz $\varepsilon = 0.005$. Upotrijebite obje varijante metode.

2.3.3 Metoda tri točke

Neka je $f: \mathbb{R} \rightarrow \mathbb{R}$ neprekidna funkcija koja općenito ne mora biti derivabilna. Kao i do sada, s x^* označimo nepoznatu točku u kojoj se postiže lokalni minimum funkcije f . Odaberimo tri međusobno različite početne

aproksimacije x_0, x_1 i x_2 i to tako da su točke $(x_0, f(x_0)), (x_1, f(x_1))$ te $(x_2, f(x_2))$ nekolinearne te da vrijedi

$$\min_{i=0,1,2} x_i \leq x^* \leq \max_{i=0,1,2} x_i.$$

U tom slučaju postoji jedinstveni interpolacijski polinom g drugog stupnja koji prolazi kroz te tri točke te glasi:

$$g(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}f_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}f_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}f_2,$$

gdje su $f_k := f(x_k), k = 0, 1, 2$. Slično kao i kod prethodnih metoda, umjesto točke u kojoj se postiže lokalni minimum funkcije f tražimo točku u kojoj se postiže globalni minimum kvadratne funkcije g . Nije teško vidjeti da iz $g'(x) = 0$ dobivamo

$$\bar{x} = \frac{1}{2}(x_0 + x_1) + \frac{1}{2} \frac{(f_0 - f_1)(x_1 - x_2)(x_2 - x_0)}{(x_1 - x_2)f_0 + (x_2 - x_0)f_1 + (x_0 - x_1)f_2}. \quad (2.9)$$

Iz formule (2.9) proizlazi metoda kod koje polazeći od trojke točaka x_0, x_1 i x_2 dobivamo novu točku \bar{x} . U svakoj od tako dobivenih četvorki točaka izračunamo vrijednost funkcije f te među njima biramo one tri točke $\tilde{x}_0, \tilde{x}_1, \tilde{x}_2$ u kojima je vrijednost funkcije najmanja. Postupak nastavljamo sve dok se ne zadovolji neka unaprijed zadana točnost. Cijeli postupak opisan je u Algoritmu 5.

Vrijedi sljedeći teorem, čiji iskaz i dokaz možemo naći primjerice u [72].

Teorem 2.3. *Neka je $f : \mathbb{R} \rightarrow \mathbb{R}$ četiri puta neprekidno derivabilna funkcija. Ako je $x^* \in \mathbb{R}$ stacionarna točka takva da je $f''(x^*) \neq 0, f'''(x^*) \neq 0$ i $f^{(iv)}(x^*) \neq 0$, onda postoji $\delta > 0$ takav da za svaki $x_0, x_1, x_2 \in \langle x^* - \delta, x^* + \delta \rangle, x_i \neq x_j, i \neq j$, Algoritam 5 konvergira prema x^* , pri čemu je brzina konvergencije metode $r = 1.32$.*

Analogno kao i do sada u svrhu dokaza teorema potrebno je postaviti znatno jači zahtjev na funkciju f od onoga koji je potreban za provođenje Algoritma 5.

Zadatak 2.9. Zadana je konveksna funkcija $f \in C^1([a, b])$ koja u točki $\xi \in [a, b]$ postiže jedinstveni globalni minimum. Neka su $x_0, x_1, x_2 \in [a, b]$ tri različite točke za koje su poznate vrijednosti $f(x_0), f(x_1)$ te $f'(x_2)$.

(a) Odredite kvadratnu interpolacijsku funkciju $g : [a, b] \rightarrow \mathbb{R}, g(x) = \alpha x^2 + \beta x + \gamma$ sa sljedećim svojstvima:

$$g(x_0) = f(x_0) =: f_0, \quad g(x_1) = f(x_1) =: f_1, \quad g'(x_2) = f'(x_2) =: f'_2.$$

Algoritam 5 (Metoda tri točke)

Input: $f, \varepsilon, x_0, x_1, x_2$ (uz uvjet $\min\{x_0, x_1, x_2\} \leq x^* \leq \max\{x_0, x_1, x_2\}$)

- 1: $f_0 = f(x_0), f_1 = f(x_1), f_2 = f(x_2)$
- 2: $\bar{x} = \frac{1}{2}(x_0 + x_1) + \frac{1}{2} \frac{(f_0 - f_1)(x_1 - x_2)(x_2 - x_0)}{(x_1 - x_2)f_0 + (x_2 - x_0)f_1 + (x_0 - x_1)f_2}$
- 3: **if** $(\bar{x} - x_0)(\bar{x} - x_2) \geq 0$ **then**
- 4: **goto** Korak 8
- 5: **else**
- 6: **goto** Korak 9
- 7: **end if**
- 8: Konstruirati novu trojku $\{x_0, x_1, x_2\}$ iz x_0, x_1, x_2 i \bar{x} za koje je vrijednost funkcije f najmanja i **goto** Korak 1
- 9: **if** $|\bar{x} - x_1| < \varepsilon$ **then**
- 10: **goto** Korak 13
- 11: **else**
- 12: **goto** Korak 8
- 13: **end if**

Output: \bar{x}

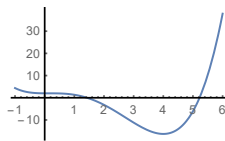
- (b) Odredite točku u kojoj funkcija g postiže globalni minimum.
- (c) Na osnovi (b) predložite iterativnu formulu koja će konvergirati jedinstvenom globalnom minimumu funkcije f .

2.3.4 Numerički primjer

U ovom ćemo odjeljku pomoću jednog jednostavnog numeričkog primjera ilustrirati efikasnost prethodno navedenih metoda te potvrditi teorijske rezultate koji se odnose na brzinu konvergencije pojedine metode. Promatrajmo u tu svrhu funkciju $f: \mathbb{R} \rightarrow \mathbb{R}$ zadanu formulom

$$f(x) = 2 - \frac{2}{25}x + \frac{61}{100}x^2 - \frac{43}{30}x^3 + \frac{1}{4}x^4.$$

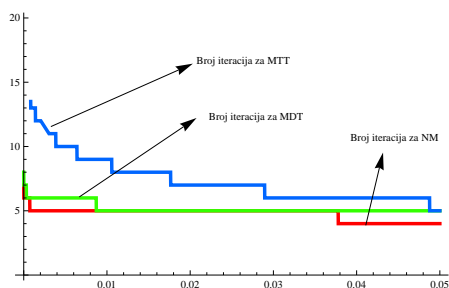
Graf funkcije f prikazan je na Slici 2.7.



Slika 2.7: Graf funkcije f

Nije teško vidjeti da funkcija f u točki $x^* = 4$ postiže lokalni minimum, koji je ujedno i njezin globalni minimum. Svaku od prethodno opisanih metoda primjenjujemo na funkciju f uz početne aproksimacije $x_0 = 1.5$, $x_1 = 3$ i $x_2 = 6$. Kriterij zaustavljanja definiramo tako da algoritam staje ako je $|x_k - 4| < \varepsilon$, gdje je $\varepsilon > 0$ unaprijed zadana točnost. Pri tome za različite $\varepsilon \in [10^{-8}, 0.01]$ određujemo broj iteracija koji je potreban da bi se odgovarajući algoritam zaustavio. Početne aproksimacije odabrane su tako da sve metode konvergiraju.

Na Slici 2.8 prikazan je broj iteracija za *Metodu jedne točke-Newtonovu metodu* (NM), *Metodu dvije točke* (MDT) i *Metodu tri točke* (MTT) u ovisnosti o parametru ε . Kako je i očekivano, na Slici 2.8 vidimo da NM treba najmanje iteracija, MDT nešto više, dok MTT treba značajno najviše iteracija kako bi se zadovoljila zadana točnost.



Slika 2.8: Broj iteracija za *Newtonovu metodu* (NM), *Metodu dvije točke* (MDT) i *Metodu tri točke* (MTT)

Poglavlje 3

Jednodimenzionalna minimizacija strogo kvazikonveksnih funkcija

U Poglavlju 1.4 uveli smo pojam kvazikonveksne funkcije, a u Lemi 1.4, str.28 naveli smo operativni kriterij za ispitivanje kvazikonveksnosti neke funkcije. Dalje ćemo se baviti strogo kvazikonveksnim funkcijama. Kažemo da je funkcija $f: [a, b] \rightarrow \mathbb{R}$ strogo kvazikonveksna ako za sve $x, y \in [a, b]$, ($x \neq y$) vrijedi

$$f(\alpha x + (1 - \alpha)y) < \max\{f(x), f(y)\}, \quad \forall \alpha \in (0, 1),$$

a za svaki $\lambda \in \mathbb{R}$ nivo-skup \mathcal{D}_λ je konveksan. Točka $x^* \in \underset{x \in [a, b]}{\operatorname{argmin}} f(x)$ strogo kvazikonveksne funkcije točka je globalnog minimuma te funkcije.

3.1 Metoda tangenti za konveksnu derivabilnu funkciju

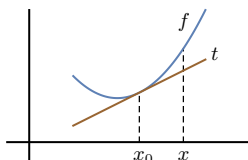
Za konveksnu derivabilnu funkciju može se konstruirati jednostavan algoritam za traženje njenog globalnog minimuma. Podsjetimo se kako je funkcija $f: [a, b] \rightarrow \mathbb{R}$ konveksna ako vrijedi

$$f(\alpha u + (1 - \alpha)v) \leq \alpha f(u) + (1 - \alpha)f(v), \quad \forall u, v \in [a, b], \quad \forall \alpha \in [0, 1]. \quad (3.1)$$

Detaljnije o pojmu konveksnosti pogledajte u t.1.3, str.24. Primijetite da u ovom poglavlju govorimo o *Metodi tangenti* za konveksnu diferencijabilnu funkciju jer je konveksna funkcija posebni slučaj kvazikonveksne funkcije.

Ovdje ćemo koristiti svojstvo konveksne derivabilne funkcije $f : [a, b] \rightarrow \mathbb{R}$ da njena tangenta $t(x) = f(x_0) + f'(x_0)(x - x_0)$ u proizvoljnoj točki $x_0 \in [a, b]$ leži ispod grafa te funkcije (vidi Sliku 3.1), tj. vrijedi tzv. „*gradijentna nejednakost*” (vidi također Lemu 1.2, str.25)

$$f(x) \geq f(x_0) + f'(x_0)(x - x_0), \quad \forall x \in [a, b].$$



Slika 3.1: Ilustracija gradijentne nejednakosti

Neka je $f : [a, b] \rightarrow \mathbb{R}$ konveksna derivabilna funkcija. Za fiksni $u_0 \in [a, b]$ definiramo funkciju

$$\Lambda(u, u_0) = f(u_0) + f'(u_0)(u - u_0), \quad u \in [a, b]. \quad (3.2)$$

Zbog svojstva (i) iz Leme 1.2 vrijedi $f(u) \geq \Lambda(u, u_0)$, $\forall u \in [a, b]$. Niže navodimo odgovarajući algoritam, čiju implementaciju možemo vidjeti u t.6.3.1, str.131.

Funkcija P_n neprekidna je po dijelovima linearna funkcija, čiji je graf sastavljen od dijelova tangenti na graf funkcije f . Zato ovu metodu zovemo **Metoda tangenti**.

Primjer 3.1. Metodu tangenti ilustrirat ćemo na primjeru minimizacije funkcije $f : [-1, 1] \rightarrow \mathbb{R}$, $f(x) = 13 + 18x + 37x^2 - 2x^3 - 6x^4 + \frac{1}{4}x^6$ koja predstavlja lokalnu LS-udaljenost točke do kubne parabole (vidi Primjer 1.2, str.7 i Primjer 6.5, str.132). Primijetite da je ovako definirana funkcija konveksna i derivabilna.

Na Slici 3.1 dana je ilustracija Algoritma 6 primjenom *Mathematica*-modula **MeTang** (vidi t.6.3.1, str.131), a u Tablici 3.1 može se detaljnije pratiti iterativni postupak. Globalni minimum postiže se u točki $u^* = -0.243094$.

Primjedba 3.1. Niz slomljenih pravaca (P_n) definiranih u Algoritmu 6 ima svojstva analogna svojstvima koja su navedena u Lemi 4.1, str.76 u *Metodi Pijavskog*:

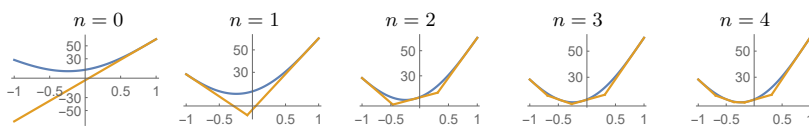
- (i) $P_n(u) \leq P_{n+1}(u)$, $\forall u \in [a, b]$ i $\forall n = 0, 1, \dots$ (**monotoni rast**)

Algoritam 6 (Metoda tangenti)**Input:** $f, u_0 \in [a, b]$

- 1: Definirati $\Lambda(u, u_0) = f(u_0) + f'(u_0)(u - u_0)$;
- 2: Odrediti $u_1 = \operatorname{argmin}_{u \in [a, b]} \Lambda(u, u_0)$ (Očigledno je $u_1 = a$ ili $u_1 = b$);
- 3: Provjeriti je li $f'(u_1) \neq 0$ [ako je $f'(u_1) = 0$, onda funkcija f u točki u_1 postiže minimum];
- 4: Definirati $\Lambda(u, u_1)$ i $P_1(u) = \max_{i=0,1} \Lambda(u, u_i)$;
- 5: Odrediti $u_2 = \operatorname{argmin}_{u \in [a, b]} P_1(u)$;
- 6: Uz pretpostavku da su čvorovi u_0, u_1, \dots, u_n ($n \geq 2$) poznati, provjeriti je li $f'(u_n) \neq 0$ [ako je $f'(u_n) = 0$, onda funkcija f postiže minimum u točki u_n];
- 7: Definirati

$$P_n(u) = \max_{i=0, \dots, n} \Lambda(u, u_i) = \max \{ \Lambda(u, u_n), P_{n-1}(u) \}; \quad (3.3)$$

- 8: Odrediti $u_{n+1} = \operatorname{argmin}_{u \in [a, b]} P_n(u)$



Slika 3.2: Ilustracija Metode tangenti za funkciju iz Primjera 3.1

n	1	2	3	4	5
u_n	-1	-0.078	-0.464	-0.266	-0.172
$f(u_n)$	28.25	11.825	12.545	10.838	11.006
$ u^* - u_n $	0.757	0.165	0.221	0.023	0.071

Tablica 3.1: Prvih 5 iteracija Metode tangenti na Primjeru 3.1

(ii) $P_n(u) \leq f(u), \forall u \in [a, b]$ i $\forall n = 0, 1, \dots$ (ograničenost odozgo)(iii) $P_n(u_i) = f(u_i), \forall i = 0, 1, \dots, n$ (podudaranje u čvorovima)

Niz (u_n) dobiven Algoritmom 6 konvergira prema točki globalnog minimuma funkcije f . Dokaz je analogan dokazu Teorema 4.1, str.78.

Primjedba 3.2. Opisana Metoda tangenti može se modificirati za traženje minimuma konveksne funkcije (ne nužno derivabilne), koja na intervalu $[a, b]$

postiže svoj minimum. Naime, konveksna funkcija $f : [a, b] \rightarrow \mathbb{R}$ neprekidna je na (a, b) i u svakoj točki $x \in (a, b)$ postoje konačni limesi

$$\lim_{h \rightarrow 0^+} \frac{f(x+h) - f(x)}{h} =: f'(x+), \quad \lim_{h \rightarrow 0^+} \frac{f(x) - f(x-h)}{h} =: f'(x-),$$

pri čemu je $f'(x-) \leq f'(x+)$ (vidi Zadatak 1.28). Zato u prethodno opisanoj *Metodi tangenti* možemo staviti

$$\Lambda(u, u_n) = f(u_n) + \gamma_n(u - u_n), \quad u \in [a, b], \quad (3.4)$$

gdje je γ_n proizvoljna konstanta koja zadovoljava nejednakost

$$f'(u_n-) \leq \gamma_n \leq f'(u_n+).$$

Primijetite da nejednakost $f(u) \geq \Lambda(u, u_n)$, $u \in [a, b]$ pri tome ostaje sačuvana (vidi Zadatak 1.30). U primjeni je najjednostavnije γ_n odrediti iz uvjeta

$$|\gamma_n| = \min\{|f'(u_n-)|, |f'(u_n+)|\}.$$

Ako se za neki n pokaže da vrijedi

$$f'(u_n-) \leq 0 \leq f'(u_n+),$$

onda je u_n točka minimuma (vidi Zadatak 1.31) i proces je završen.

U cilju izbjegavanja mogućih degeneracija $f'(a-) = -\infty$ ili $f'(b+) = +\infty$, dovoljno je iterativni proces započeti točkama $u_0 = a + h_0$, $u_1 = b - h_1$, gdje h_0 i h_1 treba izabrati tako da računalo prepozna $f'(a + h_0+)$ i $f'(b - h_1-)$.

Primjedba 3.3. Ideja *Metode tangenti* za traženje globalnog minimuma konveksne derivabilne funkcije iskoristit će se u t.4.2, str.74 za konstrukciju važne metode za traženje globalnog minimuma Lipschitz neprekidne funkcije.

Zadatak 3.1. Za funkciju $f : [1, 4] \rightarrow \mathbb{R}$, $f(x) = \frac{x}{2} + \frac{1}{x}$, odredite prve tri aproksimacije globalnog minimuma *Metodom tangenti* uz $u_0 = 1$.

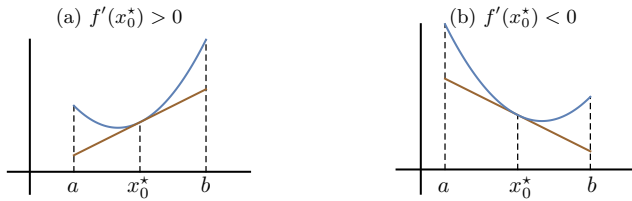
Zadatak 3.2. Provjerite može li se za minimizaciju funkcije $f : [-1, 1] \rightarrow \mathbb{R}$, $f(x) = x^3 + 3x^2$ primijeniti *Metoda tangenti* te ako može, provedite prve tri iteracije uz početnu aproksimaciju $u_0 = -1$.

3.2 Metoda polovljenja za kvazikonveksnu derivabilnu funkciju

Za kvazikonveksnu derivabilnu funkciju $f: [a, b] \rightarrow \mathbb{R}$ uz točnost $\epsilon > 0$ treba riješiti optimizacijski problem, tj. odrediti

$$x^* \in \operatorname{argmin}_{x \in [a, b]} f(x), \quad \text{takav da je} \quad f(x^*) = \min_{x \in [a, b]} f(x).$$

Najjednostavnija metoda je tzv. metoda polovljenja (bisekcije), koju opišujemo u nastavku. Najprije odredimo polovište x_0^* intervala $[a, b]$. Ako je $f'(x_0^*) = 0$, točka x_0^* rješenje je problema. Ako je $f'(x_0^*) > 0$, dalje razmatramo podinterval $[a, x_0^*]$ (vidi Sliku 3.3a). U protivnom, dalje razmatramo podinterval $[x_0^*, b]$ (vidi Sliku 3.3b). Proces se nastavlja tako dugo dok dužina podintervala ne postane manja od ϵ , ili dok derivacija funkcije f u nekoj točki ne postane jednaka nuli.



Slika 3.3: Metoda polovljenja za kvazikonveksnu derivabilnu funkciju

Primjedba 3.4. Primijetite da se ova metoda može interpretirati kao metoda polovljenja za traženje nultočke derivacije funkcije f (vidi [62]).

Neka je $[a_0, b_0] := [a, b]$ početni interval. Početnu aproksimaciju odredimo kao polovište tog intervala $x_0^* = \frac{1}{2}(a_0 + b_0)$. Zatim izaberemo novi podinterval

$$[a_1, b_1] := \begin{cases} [a_0, x_0^*], & f'(x_0^*) > 0 \\ [x_0^*, b_0], & f'(x_0^*) < 0 \end{cases}$$

i odredimo sljedeću aproksimaciju $x_1^* = \frac{1}{2}(a_1 + b_1)$. Primijetite da za apsolutnu pogrešku prve aproksimacije x_1^* vrijedi $\Delta x_1^* := |x_1^* - x^*| \leq \frac{1}{2}(b_1 - a_1)$.

Neka je općenito

$$x_k^* = \frac{1}{2}(a_k + b_k).$$

Vrijedi

$$|x^* - x_k^*| \leq \frac{1}{2}(b_k - a_k) = \left(\frac{1}{2}\right)^2(b_{k-1} - a_{k-1}) = \dots = \left(\frac{1}{2}\right)^{k+1}(b_0 - a_0). \quad (3.5)$$

Također, očigledno vrijedi

$$|x^* - x_{k+1}^*| \leq \frac{1}{2}|x^* - x_k^*|.$$

Na taj način dokazali smo sljedeći teorem o konvergenciji niza (x_k) , brzini konvergencije iterativnog postupka te odgovarajuće ocjene pogreške.

Teorem 3.1. *Neka je $f: [a, b] \rightarrow \mathbb{R}$ kvazikonveksna derivabilna funkcija. Niz (x_k^*) definiran na prethodno opisan način konvergira prema globalnom minimumu funkcije f linearnom brzinom i vrijedi ocjena pogreške*

$$|x^* - x_k^*| \leq \left(\frac{1}{2}\right)^{k+1}|b - a|.$$

Korištenjem ocjene (3.5) može se odrediti potreban broj iteracija za traženu točnost $\epsilon > 0$. Iz

$$\left(\frac{1}{2}\right)^{(k+1)}|b_0 - a_0| < \epsilon,$$

dobivamo

$$k > \frac{\ln \epsilon - \ln |b_0 - a_0|}{\ln 0.5} - 1.$$

Algoritam 7 (Metoda polovljenja za kvazikonveksnu derivabilnu funkciju)

Input: $f: [a, b] \rightarrow \mathbb{R}$;

Input: $a_0, b_0 \in [a, b]$; $\epsilon > 0$; $k=0$;

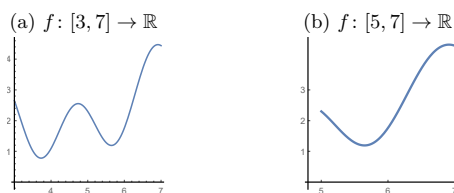
{Učitati funkciju f , početni interval $[a_0, b_0]$, točnost $\epsilon > 0$ i staviti $k = 0$;}

```

1: while  $b_k - a_k \geq \epsilon$ , do
2:    $x_p := \frac{1}{2}(a_k + b_k)$ 
3:   if  $f'(x_p) > 0$ , then
4:      $a_{k+1} = a_k$ ;  $b_{k+1} = x_p$ ;
5:   else
6:      $a_{k+1} = x_p$ ;  $b_{k+1} = b_k$ ;
7:   end if
8:    $k = k + 1$ ;
9: end while
```

Output: x_p

Primjer 3.2. *Metodom polovljenja potražimo lokalni minimum funkcije $f: [3, 7] \rightarrow \mathbb{R}$, $f(x) = \sin x + \sin 3x + \ln x + 1$ na intervalu $[5, 7]$ (vidi Sliku 3.4) s jednom točnom decimalom.*

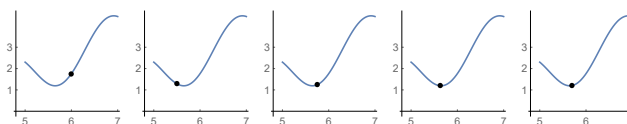


Slika 3.4: $f(x) = \sin x + \sin 3x + \ln x + 1$

Za $\epsilon = 0.05$ dobivamo $k > \frac{\ln(.05) - \ln |7-3|}{\ln 0.5} - 1 \approx 5.32$, što znači da ćemo nakon $k = 6$ iteracija dobiti aproksimaciju točnu na barem jednu decimalu (vidi Tablicu 3.2 i Sliku 3.5). Primijetite da je tražena točnost postignuta već u petoj iteraciji. Dobiva li se isti rezultat primjenom *Mathematica*-modula `NMinimize[]`?

k	a_k	b_k	x_k^*	$\frac{1}{2}(b_k - a_k)$	$f'(x_k^*)$
0	5	7	6	1	3.108
1	5	6	5.5	.5	-1.217
2	5.5	6	5.75	.25	.949
3	5.5	5.75	5.625	.125	-.210
4	5.625	5.75	5.6875	.0625	.360
5	5.625	5.6875	5.65625	.03125	0.071
6	5.625	5.65625	5.65625	.015625	0.071

Tablica 3.2: *Metoda polovljenja*: $f: [5, 7] \rightarrow \mathbb{R}$, $f(x) = \sin x + \sin 3x + \ln x + 1$



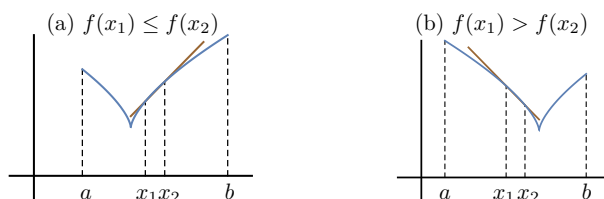
Slika 3.5: *Metoda polovljenja* za kvazikonveksnu derivabilnu funkciju

Zadatak 3.3. Za funkciju $f: [1, 4] \rightarrow \mathbb{R}$, $f(x) = \frac{x}{2} + \frac{1}{x}$ treba odrediti aproksimaciju globalnog minimizatora.

- (a) Primjenom *Metode polovljenja* odredite prve tri aproksimacije i ocijenite pogrešku dobivene aproksimacije.
- (b) Koliko iteracija *Metodom polovljenja* treba provesti da bi se dobila točnost globalnog minimuma na 4 decimalna mjesta?

3.3 Metoda polovljenja za nederivabilnu strogo kvazikonveksnu funkciju

Neka je $f: [a, b] \rightarrow \mathbb{R}$ strogo kvazikonveksna funkcija. Analogno prethodnoj metodi, konstruirat ćemo *Metodu polovljenja* za ovakvu funkciju. Uz to što funkcija f općenito nije derivabilna, ona na intervalu $[a, b]$ može imati i prekide prve vrste. Neka je $\delta > 0$ realan broj takav da je $b - a > \delta$.



Slika 3.6: *Metoda polovljenja* za strogo kvazikonveksnu nederivabilnu funkciju

Neka je $[a_0, b_0] := [a, b]$. Budući da u ovom slučaju općenito ne možemo koristiti derivaciju funkcije f u polovištu intervala $[a_0, b_0]$, u okolini polovišta simetrično ćemo izabrati dvije točke

$$\begin{aligned} x_1 &= \frac{1}{2}(a_0 + b_0) - \frac{\delta}{2}, \\ x_2 &= \frac{1}{2}(a_0 + b_0) + \frac{\delta}{2}, \end{aligned}$$

i promatrati koeficijent smjera odgovarajuće sekante

$$\frac{f(x_2) - f(x_1)}{\delta}.$$

Zato će nam kriterij za odabir podintervala u kojemu ostaje ležati točka minimuma x^* biti sljedeći (vidi Sliku 3.6):

$$[a_1, b_1] := \begin{cases} [a_0, x_2], & f(x_1) \leq f(x_2) \\ [x_1, b_0], & f(x_1) > f(x_2) \end{cases}.$$

Širina novog podintervala je

$$b_1 - a_1 (= x_2 - a_0 = b_0 - x_1) = \frac{1}{2}(b_0 - a_0) + \frac{1}{2}\delta.$$

Prvu aproksimaciju x_1^* možemo definirati u polovištu intervala $[a_1, b_1]$

$$x_1^* = \frac{1}{2}(a_1 + b_1).$$

Ocjena apsolutne pogreške prve aproksimacije je

$$\Delta x_1^* := |x_1^* - x^*| \leq \frac{1}{2}(b_1 - a_1) = \frac{1}{2^2}(b_0 - a_0) + \frac{1}{2^2}\delta,$$

što ćemo zapisati u obliku

$$\Delta x_1^* := |x_1^* - x^*| \leq \frac{1}{2^2}(b_0 - a_0) + \frac{1}{2}\left(1 - \frac{1}{2}\right)\delta,$$

Ponavljajući postupak nadalje definiramo dvije simetrične točke u okolini polovišta intervala $[a_1, b_1]$

$$\begin{aligned} x_3 &= \frac{1}{2}(a_1 + b_1) - \frac{1}{2}\delta, \\ x_4 &= \frac{1}{2}(a_1 + b_1) + \frac{1}{2}\delta, \end{aligned}$$

i novi podinterval

$$[a_2, b_2] := \begin{cases} [a_1, x_4], & f(x_3) \leq f(x_4) \\ [x_3, b_1], & f(x_3) > f(x_4) \end{cases},$$

u kojemu ostaje točka minimuma x^* funkcije f . Širina novog podintervala je

$$b_2 - a_2 = \frac{1}{2}(b_1 - a_1) + \frac{1}{2}\delta = \frac{1}{2^2}(b_0 - a_0) + \frac{1}{2^2}\delta + \frac{1}{2}\delta.$$

Novu aproksimaciju x_2^* odredit ćemo u polovištu intervala $[a_2, b_2]$

$$x_2^* = \frac{1}{2}(a_2 + b_2),$$

čiju apsolutnu pogrešku možemo ocijeniti s

$$\Delta x_2^* := |x_2^* - x^*| \leq \frac{1}{2}(b_2 - a_2) = \frac{1}{2^3}(b_0 - a_0) + \frac{1}{2} \cdot \frac{3}{4}\delta,$$

što ćemo zapisati kao

$$\Delta x_2^* := |x_2^* - x^*| \leq \frac{1}{2^3}(b_0 - a_0) + \frac{1}{2}\left(1 - \frac{1}{2^2}\right)\delta.$$

Općenito, neka je poznat interval $[a_{k-1}, b_{k-1}]$ širine

$$b_{k-1} - a_{k-1} = \frac{1}{2}(b_{k-2} - a_{k-2}) + \frac{1}{2}\delta, \quad k \geq 2,$$

u kojemu leži točka minimuma x^* . Kao i ranije, definiramo dvije simetrične točke u okolini polovišta

$$\begin{aligned} x_{2k-1} &= \frac{1}{2}(a_{k-1} + b_{k-1}) - \frac{1}{2}\delta, \\ x_{2k} &= \frac{1}{2}(a_{k-1} + b_{k-1}) + \frac{1}{2}\delta, \end{aligned}$$

i sljedeći podinterval

$$[a_k, b_k] := \begin{cases} [a_{k-1}, x_{2k}], & f(x_{2k-1}) \leq f(x_{2k}) \\ [x_{2k-1}, b_{k-1}], & f(x_{2k-1}) > f(x_{2k}) \end{cases}$$

širine

$$b_k - a_k = \frac{1}{2}(b_{k-1} - a_{k-1}) + \frac{\delta}{2} = \frac{1}{2^k}(b_0 - a_0) + \frac{1}{2} \left(1 + \dots + \frac{1}{2^{k-1}}\right) \delta.$$

Sljedeću aproksimaciju x_k^* definirat ćemo s

$$x_k^* := \frac{1}{2}(a_k + b_k).$$

Apsolutnu pogrešku možemo ocijeniti s

$$\begin{aligned} \Delta x_k^* := |x_k^* - x^*| &\leq \frac{1}{2}(b_k - a_k) = \frac{1}{2^{k+1}}(b_0 - a_0) + \frac{1}{2^2} \left(1 + \dots + \frac{1}{2^{k-1}}\right) \delta \\ &= \frac{1}{2^{k+1}}(b_0 - a_0) + \frac{1}{2} \left(1 - \frac{1}{2^k}\right) \delta. \end{aligned}$$

Za $k \rightarrow +\infty$ ocjena apsolutne pogreške konvergira prema $\frac{\delta}{2}$. Dakle, nakon dovoljnog broja iteracija dobivena aproksimacija može odstupati od točke minimuma za najviše $\frac{\delta}{2}$.

Metodu nazivamo *Metoda polovljenja* jer u svakom koraku interval dijelimo gotovo na dvije polovine – čim je δ manji, time je dijeljenje intervala bliže raspolavljanju. Naravno, ova metoda može se primijeniti i na proizvoljnu strogo konveksnu funkciju (vidi Primjedbu 1.4, str.28).

Algoritam 8 (Metoda polovljenja za kvazikonveksnu funkciju)

Input: $f: [a, b] \rightarrow \mathbb{R}$;

Input: $a, b; \epsilon > 0; 0 < \delta < b - a; k = 0$;

{Učitati funkciju f , početni interval $[a, b]$, točnost $\epsilon > 0$ i staviti $k = 0$;}

1: $x_p = (a + b)/2$;

2: **while** $b - a \geq \epsilon$, **do**

3: $k = k + 1$;

4: $x_1 = x_p - \frac{\delta}{2}$; $x_2 = x_p + \frac{\delta}{2}$;

5: **if** $f(x_1) \leq f(x_2)$, **then**

6: $b = x_2$;

7: **else**

8: $a = x_1$;

9: **end if**

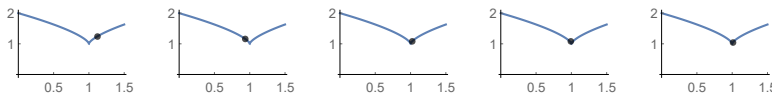
10: $x_p = (a + b)/2$;

11: **end while**

Output: x_p

Primjedba 3.5. Primijetite da se ova metoda može interpretirati kao metoda polovljenja gdje je derivacija zamijenjena s podijeljenom razlikom (vidi [62]).

Primjer 3.3. Metodu polovljenja ilustrirat ćemo na primjeru minimizacije funkcije $f: [-1, 2] \rightarrow \mathbb{R}$, $f(x) = 1 + \sqrt[3]{(x-1)^2}$. Na Slici 3.7 prikazano je nekoliko iteracija.



Slika 3.7: Ilustracije Metode polovljenja na funkciji iz Primjera 3.3

Zadatak 3.4. Je li $f: [-2, 2] \rightarrow \mathbb{R}$, $f(x) = 1 - e^{-|x-1|}$ kvazikonveksna funkcija? Ako je, Metodom polovljenja odredite aproksimaciju njezine točke globalnog minimuma.

Zadatak 3.5. Lokalni minimumi funkcije iz Primjera 1.2, str.7 postižu se u točkama

$$\begin{aligned} x_1^* &\approx -3.217, & x_2^* &\approx 0.243, & x_3^* &\approx 3.617, \\ f(x_1^*) &\approx 6.251, & f(x_2^*) &\approx 3.289, & f(x_3^*) &\approx 0.618. \end{aligned}$$

Definirajte intervale u kojima ćete Metodom polovljenja odrediti pojedine točke lokalnih minimuma.

Zadatak 3.6. Metodom polovljenja odredite prve četiri aproksimacije minimuma funkcije $f: [0, 3] \rightarrow \mathbb{R}$, $f(x) = |x-1| + 2$ uz $\delta = 0.2$. Ocijenite pogrešku dobivene aproksimacije. Kolika je stvarna apsolutna pogreška?

Zadatak 3.7. Za dani $\epsilon > 0$ i parametar $\delta > 0$ odredite broj iteracija k koji je potreban da Metoda polovljenja daje točku globalnog minimuma x^* kvazikonveksne funkcije $f: [a, b] \rightarrow \mathbb{R}$ s točnošću $\frac{\delta}{2} + \epsilon$. Kako treba odabrati $\delta > 0$ i k da bismo postigli točnost na dvije decimale? U tom smislu analizirajte Primjer 3.3.

Zadatak 3.8. Neka je $f: [a, b] \rightarrow \mathbb{R}$ strogo kvazikonveksna funkcija. Definirajte modifikaciju Metode polovljenja za minimizaciju funkcije f u kojoj je vrijednost parametra δ moguće mijenjati u svakoj iteraciji $\delta = \delta_k$. Izvedite ocjenu apsolutne pogreške i dokažite konvergenciju metode uz dodatne pretpostavke na niz (δ_k) . Što možete reći o brzini konvergencije metode?

3.4 Metoda zlatnog reza

Neka je ℓ dužina duljine 1. Točka (odnosno realni broj) $.5 < x < 1$ dijeli dužinu ℓ u omjeru zlatnog reza ako se omjer duljine cijele dužine ℓ u odnosu na veći dio x odnosi kao omjer većeg dijela x u odnosu na manji dio $1-x$,

tj.

$$\frac{1}{x} = \frac{x}{1-x}.$$

Pozitivno rješenje ove jednadžbe je $x = \frac{\sqrt{5}-1}{2} \approx 0.618$, pri čemu broj $\frac{1}{x} = \frac{\sqrt{5}+1}{2} \approx 1.61812$ nazivamo omjer zlatnog reza.



Slika 3.8: Dijeljenje dužine ℓ u omjeru zlatnog reza

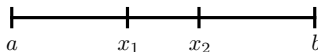
Zadatak 3.9. Pokažite da i broj $1-x = \frac{3-\sqrt{5}}{2} \approx 0.382$ također dijeli dužinu ℓ u omjeru zlatnog reza.

Neka je $f: [a, b] \rightarrow \mathbb{R}$ strogo kvazikonveksna funkcija. Niz aproksimacija točke globalnog minimuma $x^* \in [a, b]$ funkcije f definirat ćemo na sljedeći način. Formalno označimo $[a_0, b_0] := [a, b]$.

U cilju definiranja prve aproksimacije najprije definirajmo točke $x_1, x_2 \in [a_0, b_0]$, $x_1 < x_2$, na sljedeći način

$$x_1 = a_0 + (1-x)(b_0 - a_0) = a_0 + \frac{3-\sqrt{5}}{2}(b_0 - a_0), \quad (3.6)$$

$$x_2 = a_0 + x(b_0 - a_0) = a_0 + \frac{\sqrt{5}-1}{2}(b_0 - a_0). \quad (3.7)$$



Slika 3.9: Izbor točaka x_1, x_2

Točka x_1 čini zlatni rez intervala $[a_0, b_0]$. Naime,

$$\frac{b_0 - a_0}{b_0 - x_1} = \frac{b_0 - x_1}{x_1 - a_0}.$$

Slično se može pokazati da i točka x_2 čini zlatni rez intervala $[a_0, b_0]$ jer vrijedi

$$\frac{b_0 - a_0}{x_2 - a_0} = \frac{x_2 - a_0}{b_0 - x_2}.$$

Točke x_1 i x_2 poslužit će nam slično kao u t.3.3, str.60 za definiranje intervala $[a_1, b_1]$ i prve aproksimacije x_1^*

$$[a_1, b_1] = \begin{cases} [a_0, x_2], & f(x_1) \leq f(x_2), \\ [x_1, b_0], & f(x_1) > f(x_2), \end{cases} \quad \& \quad \begin{cases} x_1^* := x_1, \\ x_1^* := x_2. \end{cases}$$

Širina intervala $[a_1, b_1]$ je

$$b_1 - a_1 = \begin{cases} x_2 - a_0 \stackrel{(3.7)}{=} \frac{\sqrt{5}-1}{2}(b_0 - a_0) \\ b_0 - x_1 \stackrel{(3.6)}{=} x(b_0 - a_0) = \frac{\sqrt{5}-1}{2}(b_0 - a_0) \end{cases}.$$

Pogrešku prve aproksimacije procijenit ćemo slično kao u t.3.3

$$\Delta x_1^* := |x_1^* - x^*| \leq \left(\frac{\sqrt{5}-1}{2}\right)(b_1 - a_1) = \left(\frac{\sqrt{5}-1}{2}\right)^2(b_0 - a_0).$$

Također primijetimo da je

$$x_2 - x_1 = (\sqrt{5} - 2)(b_0 - a_0).$$

Ponavljajući postupak, definiramo točke x_3, x_4 u intervalu $[a_1, b_1]$

$$x_3 = a_1 + \frac{3-\sqrt{5}}{2}(b_1 - a_1),$$

$$x_4 = a_1 + \frac{\sqrt{5}-1}{2}(b_1 - a_1).$$

Primjedba 3.6. Označimo li $\phi = \frac{\sqrt{5}-1}{2}$, možemo uočiti da ako je $f(x_1) \leq f(x_2)$, onda zbog $\phi^2 = 1 - \phi = \frac{3-\sqrt{5}}{2}$ vrijedi

$$\begin{aligned} x_4 &= a_0 + \phi(x_2 - a_0) = a_0 + \phi(a_0 + \phi(b_0 - a_0) - a_0) \\ &= a_0 + \phi^2(b_0 - a_0) = x_2. \end{aligned}$$

S druge strane, ako je $f(x_1) > f(x_2)$, onda

$$\begin{aligned} x_3 &= x_1 + (1 - \phi)(b_0 - x_1) \\ &= a_0 + (1 - \phi)(b_0 - a_0) + (1 - \phi)(b_0 - a_0 - (1 - \phi)(b_0 - a_0)) \\ &= a_0 + (1 - \phi^2)(b_0 - a_0) = x_1. \end{aligned}$$

Stoga, u svakoj je iteraciji potrebno izračunati samo jednu novu točku.

Pokažite da i točka x_3 i točka x_4 dijele segment $[a_1, b_1]$ u omjeru zlatnog reza.

Pomoću točaka x_3, x_4 definirat ćemo novi podinterval $[a_2, b_2]$ i drugu aproksimaciju x_2^*

$$[a_2, b_2] = \begin{cases} [a_1, x_4], & f(x_3) \leq f(x_4), \\ [x_3, b_1], & f(x_3) > f(x_4), \end{cases} \quad \& \quad \begin{cases} x_2^* := x_3, \\ x_2^* := x_4. \end{cases}$$

Širina intervala $[a_2, b_2]$ je

$$b_2 - a_2 (= x_4 - a_1 = b_1 - x_3) = \frac{\sqrt{5}-1}{2}(b_1 - a_1) = \left(\frac{\sqrt{5}-1}{2}\right)^2 (b_0 - a_0).$$

Pogreška druge aproksimacije x_2^* procijenjena je s

$$\Delta x_2^* := |x_2^* - x^*| \leq \left(\frac{\sqrt{5}-1}{2}\right)(b_2 - a_2) = \left(\frac{\sqrt{5}-1}{2}\right)^3 (b_0 - a_0).$$

Također,

$$x_4 - x_3 = (\sqrt{5} - 2)(b_1 - a_1) = \frac{\sqrt{5}-1}{2}(\sqrt{5} - 2)(b_0 - a_0).$$

Općenito, pretpostavimo da je poznat interval $[a_{k-1}, b_{k-1}]$ u kojemu leži točka minimuma x^* . Kao i ranije, definiramo dvije točke koje dijele interval $[a_{k-1}, b_{k-1}]$ u omjeru zlatnog reza

$$\begin{aligned} x_{2k-1} &= a_{k-1} + \frac{3-\sqrt{5}}{2}(b_{k-1} - a_{k-1}), \\ x_{2k} &= a_{k-1} + \frac{\sqrt{5}-1}{2}(b_{k-1} - a_{k-1}). \end{aligned}$$

Pomoću točaka x_{2k-1}, x_{2k} definirat ćemo novi podinterval $[a_k, b_k]$ i novu aproksimaciju x_k^*

$$[a_k, b_k] = \begin{cases} [a_{k-1}, x_{2k}], & f(x_{2k-1}) \leq f(x_{2k}), \\ [x_{2k}, b_{k-1}], & f(x_{2k-1}) > f(x_{2k}), \end{cases} \quad \& \quad \begin{cases} x_k^* := x_{2k-1}, \\ x_k^* := x_{2k}. \end{cases}$$

Širina podintervala $[a_k, b_k]$ je

$$b_k - a_k (= x_{2k} - a_k = b_k - x_{2k-1}) = \frac{\sqrt{5}-1}{2}(b_{k-1} - a_{k-1}) = \left(\frac{\sqrt{5}-1}{2}\right)^k (b_0 - a_0),$$

a apsolutna pogreška k -te aproksimacije zadana je s

$$\Delta x_k^* := |x_k^* - x^*| \leq \left(\frac{\sqrt{5}-1}{2}\right)(b_k - a_k) = \left(\frac{\sqrt{5}-1}{2}\right)^{k+1} (b_0 - a_0),$$

te je

$$x_{2k} - x_{2k-1} = (\sqrt{5} - 2)(b_{k-1} - a_{k-1}) = \left(\frac{\sqrt{5}-1}{2}\right)^{k-1} (\sqrt{5} - 2)(b_0 - a_0).$$

Primijetite da za $k \rightarrow +\infty$ niz apsolutnih pogrešaka (Δx_k^*) konvergira prema nuli te da isto tako niz $(x_{2k} - x_{2k-1})$ konvergira prema nuli. To znači da primjenom *Metode zlatnog reza* za traženje globalnog minimizatora strogo kvazikonveksne funkcije f (za razliku od *Metode polovljenja*) možemo postići

proizvoljnu točnost aproksimacije. Može se također pokazati¹ da je pri tome izbor točaka $\{x_{2k-1}, x_{2k}\}$ gotovo optimalan.

Više o metodi vidi u [25, 46, 69].

Algoritam 9 (Metoda zlatnog reza)

Input: $f: [a, b] \rightarrow \mathbb{R}$;

Input: $a, b; \epsilon > 0; \tau = \frac{\sqrt{5}-1}{2}; k = 0;$

{Učitati funkciju f , početni interval $[a, b]$, točnost $\epsilon > 0$ i staviti $k = 0$;}

1: **while** $b - a \geq \epsilon$, **do**

2: $k = k + 1$;

3: $x_1 = a + \frac{3-\sqrt{5}}{2}(b-a); \quad x_2 = a + \frac{\sqrt{5}-1}{2}(b-a);$

4: **if** $f(x_1) \leq f(x_2)$, **then**

5: $b = x_2; x^* = x_1$

6: **else**

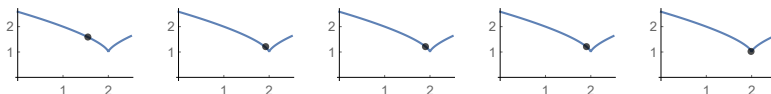
7: $a = x_1; x^* = x_2$

8: **end if**

9: **end while**

Output: x^*

Primjer 3.4. Metodu zlatnog reza ilustrirat ćemo na primjeru minimizacije funkcije $f: [0, 2.5] \rightarrow \mathbb{R}$, $f(x) = 1 + \sqrt[3]{(x-2)^2}$. Na Slici 3.10 prikazano je nekoliko iteracija.



Slika 3.10: Traženje globalnog minimuma funkcije iz Primjera 3.4 Metodom zlatnog reza

Zadatak 3.10. Neka je $f: [a, b] \rightarrow \mathbb{R}$ strogo kvazikonveksna funkcija. Odredite broj iteracija k koji je potreban da bi se Metodom zlatnog reza odredila aproksimacija točke globalnog minimuma s točnošću $\epsilon > 0$. Koliko treba iteracija da bi se Metodom zlatnog reza pronašla točka globalnog minimuma funkcije iz Primjera 3.4 s točnošću na dvije decimale?

Zadatak 3.11. Metodom zlatnog reza odredite prve četiri aproksimacije minimuma funkcije $f: [0, 3] \rightarrow \mathbb{R}$, $f(x) = |x-1|+2$. Ocijenite pogrešku dobivene aproksimacije.

¹Iz povijesnih razloga navodimo originalnu referencu:

Ф.П.Васи́лев, *Лекции по методам решения экстремальных задач*, Издательство Московского университета, Москва, 1974.

Kolika je stvarna apsolutna pogreška? Koliko je iteracija potrebno za točnost $\varepsilon = 0.0005$?

Zadatak 3.12. Neka je $f: [a, b] \rightarrow \mathbb{R}$ konveksna derivabilna funkcija. Usporedite broj iteracija k koji je potreban za određivanje točke globalnog minimuma funkcije f s točnošću $\epsilon > 0$ pomoću *Metode polovljenja* i pomoću *Metode zlatnog reza*. Usporedbu provedite općenito i također na primjeru funkcije iz Primjera 3.4.

Zadatak 3.13. Neka je $f: [a, b] \rightarrow \mathbb{R}$ strogo kvazikonveksna funkcija. Izbor intervala koji sadrži točku minimuma funkcije f u *Metodi zlatnog reza* temelji se na izboru točaka

$$\begin{aligned}x_{2k-1} &= a_{k-1} + (1 - \tau)(b_{k-1} - a_{k-1}), \\x_{2k} &= a_{k-1} + \tau(b_{k-1} - a_{k-1}),\end{aligned}$$

gdje je $\tau = \frac{\sqrt{5}-1}{2}$. Za proizvoljni $\tau \in (1/2, 1)$, koliko iznosi stopa smanjenja duljine intervala

$$\frac{b_k - a_k}{b_{k-1} - a_{k-1}}?$$

Izradite *Mathematica*-modul za *Metodu zlatnog reza* s proizvoljnim parametrom $\tau \in (\frac{1}{2}, 1)$. Na primjeru funkcije $f: [0, 2.5] \rightarrow \mathbb{R}$, $f(x) = 1 + \sqrt[3]{(x-2)^2}$ testirajte brzinu konvergencije metode za različite izbore parametra τ .

Poglavlje 4

Jednodimenzionalna globalna optimizacija

4.1 Lipschitz-neprekidne funkcije

U ovom odjeljku razmatrat ćemo jednu metodu traženja globalnog minimuma realne neprekidne funkcije jedne varijable definirane na nekom intervalu $[a, b]$. Od funkcije se neće zahtijevati derivabilnost, već samo ograničeni rast na tom intervalu. Metodu su gotovo istovremeno, ali nezavisno, razradili S. A. Pijavskij¹ i B. O. Shubert².

Definicija 4.1. *Kažemo da je funkcija $f: \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}$ Lipschitz-neprekidna (zadovoljava Lipschitzov uvjet) s konstantom $L > 0$ na \mathcal{D} i pišemo $f \in \text{Lip}_L \mathcal{D}$ ako za sve $x, y \in \mathcal{D}$ vrijedi*

$$|f(x) - f(y)| \leq L|x - y|. \quad (4.1)$$

Konstantu $L > 0$ nazivamo Lipschitzova konstanta.

Primjer 4.1. *Funkcija $f: \mathbb{R} \rightarrow \mathbb{R}$, $f(x) = |x|$ je Lipschitz-neprekidna s konstantom $L = 1$ na \mathbb{R} jer je $||x| - |y|| \leq 1 \cdot |x - y|$ za sve $x, y \in \mathbb{R}$.*

¹Iz povijesnih razloga navodimo originalne reference:

С.А. Пиявский, Один алгоритм отыскания абсолютного экстремума функции, Ж. вычисл. матем. и матем. физ. **12**(1972), 888-896. http://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=zvmmf&paperid=6654&option_lang=eng

Vidi također:

Ф.П. Васильев, Лекции по методам решения экстремальных задач, Издательство Московского университета, Москва, 1974.

²Bruno O. Shubert, *A sequential method seeking the global maximum of a function*, SIAM Journal on Numerical Analysis, **9**(1972), 379-388

Primjer 4.2. *Odredimo Lipschitzovu konstantu funkcije $f: [-3, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 6x + 2$.*

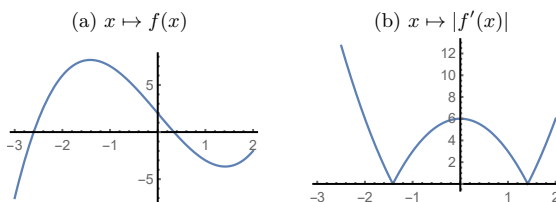
Primijetite da je ova funkcija derivabilna u svakoj točki segmenta $[-3, 2]$ i da za nju vrijedi Lagrangeov teorem o srednjoj vrijednosti, tj. za proizvoljni $x, y \in [-3, 2]$, $x < y$, postoji $c \in (x, y)$ takav da je

$$f(x) - f(y) = f'(c)(x - y).$$

Zbog neprekidnosti funkcije f' na segmentu $[-3, 2]$ sukladno Weierstrasseovom teoremu (vidi [75]) postoji $L = \max_{x \in [a, b]} |f'(x)|$ iz čega slijedi

$$|f(x) - f(y)| \leq L|x - y|, \quad \forall x, y \in [-3, 2].$$

U ovom slučaju iz Slike 4.1 vidi se da je najmanja moguća vrijednost za Lipschitzovu konstantu $L = f'(-3) = 21$, iako i svaki drugi broj veći od 21 zadovoljava (4.1).



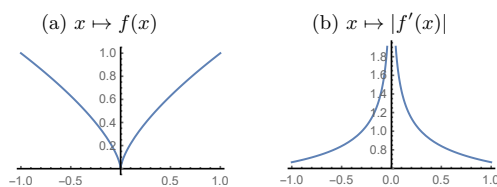
Slika 4.1: $f: [-3, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 6x + 2$ je Lipschitz-neprekidna funkcija

U nastavku pretpostavljamo da je područje definicije promatrane funkcije f interval realnih brojeva $[a, b]$. Za $x \neq y$ Lipschitzov uvjet (4.1) možemo zapisati u obliku

$$-L \leq \frac{f(x) - f(y)}{x - y} \leq L \quad \forall x, y \in [a, b],$$

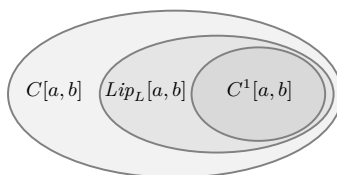
odakle se vidi da je relativna promjena (kvocijent diferencija) funkcije f ograničena između $-L$ i L . Ako x fiksiramo, a y pustimo prema x , onda možemo zaključiti da ako postoji derivacija funkcije f , ona je ograničena između $-L$ i L .

Primjer 4.3. *Promotrajmo funkciju $f: [-1, 1] \rightarrow \mathbb{R}$, $f(x) = \sqrt[3]{x^2}$. Kao što se može vidjeti na Slici 4.2, derivacija (brzina pada) ove funkcije u okolini 0 raste u beskonačnost. Zato ne postoji $L > 0$ takav da vrijedi (4.1), pa ova funkcija nije Lipschitz-neprekidna na $[-1, 1]$.*



Slika 4.2: $f: [-1, 1] \rightarrow \mathbb{R}$, $f(x) = \sqrt[3]{x^2}$ nije Lipschitz-neprekidna funkcija

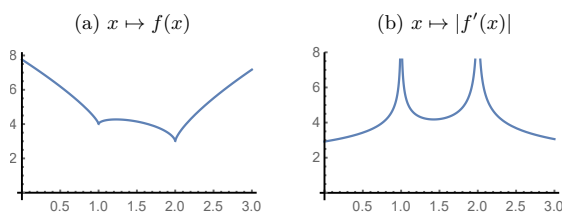
Iako je određivanje Lipschitzove konstante općenito vrlo složen problem (vidi primjerice [58, 77]), u slučaju da promatrana funkcija f ima konačnu derivaciju u svakoj točki intervala $[a, b]$, problem se može riješiti relativno jednostavno (kao u Primjerima 4.1, 4.2 ili 4.5).



Slika 4.3: Skup neprekidnih, Lipschitz-neprekidnih i neprekidno derivabilnih funkcija na $[a, b]$

Primjedba 4.1. Svaka Lipschitz-neprekidna funkcija neprekidna je funkcija, ali neprekidna funkcija ne mora biti Lipschitz-neprekidna (vidi Primjer 4.4). Nadalje, ako je $f \in C^1[a, b]$, onda je $f \in Lip_L[a, b]$ (Lagrangeov teorem o srednjoj vrijednosti), ali obrat ne vrijedi (vidi Primjere 4.1 i 4.5). Dakle, mogli bismo reći da je Lipschitz-neprekidna funkcija neprekidna funkcija, a ako ima derivaciju, ona mora biti ograničena.

Ako je $f: [a, b] \rightarrow \mathbb{R}$ neprekidna funkcija, koja ima po dijelovima neprekidnu ograničenu derivaciju, onda Lipschitzovu konstantu $L > 0$ možemo odrediti primjenom Lagrangeovog teorema o srednjoj vrijednosti (vidi Primjer 4.5). Preciznije, neka je $f: [a, b] \rightarrow \mathbb{R}$ neprekidna po dijelovima glatka funkcija, u smislu da je f derivabilna, osim eventualno u konačno mnogo točaka $x_1 < x_2 < \dots < x_n$, te da je f' neprekidna na intervalima (x_j, x_{j+1}) , $j = 1, \dots, n-1$. Ako je $|f'(x)| \leq L$ za sve $x \in [a, b] \setminus \{x_1, \dots, x_n\}$, tada je $f \in Lip_L[a, b]$.



Slika 4.4: Funkcija $f: [0, 3] \rightarrow \mathbb{R}$, $f(x) = 1 + 2\sqrt[3]{(x-1)^2} + 3\sqrt[3]{(x-2)^2}$ nije Lipschitz-neprekidna funkcija na $[0, 3]$

Primjer 4.4. Odredimo Lipschitzovu konstantu funkcije $f: [0, 3] \rightarrow \mathbb{R}$, $f(x) = 1 + 2\sqrt[3]{(x-1)^2} + 3\sqrt[3]{(x-2)^2}$ (vidi Sliku 4.4). Njena derivacija

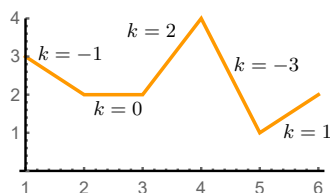
$$f'(x) = \frac{4}{3\sqrt[3]{x-1}} + \frac{2}{\sqrt[3]{x-2}}$$

nije definirana u točkama iz $\{1, 2\}$, a Lipschitzova konstanta $L > 0$ ne postoji.

Primjer 4.5. Odredimo Lipschitzovu konstantu funkcije $f: [1, 6] \rightarrow \mathbb{R}$,

$$f(x) = \begin{cases} -x + 4, & 1 \leq x \leq 2, \\ 2, & 2 \leq x \leq 3, \\ 2x - 4, & 3 \leq x \leq 4, \\ -3x + 16, & 4 \leq x \leq 5, \\ x - 4, & 5 \leq x \leq 6, \end{cases}$$

čiji je graf prikazan na Slici 4.5. Očigledno je $L = 3$.



Slika 4.5: Lipschitz-neprekidna funkcija s konstantom $L = 3$

Primjedba 4.2. Definicija Lipschitz-neprekidne funkcije lako se može proširiti na funkciju više varijabli $f: \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$. Takva je funkcija Lipschitz-neprekidna (zadovoljava Lipschitzov uvjet) ako postoji konstanta $L > 0$ tako

da za sve $x, y \in \mathcal{D}$ vrijedi

$$|f(x) - f(y)| \leq L\|x - y\|,$$

gdje je $\|\cdot\|$ neka norma. Uočite da je definicija neovisna o izboru norme $\|\cdot\|$ s obzirom na to da su sve norme na \mathbb{R}^n ekvivalentne.

Zadatak 4.1. Provjerite jesu li sljedeće funkcije Lipschitz-neprekidne te ako jesu, odredite Lipschitzovu konstantu.

(a) $f_1: [0, 2] \rightarrow \mathbb{R}, f_1(x) = e^x,$

(b) $f_2: [0, 2] \rightarrow \mathbb{R},$

$$f_2(x) = \begin{cases} 5 - x, & 0 \leq x \leq 3, \\ 4 - \frac{1}{2}(x - 5)^2, & 3 < x \leq 7, \\ \frac{1}{2}(x - 9)^2, & 7 < x \leq 9, \end{cases}$$

(c) $f_3: [-1, 1] \rightarrow \mathbb{R}, f_3(x) = 1 + \sqrt[3]{x^2}.$

Zadatak 4.2. Neka je $f_1 \in Lip_{L_1}[a, b]$ i $f_2 \in Lip_{L_2}[a, b]$. Pokažite da su tada i sljedeće funkcije Lipschitz-neprekidne i odredite Lipschitzovu konstantu:

(a) $f_1 + f_2,$

(b) $cf_1, c \in \mathbb{R},$

(c) $c_1f_1 + c_2f_2, c_1, c_2 \in \mathbb{R},$

(d) $f_1 \cdot f_2,$

(e) $f_1 \circ f_2,$ pod pretpostavkom da je dobro definirana.

Zadatak 4.3. Neka je $f: [a, b] \rightarrow \mathbb{R}$ po dijelovima Lipschitz-neprekidna funkcija u smislu da postoje $a = x_0 < x_1 < \dots < x_n = b$ takvi da je f Lipschitz-neprekidna na intervalima $\langle x_i, x_{i+1} \rangle, i = 0, \dots, n-1$. Pokažite da je tada f Lipschitz-neprekidna te odredite Lipschitzovu konstantu.

Zadatak 4.4. Dokažite da konveksna funkcija $f: [a, b] \rightarrow \mathbb{R}$ zadovoljava Lipschitzov uvjet na svakom segmentu $[\alpha, \beta] \subset [a, b]$. Korištenjem funkcije $f(x) = -\sqrt{1-x^2}, |x| \leq 1$, pokažite da u općem slučaju nije dobro birati $\alpha = a, \beta = b$.

Uputa. Korištenjem nejednakosti (1.24) i (1.25) pokažite da je

$$f'(\alpha+) \leq \frac{f(u)-f(v)}{u-v} \leq f'(\beta-), \quad \forall u, v \in [\alpha, \beta].$$

Zadatak 4.5. Neka je $f_\lambda, \lambda \in \Lambda$ proizvoljna familija funkcija iz $Lip_L[a, b]$. Provjerite jesu li tada i funkcije $\inf_{\lambda \in \Lambda} f_\lambda$ i $\sup_{\lambda \in \Lambda} f_\lambda$ Lipschitz-neprekidne na $[a, b]$ te odredite Lipschitzove konstante.

Zadatak 4.6. Provjerite je li funkcija $f: [0, 1] \times [0, 2] \rightarrow \mathbb{R}, f(x_1, x_2) = x_1x_2^2$ Lipschitz-neprekidna te ako jest, odredite Lipschitzovu konstantu.

Zadatak 4.7. Neka je $\mathcal{D} = [a_1, b_1] \times \cdots \times [a_n, b_n]$ i $f: \mathcal{D} \rightarrow \mathbb{R}$ funkcija. Dokažite:

- a) f je Lipschitz-neprekidna ako i samo ako postoji konstanta $L > 0$ takva da je f Lipschitz-neprekidna s konstantom L u svakoj varijabli. Preciznije, za svaki $i = 1, \dots, n$ i za proizvoljne $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$ funkcija

$$x \mapsto f(x_1, \dots, x_{i-1}, x, x_{i+1}, \dots, x_n)$$

je Lipschitz-neprekidna na $[a_i, b_i]$ s konstantom L .

- b) Ako su sve parcijalne derivacije funkcije f definirane i ograničene na \mathcal{D} , tada je f Lipschitz-neprekidna.

4.2 Pijavskijeva metoda slomljenih pravaca

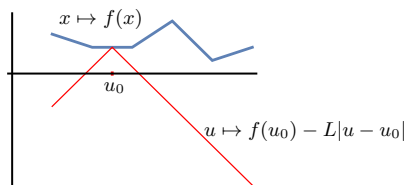
Pretpostavimo da je $f: [a, b] \rightarrow \mathbb{R}$ Lipschitz-neprekidna funkcija. Tada prema (4.1) za sve $x, y \in [a, b]$ vrijedi

$$-L|x - y| \leq f(x) - f(y) \leq L|x - y|,$$

iz čega slijedi

$$f(x) \geq f(y) - L|x - y|, \quad \forall x, y \in [a, b], \quad (4.2)$$

(vidi Sliku 4.6). Iz toga zaključujemo da funkciju f u bilo kojoj točki $y \in [a, b]$ možemo odozdo ograničiti funkcijom $x \mapsto f(y) - L|x - y|$.



Slika 4.6: Lipschitz-donja ograda funkcije

Izaberimo proizvoljni $u_0 \in [a, b]$ i definirajmo donju ogradu funkcije f kao Lipschitz-neprekidnu po dijelovima linearnu funkciju $K: [a, b] \rightarrow \mathbb{R}$ (vidi Sliku 4.6):

$$K(u; u_0) = f(u_0) - L|u - u_0|, \quad u \in [a, b]. \quad (4.3)$$

Za slomljeni pravac K lako se vidi da vrijedi:

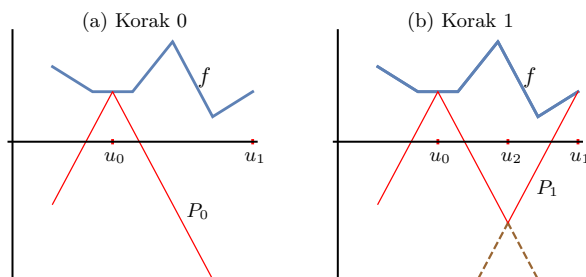
$$K(u_0; u_0) = f(u_0), \quad (4.4)$$

$$K(u; u_0) \leq f(u), \quad \forall u \in [a, b]. \quad (4.5)$$

Svojstvo (4.5) slijedi iz

$$\begin{aligned} K(u; u_0) - f(u) &= f(u_0) - f(u) - L|u - u_0| \leq |f(u_0) - f(u)| - L|u - u_0| \\ &\leq L|u - u_0| - L|u - u_0| = 0. \end{aligned}$$

Uz pretpostavku da postoji $u^* \in [a, b]$ na kojemu se postiže globalni minimum funkcije $f: [a, b] \rightarrow \mathbb{R}$, tj. postoji $u^* \in \operatorname{argmin}_{u \in [a, b]} f(u)$, Pijavskij je konstruirao sljedeći Algoritam slomljenih pravaca.



Slika 4.7: Algoritam Pijavskog

Najprije, uz poznavanje u_0 i $K(u; u_0)$ odredimo $u_1 \in \operatorname{argmin}_{u \in [a, b]} K(u; u_0)$ i stavimo $P_0(u) := K(u; u_0)$ (vidi Sliku 4.9a). Očigledno će biti $u_1 = a$ ili $u_1 = b$.

U prvom koraku definiramo

$$\begin{aligned} K(u; u_1) &:= f(u_1) - L|u - u_1|, \\ P_1(u) &:= \max_{i=0,1} K(u; u_i) = \max\{K(u; u_1), P_0(u)\}, \end{aligned}$$

i odredimo $u_2 \in \operatorname{argmin}_{u \in [a, b]} P_1(u)$ (vidi Sliku 4.9b).

U sljedećem (drugom) koraku definiramo

$$\begin{aligned} K(u; u_2) &:= f(u_2) - L|u - u_2|, \\ P_2(u) &:= \max_{i=0,1,2} K(u; u_i) = \max\{K(u; u_2), P_1(u)\}, \end{aligned}$$

i odredimo $u_3 \in \operatorname{argmin}_{u \in [a, b]} P_2(u)$;

Općenito, u k -tom koraku, poznavanjem čvorova u_0, u_1, \dots, u_k i funkcija $K(u; u_0), K(u; u_1), \dots, K(u; u_{k-1})$ definiramo

$$K(u; u_k) := f(u_k) - L|u - u_k|, \quad (4.6)$$

$$P_k(u) = \max_{i=0, \dots, k} K(u; u_i) = \max \{K(u; u_k), P_{k-1}(u)\} \quad (4.7)$$

i odredimo $u_{k+1} \in \operatorname{argmin}_{u \in [a, b]} P_n(k)$. Ako se $\min_{u \in [a, b]} P_n(k)$ postiže u više točaka, onda za u_{k+1} biramo bilo koju od njih. Pseudokod Algoritma 10 naveden je niže, a animaciju algoritma i odgovarajući modul možemo vidjeti u t.6.4.1, str.138).

Algoritam 10 (Algoritam slomljenih pravaca)

Input: $f: [a, b] \rightarrow \mathbb{R}$, $L > 0$

Input: $u_0 \in [a, b]$; $n \geq 1$; $k = 0$;

{Učitati funkciju i dozvoljeni broj iteracija;}

1: Definirati $K(u; u_0) := f(u_0) - L|u - u_0|$ i $P_0(u) := K(u; u_0)$;

2: Odrediti $u_1 \in \operatorname{argmin}_{u \in [a, b]} P_0(u) \subseteq \{a, b\}$;

3: **while** $k = k + 1$; $k < n$, **do**

4: Definirati $K(u; u_k) := f(u_k) - L|u - u_k|$;

5: $P_k(u) := \max_{i=0, 1, \dots, k} K(u; u_i) = \max \{K(u; u_k), P_{k-1}(u)\}$;

6: Odrediti $u_{k+1} \in \operatorname{argmin}_{u \in [a, b]} P_k(u)$;

7: **end while**

Output: $\{u_{k+1}, f(u_{k+1})\}$

Sljedeća lema pokazuje da Algoritam 10 za funkciju $f \in \operatorname{Lip}_L[a, b]$ daje rastući niz donjih ograda (P_k) , $P_k \in \operatorname{Lip}_L[a, b]$, (po dijelovima linearne funkcije čiji dijelovi leže na pravcima s koeficijentima smjera L ili $(-L)$), koje se s funkcijom f podudaraju u točkama u_k .

Lema 4.1. *Neka je $f \in \operatorname{Lip}_L[a, b]$, (u_n) niz čvorova i (P_n) niz funkcija definiranih kao u Algoritmu 10. Tada vrijedi:*

(i) P_n je Lipschitz-neprekidna po dijelovima linearna funkcija, čiji dijelovi leže na pravcima s koeficijentima smjera L ili $(-L)$, tj. $P_n \in \operatorname{Lip}_L[a, b]$,

(ii) $P_n(u) \leq P_{n+1}(u)$, $\forall u \in [a, b]$ i $\forall n = 0, 1, \dots$ (*monotoni rast*),

(iii) $P_n(u) \leq f(u)$, $\forall u \in [a, b]$ i $\forall n = 0, 1, \dots$ (*ograničenost odozgo*),

(iv) $P_n(u_i) = f(u_i)$, $\forall i = 0, 1, \dots, n$ (*podudaranje u čvorovima*).

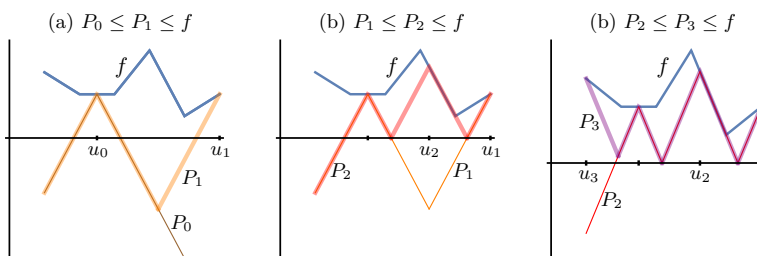
Dokaz. Tvrdnje (i) i (ii) su očigledne.

Tvrdnja (iii) dokazuje se induktivno. Najprije primijetimo da prema (4.5) vrijedi

$$P_0(u) = K(u; u_0) \leq f(u),$$

i pretpostavimo da je

$$P_{n-1}(u) \leq f(u).$$



Slika 4.8: Monotonost i ograničenost niza funkcija (P_n)

Kako je sukladno (4.5), $K(u; u_n) \leq f(u)$, vrijedi

$$P_n(u) = \max\{K(u; u_n), P_{n-1}(u)\} \leq \max\{f(u), f(u)\} = f(u),$$

tvrdnja (iii) vrijedi za svaki $n \in \mathbb{N}$.

U svrhu dokaza tvrdnje (iv) primijetimo najprije da vrijedi

$$f(u_i) = K(u_i; u_i), \quad \forall i \in \{0, 1, \dots, n\}. \quad (4.8)$$

Nadalje, na osnovi (4.7) vrijedi $[P_n(u) := \max_{i=0,1,\dots,n} K(u; u_i) \geq K(u; u_i)]$

$$K(u_i; u_i) \leq P_n(u_i), \quad \forall i \in \{0, 1, \dots, n\},$$

iz čega zaključujemo

$$f(u_i) = K(u_i; u_i) \leq P_n(u_i), \quad \forall i \in \{0, 1, \dots, n\}.$$

Zato korištenjem (iii) dobivamo

$$f(u_i) \leq P_n(u_i) \stackrel{(iii)}{\leq} f(u_i),$$

iz čega slijedi tražena tvrdnja (iv). □

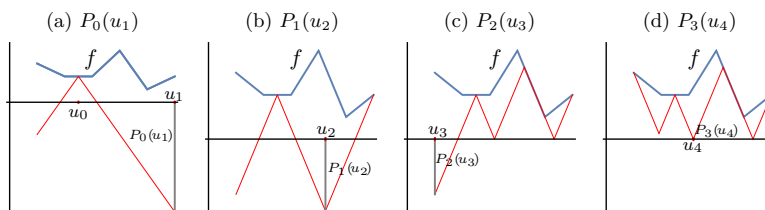
Sljedeći teorem pokazuje da Algoritam 10 pronalazi sve točke globalnog minimuma funkcije $f \in Lip_L[a, b]$.

Teorem 4.1. *Pretpostavimo:*

- (a) $f \in Lip_L[a, b]$;
- (b) Postoji $u^* \in \operatorname{argmin}_{u \in [a, b]} f(u)$;
- (c) Niz realnih brojeva (u_n) i niz funkcija (P_n) definiran je kao u Algoritmu 10.

Tada vrijedi:

- (i) $\lim_{n \rightarrow \infty} P_n(u_{n+1}) = \min_{u \in [a, b]} f(u) = f(u^*) =: f^*$;
- (ii) Ako je \hat{u} proizvoljno gomilište niza (u_n) , onda je $\hat{u} \in \operatorname{argmin}_{u \in [a, b]} f(u)$;
- (iii) Ako je $u^* = \operatorname{argmin}_{u \in [a, b]} f(u)$ (jednočlan), onda cijeli niz (u_n) konvergira prema u^* .



Slika 4.9: Monotonost i ograničenost niza $(P_n(u_{n+1}))$

Dokaz. Dokažimo najprije da je niz $(P_n(u_{n+1}))$ konvergentan. On je monotonu rastući jer je

$$\begin{aligned}
 P_{n-1}(u_n) &= \min_{u \in [a, b]} P_{n-1}(u) && \text{[po definiciji]} \\
 &\leq P_{n-1}(u_{n+1}) && \text{[po definiciji]} \\
 &\leq P_n(u_{n+1}). && \text{[prema Lemi 4.1(ii)]}
 \end{aligned}$$

Niz $(P_n(u_{n+1}))$ ograničen je odozgo jer vrijedi

$$\begin{aligned} P_n(u_{n+1}) &= \min_{u \in [a,b]} P_n(u) && \text{(po definiciji)} \\ &\leq \min_{u \in [a,b]} f(u) = f(u^*) = f^*. && \text{(prema Lemi 4.1(iii))} \end{aligned}$$

Dakle, postoji $\lim_{n \rightarrow \infty} P_n(u_{n+1}) =: P^*$, a prema Lemi 4.1(iii) i zbog neprekidnosti funkcija P_n i f iz prethodne nejednakosti slijedi

$$P^* \leq f^*. \quad (4.9)$$

Pokažimo da je $P^* = f^*$. U tu svrhu najprije pokažimo da vrijedi nejednakost

$$0 \leq f(u_i) - P_n(u_{n+1}) \leq L|u_i - u_{n+1}|, \quad \forall n \in \mathbb{N} \text{ i } \forall i = 0, \dots, n. \quad (4.10)$$

Naime, kako je $P_n(u_{n+1}) = \min_{u \in [a,b]} P_n(u) \leq P_n(u_i)$, za proizvoljni $n \in \mathbb{N}$ i proizvoljni $(0 \leq i \leq n)$ vrijedi

$$P_n(u_{n+1}) \leq P_n(u_i) = f(u_i) \quad i = 0, 1, \dots, n \quad [\text{Lema 4.1(iv)}],$$

iz čega slijedi

$$\begin{aligned} 0 &\leq f(u_i) - P_n(u_{n+1}) \\ &= P_n(u_i) - P_n(u_{n+1}) && [\text{prema Lemi 4.1(iv)}] \\ &\leq L|u_i - u_{n+1}| && [\text{prema Lemi 4.1(i)}] \end{aligned}$$

Kako je niz $(u_n) \subset [a, b]$ ograničen, postoji barem jedno gomilište \hat{u} i podniz (u_{n_k}) koji konvergira prema tom gomilištu.

Primijetite da zbog $n_{k-1} < n_k$ vrijedi $n_{k-1} \leq n_k - 1$. Zato iz nejednakosti (4.10) za $n := n_k - 1$ i $i := n_{k-1}$ dobivamo

$$0 \leq f(u_{n_{k-1}}) - P_{n_k-1}(u_{n_k}) \leq L|u_{n_{k-1}} - u_{n_k}|,$$

odakle zbog neprekidnosti funkcije f za $k \rightarrow \infty$ dobivamo

$$0 \leq f(\hat{u}) - \lim_{k \rightarrow \infty} P_{n_k-1}(u_{n_k}) \leq 0 \quad \implies \quad f(\hat{u}) = P^*.$$

Kako je $f^* \leq f(\hat{u})$, iz prethodne jednakosti i nejednakosti (4.9) dobivamo

$$f^* \leq f(\hat{u}) = P^* \leq f^*,$$

tj. $P^* = \lim_{n \rightarrow \infty} P_n(u_{n+1}) = f^* = f(\hat{u})$. Time su tvrdnje (i) i (ii) dokazane.

Tvrdnja (iii) slijedi iz prve dvije. \square

Primjedba 4.3. U svakom koraku *Metode slomljenih pravaca* treba riješiti minimizacijski problem za po dijelovima linearnu funkciju P_n (korak 6. u Algoritmu 10). To je moguće vrlo efikasno napraviti pretraživanjem vrhova funkcije P_n . U tu svrhu Algoritam 10 treba implementirati tako da u svakoj iteraciji ažurira listu koordinata vrhova

$$V_n = \{(v_1, z_1), \dots, (v_m, z_m)\},$$

koji predstavljaju lokalne minimume funkcije P_n . Primjerice, za točku $u_0 \in (a, b)$ u početnoj iteraciji skup V_0 sastoji se od dva vrha: $(a, K(a; u_0))$ i $(b, K(b; u_0))$, gdje je $K(u; u_0)$ slomljeni pravac definiran u (4.3). Listu V_n treba organizirati tako da vrijedi $z_1 \geq \dots \geq z_m$, iz čega slijedi da je točka globalnog minimuma u_n funkcije P_n jednostavno v_m .

U sljedećoj iteraciji promatramo funkciju $P_{n+1}(u) = \max\{K(u; u_n), P_n(u)\}$ koja se od P_n razlikuje u dva nova vrha, osim u slučaju kada je $u_n = a$ ili $u_n = b$. Lista vrhova ažurira se tako da se iz nje izbaci posljednji vrh $(v_m, z_m) = (u_n, P_n(u_n))$ i dodaju dva nova vrha. Označimo nove vrhove kao lijevi (v_l, z_l) i desni (v_r, z_r) tako da je $v_l < u_n < v_r$. Uočimo da je točka (v_l, z_l) sjecište funkcije $K(u; u_n)$ sa susjednim lijevim slomljenim pravcem $K(u; u_l)$ koji je određen nekim čvorom koji ćemo označiti s u_l . Sada ćemo pokazati da je (v_l, z_l) moguće izračunati bez poznavanja vrijednosti u_l .

S obzirom da je u_l lijevo od u_n , točka (v_l, z_l) mora biti sjecište pravaca $u \mapsto f(u_n) + L(u - u_n)$ i $u \mapsto f(u_l) - L(u - u_l)$. Lako je izračunati da je tada

$$\begin{aligned} v_l &= \frac{1}{2}(u_l + u_n) + \frac{1}{2L}(f(u_l) - f(u_n)), \\ z_l &= K(v_l; u_n) = \frac{1}{2}(f(u_l) + f(u_n)) - \frac{L}{2}(u_n - u_l). \end{aligned} \tag{4.11}$$

Uočimo nadalje da za posljednji vrh u listi vrhova V_n vrijedi $(v_m, z_m) = (u_n, P_n(u_n)) = (u_n, K(u_n; u_l))$. Stoga (4.11) možemo zapisati kao

$$\begin{aligned} v_l &= u_n + \frac{1}{2L}(f(u_l) - L(u_n - u_l) - f(u_n)) \\ &= u_n + \frac{1}{2L}(K(u_n; u_l) - f(u_n)) \\ &= v_m + \frac{1}{2L}(z_m - f(v_m)), \end{aligned}$$

$$\begin{aligned}
z_l &= \frac{1}{2}(f(u_l) - L(u_n - u_l) + f(u_n)) \\
&= \frac{1}{2}(K(u_n; u_l) + f(u_n)) \\
&= \frac{1}{2}(z_m + f(v_m)).
\end{aligned}$$

Analogno se može pokazati da se desni vrh (v_r, z_r) može izračunati kao

$$\begin{aligned}
v_r &= v_m - \frac{1}{2L}(z_m - f(v_m)), \\
z_r &= \frac{1}{2}(z_m + f(v_m)).
\end{aligned}$$

Dakle, u svakoj iteraciji treba ažurirati listu vrhova izbacivanjem posljednjeg vrha (v_m, z_m) i dodavanjem dva nova vrha čije se koordinate jednostavno izračunavaju iz v_m, z_m i $f(v_m)$. Nakon toga dobivenu listu treba sortirati po drugoj komponenti što daje listu V_{n+1} na kojoj se cijeli postupak ponavlja u sljedećoj iteraciji. Ukoliko je u nekoj iteraciji $v_m = a$, tada se dodaje samo desni vrh (v_r, z_r) , a ako je $v_m = b$, onda se dodaje samo (v_l, z_l) .

Ovakva implementacija Algoritma 10 napravljena je u modulu `Pijavskij` (vidi t.6.4.1, str.138).

Primjedba 4.4. Najpoznatije varijante *Metode slomljenih pravaca* su: *Algoritam Pijavskog*, *Shubertov algoritam* i *Algoritam DIRECT* [26]. *Algoritam DIRECT* može se generalizirati za funkcije više varijabli [13, 14], dok za *Algoritam Pijavskog* i *Shubertov algoritam* to nije moguće. *Metoda slomljenih pravaca* ima niz prednosti:

1. Minimizirajuća funkcija može imati više stacionarnih točaka i lokalnih minimuma, a *Metoda slomljenih pravaca* sve će ih detektirati. Primjerice, funkcija $x \rightarrow |\sin 30x|$ na intervalu $[0, 1]$ postiže globalni minimum u 10 točaka.
2. Minimizirajuća funkcija ne mora biti derivabilna u svim točkama intervala $[a, b]$, već samo ispunjavati Lipschitzov uvjet (4.1).
3. Za proizvoljni $u_0 \in [a, b]$ metoda konvergira prema globalnom minimumu, a za njenu implementaciju potrebno je „samo” poznavanje Lipschitzove konstante L .
4. U svakom koraku metode treba riješiti minimizacijski problem za po dijelovima linearnu funkciju P_n , što se svodi na ispitivanje njenih poznatih vrhova. Pri tome se P_n od P_{n-1} može razlikovati u najviše dva nova vrha (vidi Primjedbu 4.3, str.80).

5. Može se pokazati da je *Metoda slomljenih pravaca* bliska optimalnoj strategiji traženja globalnog minimuma Lipschitz-neprekidne funkcije.
6. Metoda se može poopćiti za slučaj funkcija koje imaju konačno mnogo prekida prve vrste i koje su definirane na nepovezanom području [70].

Nedostaci *Metode slomljenih pravaca* su:

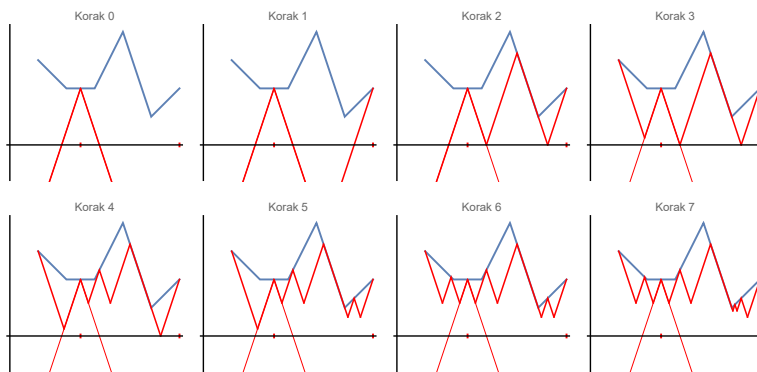
1. Određivanje Lipschitzove konstante L može biti složeno i numerički zahtjevno (vidi [58, 77]).
2. Općenito, metoda može imati malu brzinu konvergencije zbog:
 - Velike vrijednost Lipschitzove konstante $L > 0$. U graničnom slučaju kada $L \rightarrow \infty$, algoritam degenerira u izračunavanje vrijednosti funkcije na diskretnom skupu.
 - Svojstva algoritma da pronalazi sve točke globalnog minimuma, što može značajno opteretiti računski postupak.
3. Određivanje kriterija zaustavljanja nije jasno definirano u literaturi. Mogućnosti su sljedeće:
 - Unaprijed odrediti maksimalni broj iteracija;
 - Postupak završiti u k -tom koraku ako je $|u_{k+1} - u_k| < \epsilon$, gdje je $\epsilon > 0$ unaprijed zadan;
 - Postupak završiti u k -tom koraku ako je $|f(u_{k+1}) - f(u_k)| < \epsilon$, gdje je $\epsilon > 0$ unaprijed zadan.

Primjedba 4.5. Metodu možemo modificirati tako da na početku izaberemo $n + 1$ ekvidistantnih čvorova $a = u_0 < u_1 < \dots < u_n = b$ u kojima definiramo:

$$\begin{aligned}
 K(u; u_i) &:= f(u_i) - L|u - u_i|, \quad i = 0, 1, \dots, n \\
 P_n(u) &:= \max_{i=0,1,\dots,n} K(u; u_i) \\
 u_{n+1} &= \operatorname{argmin}_{u \in [a,b]} P_n(u) \quad [\text{pretraživanje } n + 1 \text{ čvorova}]
 \end{aligned}$$

Dalje postupak može teći kao u Algoritmu 10.

Primjer 4.6. *Metodu slomljenih pravaca ilustrirat ćemo na traženju globalnog minimuma funkcije iz Primjera 4.5, str.72.*



Slika 4.10: Metoda slomljenih pravaca

Iterativni proces može se pratiti u niže navedenoj tablici i na Slici 4.10 (pogledati također t.6.4.1, str.138). Može se vidjeti da je nakon 12 iteracija udaljenost susjednih čvorova, kao i apsolutna razlika funkcijskih vrijednosti pala na .05.

k	0	1	2	3	4	5	6	7	8	9	10	11	12
u_k	2.5	6	4.25	1	3.17	5.33	1.92	5.11	5.56	5.037	5.19	5.01	5.06
$ u_{k+1} - u_k $	-	3.5	1.75	3.25	2.17	2.17	3.42	3.19	0.44	0.52	0.15	0.17	0.05
$f(u_k)$	2	2	3.25	3	2.33	1.33	2.08	1.11	1.56	1.04	1.19	1.01	1.06
$ f(u_{k+1}) - f(u_k) $	-	0	1.25	0.25	0.67	1.	0.75	0.97	0.44	0.52	0.15	0.17	0.05

Zadatak 4.8. Za funkciju $f: [0, 9] \rightarrow \mathbb{R}$ zadanu s

$$f(x) = \begin{cases} 5 - x, & 0 \leq x \leq 3, \\ 4 - \frac{1}{2}(x - 5)^2, & 3 < x \leq 7, \\ \frac{1}{2}(x - 9)^2, & 7 < x \leq 9, \end{cases}$$

Metodom slomljenih pravaca odredite u_3 uz početnu točku $u_0 = 4$.

Zadatak 4.9. Za $u_0 = -6$ Metodom slomljenih pravaca odredite prve tri aproksimacije točke globalnog minimuma funkcije $f: [-6, 12] \rightarrow \mathbb{R}$ zadane s

$$f(x) = \min \{|x - 1| + 1, |x - 8| + 2\}.$$

4.3 Shubertova metoda

Pretpostavimo da je $f: [a, b] \rightarrow \mathbb{R}$ Lipschitz-neprekidna funkcija. Ako u nejednakost (4.1) umjesto y stavimo a , onda je $x - a \geq 0$ i vrijedi

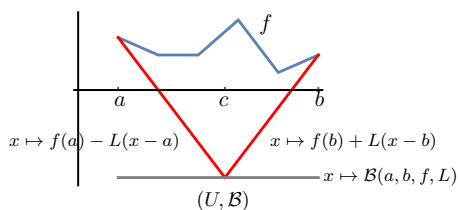
$$|f(x) - f(a)| \leq L(x - a), \quad \forall x \in [a, b],$$

odnosno

$$-L(x - a) \leq f(x) - f(a) \leq L(x - a), \quad \forall x \in [a, b],$$

odakle dobivamo jednu donju ogradu funkcije f na intervalu $[a, b]$

$$f(x) \geq f(a) - L(x - a). \quad (4.12)$$



Slika 4.11: Donja ograda funkcije f na intervalu $[a, b]$

Slično, ako za y stavimo b , onda je $x - b \leq 0$ i vrijedi

$$|f(x) - f(b)| \leq -L(x - b), \quad \forall x \in [a, b],$$

odnosno

$$L(x - b) \leq f(x) - f(b) \leq -L(x - b), \quad \forall x \in [a, b],$$

odakle dobivamo drugu donju ogradu funkcije f

$$f(x) \geq f(b) + L(x - b). \quad (4.13)$$

Funkcija

$$x \mapsto \max\{f(a) - L(x - a), f(b) + L(x - b)\}, \quad (4.14)$$

također je donja ograda funkcije f (vidi Sliku 4.11), koja postiže svoj globalni minimum $\mathcal{B}(a, b, f, L)$ u točki $U(a, b, f, L)$, gdje je

$$U(a, b, f, L) = \frac{1}{2}(a + b) + \frac{1}{2L}(f(a) - f(b)), \quad (4.15)$$

$$\mathcal{B}(a, b, f, L) = \frac{1}{2}(f(a) + f(b)) - \frac{L}{2}(b - a), \quad (4.16)$$

a njezin je graf slomljeni pravac.

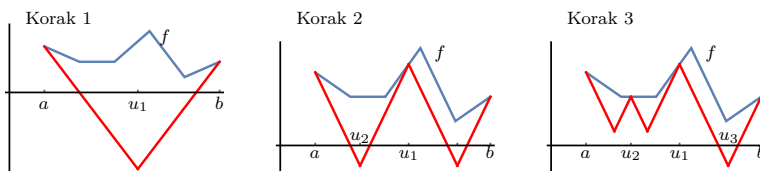
Primijetimo da je $x \mapsto \mathcal{B}(a, b, f, L)$ također donja ograda funkcije f na intervalu $[a, b]$. Naime, zbrajanjem (4.12) i (4.13) dobivamo

$$f(x) \geq \frac{1}{2}(f(a) + f(b)) - \frac{L}{2}(b - a) = \mathcal{B}(a, b, f, L).$$

Algoritam 11 (Shubertov algoritam)

Input: $f, L > 0$

- 1: Odrediti pravce $x \mapsto f(a) - L(x - a)$ i $x \mapsto f(b) + L(x - b)$ i točku njihovog sjecišta $(U(a, b, f, L), \mathcal{B}(a, b, f, L))$;
 - 2: Staviti $u_1 = U(a, b, f, L)$. Izabrati jedan od dva intervala $[a, u_1]$ i $[u_1, b]$.
 - 3: Ako je $[a, u_1]$ izabrani interval, staviti $u_2 = U(a, u_1, f, L)$.
Od tri intervala: $[a, u_2]$, $[u_2, u_1]$, $[u_1, b]$ izabrati onaj s najmanjom \mathcal{B} -vrijednosti i na njemu ponoviti postupak.
-



Slika 4.12: Shubertov algoritam

Animacija prvih koraka algoritma prikazana je na Slici 4.12. Dodatne primjere i ilustracije možemo dobiti korištenjem *Mathematica*-modula opisanog u t.6.4.2, str.139.

Primjedba 4.6. Primijetite da je *Shubertova metoda* zapravo specijalni slučaj *Algoritma Pijavskog* ako za početnu aproksimaciju izaberemo jedan od rubova intervala $[a, b]$. Neka je primjerice, $u_0 = a$. Tada prema algoritmu Pijavskog imamo (vidi Sliku 4.13)

$$u_1 = \operatorname{argmin}\{f(a) - L|u - a|\} = b \quad [\text{Korak 1}]$$

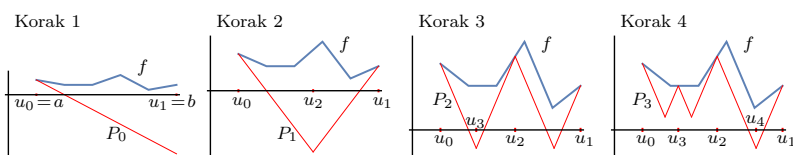
$$P_1(u) = \max\{f(a) - L|u - a|, f(b) - L|u - b|\},$$

$$u_2 = \operatorname{argmin} P_1(u) \quad [\text{Korak 2}].$$

Nakon toga definiramo

$$P_2(u) = \max\{P_1(u), f(u_2) - L|u - u_2|\},$$

$$u_3 = \operatorname{argmin} P_2(u) \quad [\text{Korak 3}]$$



Slika 4.13: Shubertov algoritam kao specijalan slučaj Algoritma Pijavskog

Očigledno se odluka može donijeti poznavanjem \mathcal{B} -vrijednosti funkcije f na intervalima $[a, u_2]$ i $[u_2, b]$. Ovaj kriterij podudara se s kriterijem navedenim u Koraku 2 Shubertovog algoritma.

Primjedba 4.7. Nažalost, ni Metoda slomljenih pravaca, a onda ni Shubertova metoda, ne može se generalizirati za slučaj više dimenzija.

Zadatak 4.10. Za funkciju iz Zadatka 4.8 Shubertovom metodom odredite prve četiri aproksimacije točke globalnog minimuma.

4.4 Optimizacijski algoritam DIRECT

Pretpostavimo da je $f: [a, b] \rightarrow \mathbb{R}$ Lipschitz-neprekidna funkcija. Prema [26], Metodu Pijavskog, odnosno Shubertovu metodu, modificirat ćemo tako da je generalizacija na višedimenzionalni slučaj moguća. Naziv te metode je DIRECT, što dolazi od DI(viding)RECT(angles), a aludira na odgovarajući algoritam u više dimenzija (vidi t.5.2, str.104).

Ako u (4.1) za y stavimo $c = \frac{a+b}{2}$, onda za $x \leq c$, odnosno $x - c \leq 0$, dobivamo

$$L(x - c) \leq f(x) - f(c) \leq -L(x - c),$$

odnosno

$$f(x) \geq f(c) + L(x - c), \quad x \leq c, \quad (4.17)$$

iz čega zaključujemo da je funkcija $x \mapsto f(c) + L(x - c)$ jedna donja ograda funkcije f na intervalu $[a, c]$ (vidi Sliku 4.14).

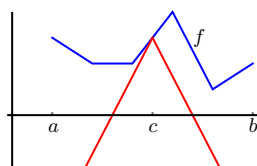
Slično, za $x \geq c$, odnosno $x - c \geq 0$, dobivamo

$$-L(x - c) \leq f(x) - f(c) \leq L(x - c),$$

odnosno

$$f(x) \geq f(c) - L(x - c), \quad x \geq c, \quad (4.18)$$

iz čega zaključujemo da je i funkcija $x \mapsto f(c) - L(x - c)$ jedna donja ograda funkcije f na intervalu $[c, b]$ (vidi Sliku 4.14).

Slika 4.14: Donja ograda funkcije f na intervalu $[a, b]$

Prema tome, i funkcija

$$x \mapsto \begin{cases} f(c) + L(x - c), & x \leq c \\ f(c) - L(x - c), & x \geq c \end{cases} \quad (4.19)$$

također je donja ograda funkcije f na intervalu $[a, b]$ (vidi Sliku 4.14), koja postiže svoj globalni minimum $f(c) - L\frac{b-a}{2}$ u lijevom i desnom rubu intervala $[a, b]$. Kod izbora donje ograde funkcije f na intervalu $[a, b]$ uvijek nastojimo da ta donja ograda bude što bliža funkciji f . U ovom slučaju to znači da bi konstanta $L > 0$ trebala biti što manja. U konkretnim praktičnim problemima obično ne poznajemo tu najmanju moguću vrijednost Lipschitzove konstante $L > 0$, pa se moramo zadovoljiti nekom aproksimacijom. Efikasnost algoritma kojeg ćemo konstruirati ovisit će upravo o tome.

Primijetite da uz poznavanje Lipschitzove konstante L donja ograda funkcije f ovisi samo o vrijednosti funkcije f u centru intervala $[a, b]$ i širini intervala. Ovo svojstvo imat će ključnu ulogu prilikom konstrukcije optimizacijskog algoritma, ali posebno i kod generalizacije na višedimenzionalni slučaj. Minimum donje ograde funkcije $f: [a, b] \rightarrow \mathbb{R}$ definirane s (4.19) zvat ćemo \mathcal{B} -vrijednost i označavati s

$$\mathcal{B}(c, d) = f(c) - Ld, \quad d = \frac{b-a}{2}. \quad (4.20)$$

Primjedba 4.8. Primijetite da se \mathcal{B} -vrijednost podudara s minimumom funkcije $u \mapsto K(u; c)$ definirane s (4.3) kod *Metode slomljenih pravaca* (vidi t.4.2, str.74).

Modul koji implementira *Algoritam DIRECT* uz poznatu vrijednost Lipschitzove konstante $L > 0$ opisan je u t.6.4, str.138.

Primjedba 4.9. Primijetite da vrijedi:

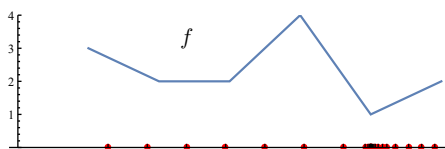
- Prema Teoremu 4.1, proizvoljno gomilište niza (c_n) točka je u kojoj funkcija f postiže svoj globalni minimum;

Algoritam 12 (Algoritam DIRECT, [26])**Input:** $f \in Lip_L[a, b]$, $L > 0$

- 1: Interval $[a, b]$ poluširine $d = \frac{b-a}{2}$ s centrom $c = \frac{a+b}{2}$ podijeliti na 3 podintervala jednake poluširine $\frac{d}{3}$ s centrima $c_1 = c - 2\frac{d}{3}$, $c_2 = c$, $c_3 = c + 2\frac{d}{3}$;
- 2: Staviti $n = 3$;
- 3: Za niz podintervala $[a_1, b_1], \dots, [a_n, b_n]$ s centrima $c_i = \frac{a_i+b_i}{2}$ i poluširinama $d_i = \frac{b_i-a_i}{2}$ izračunati \mathcal{B} -vrijednosti $\mathcal{B}(c_i, d_i) = f(c_i) - Ld_i$, $i = 1, \dots, n$;
- 4: Odrediti $i_0 = \operatorname{argmin}_{i=1, \dots, n} f(c_i)$ i staviti $\min = \{c_{i_0}, f(c_{i_0})\}$;
- 5: Odrediti $j = \operatorname{argmin}_{i=1, \dots, n} \mathcal{B}(c_i, d_i)$;
- 6: Interval s centrom u c_j podijeliti na 3 podintervala poluširine $\frac{d_j}{3}$;
- 7: Korigirati \min s vrijednosti funkcije u dva nova centra: $c_j - 2\frac{d_j}{3}$, $c_j + 2\frac{d_j}{3}$;
- 8: Izračunati \mathcal{B} -vrijednosti za tri nova podintervala;
- 9: Staviti $n := n + 2$ i prijeći na *Korak 5*;

- \mathcal{B} -vrijednost izabranog intervala nije veća od vrijednosti funkcije f u centru tog intervala $[\mathcal{B}(c_j, d_j) = f(c_j) - Ld_j \leq f(c_j)]$;
- Iterativni postupak može se ubrzati korištenjem činjenicu da se u svakom koraku od postojećeg niza podintervala ispušta jedan i dodaju tri nova podintervala.

Primjer 4.7. Primjenom Algoritma DIRECT odredit ćemo globalni minimum funkcije $f: [1, 6] \rightarrow \mathbb{R}$ zadane u Primjeru 4.5, str.72. Na Slici 4.15 prikazan je niz aproksimacija (c_n) globalnog minimuma koji se postiže u točki $x^* = 5$.



Slika 4.15: Niz aproksimacija (c_n) globalnog minimuma funkcije iz Primjera 4.5, str.72

Zadatak 4.11. Za funkciju iz Zadatka 4.8 provedite prve dvije iteracije Algoritma DIRECT.

Zadatak 4.12. Za funkciju $f: [0, 18] \rightarrow \mathbb{R}$, $f(x) = \min\{|x-5|+1, |x-10|+2\}$ provedite prve tri iteracije Algoritma DIRECT i u svakom koraku zapišite podintervale s kojima se nastavlja algoritam.

4.4.1 Optimizacija algoritma DIRECT

U *Algoritmu* DIRECT najvažniji korak je Korak 5 u kojemu tražimo interval s najmanjom \mathcal{B} -vrijednosti. Pokazat ćemo kako se to može provesti, a da se pri tome ne koristi informacija o Lipschitzovoj konstanti L . To je izuzetno važna činjenica jer smo već ranije spomenuli da je traženje povoljne (što manje) vrijednosti Lipschitzove konstante L općenito vrlo zahtjevan proces [77], a korištenje grube aproksimacije (velika vrijednost za L) jako usporava algoritam.

Sukladno ideji iznesenoj u radu [26], naglasimo još jednom važnost činjenice da smo \mathcal{B} -vrijednosti (4.20) intervala $[a_1, b_1], \dots, [a_n, b_n]$ u *Algoritmu* DIRECT predstavili samo njihovim centrima c_i , $i = 1, \dots, n$ i njihovim poluširinama $d_i = \frac{b_i - a_i}{2}$. Zato ćemo svakom intervalu $[a_i, b_i]$ s centrom c_i pridružiti točku

$$T_i = (d_i, f(c_i)),$$

a \mathcal{B} -vrijednost intervala $[a_i, b_i]$

$$\mathcal{B}_i := \mathcal{B}(c_i, d_i) = f(c_i) - Ld_i, \quad \text{gdje je} \quad d_i = \frac{b_i - a_i}{2},$$

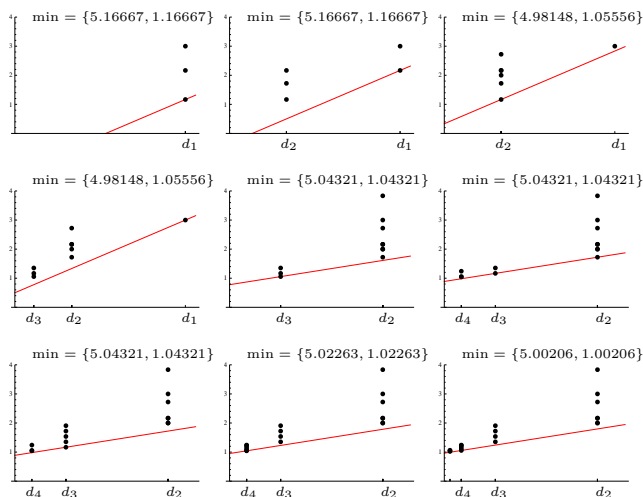
može se interpretirati kao odsječak na ordinati linearne funkcije l_i čiji graf prolazi točkom T_i i ima koeficijent smjera L :

$$l_i(x) = Lx + \mathcal{B}_i. \quad (4.21)$$

Razmotrimo nekoliko prvih iteracija *Algoritma* DIRECT kod traženja globalnog minimuma funkcije f iz Primjera 4.5, str.72. Svaki grafikon na Slici 4.16 prikazuje točke T_i , $i = 1, \dots, n$ dobivene u nekoj iteraciji *Algoritma* DIRECT i pravac l_j kroz točku T_j s koeficijentom smjera L , pri čemu je j izabran tako da bude $\mathcal{B}_j = \min_{i=1, \dots, n} \mathcal{B}_i$. Dakle, j je redni broj intervala s najmanjom \mathcal{B} -vrijednosti, odnosno najmanjim odsječkom na ordinati odgovarajućeg pravca l_j . U prvoj iteraciji pojavljuje se $n = 3$ točke, u drugoj $n = 5$ točaka, u trećoj $n = 7$ točaka, itd. Pritom se neke točke podudaraju.

Ako se na intervalu $[a_j, b_j]$, $j \in I$ postigne najmanja \mathcal{B} -vrijednost, onda je odsječak na ordinati pravca l_j s koeficijentom smjera L a koji prolazi točkom T_j najmanji u odnosu na sve druge takve pravce koji prolaze točkama T_1, \dots, T_n , a sve točke T_i leže iznad ili na pravcu l_j .

Postavlja se pitanje: „Postoji li možda neki pravac iznad kojeg leže sve točke T_i s manjim koeficijentom smjera $0 < \hat{L} < L$?” Već na drugom grafikonu Slike 4.16 vidi se da je bilo moguće izabrati „bolji pravac” – donju ogradu točaka T_i , $i = 1, \dots, n$ izborom nove točke T_j i manjeg koeficijenta



Slika 4.16: Niz aproksimacija (c_n) globalnog minimuma funkcije iz Primjera 4.5, str.72

smjera $0 < \hat{L} \leq L$. Problem bismo mogli proširiti na sljedeći način: „Trebamo pronaći sve točke T_j , $j \in I$, za koje je moguće pronaći neku konstantu $\hat{L} > 0$, tako da nijedna od točaka T_i , $i \in I$ ne bude ispod pravca $x \mapsto \hat{L}x + (f(c_j) - \hat{L}d_j)$, odnosno tako da odgovarajuća \mathcal{B} -vrijednost koja pripada točki T_j bude najmanja: $f(c_j) - \hat{L}d_j \leq f(c_i) - \hat{L}d_i$ za svaki $i \in I$.” S tom motivacijom uvodimo sljedeću definiciju.

Definicija 4.2. Interval $[a_j, b_j]$ s centrom c_j i poluširinom $d_j = \frac{b_j - a_j}{2}$ je potencijalno optimalan ako postoji $\hat{L} > 0$, tako da bude

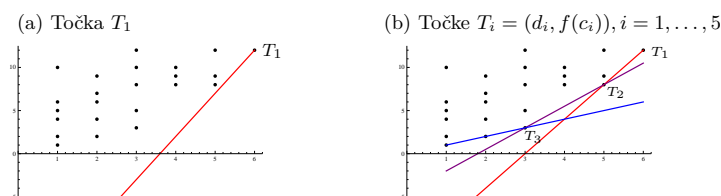
$$f(c_j) - \hat{L}d_j \leq f(c_i) - \hat{L}d_i, \quad \forall i \in I = \{1, \dots, n\}. \quad (4.22)$$

Formula (4.22) znači da je za dani $\hat{L} > 0$ odsječak na ordinati $\mathcal{B}_j = f(c_j) - \hat{L}d_j$ pravca (4.21), koji prolazi točkom T_j manji od ili jednak odsječcima na ordinati $\mathcal{B}_i = f(c_i) - \hat{L}d_i$ pravaca koji prolaze drugim točkama T_i , $i \in I$. Izborom različitih vrijednosti konstante $\hat{L} > 0$ u svakoj iteraciji može postojati više potencijalno optimalnih intervala.

Geometrijski gledano, interval $[a_j, b_j]$ bit će potencijalno optimalan ako birajući $\hat{L} > 0$ možemo pronaći pravac kroz točku T_j koji je ujedno donja ograda za sve druge točke T_i , $i = 1, \dots, n$. Ako pri tome sve točke jednake apscise (jednake poluširine intervala) uredimo prema veličini ordinate, onda

potencijalno optimalne intervale treba tražiti između točaka najmanje ordinate (vidi Sliku 4.16). Od svih tih „donjih” točaka treba izabrati one kroz koje prolazi pravac iznad kojega se nalaze sve ostale točke. Primijetite da su to točke koje leže na konveksnoj ljusci skupa točaka $T_i, i \in I$.

Primjer 4.8. Na Slici 4.17 ilustriran je postupak traženja potencijalno optimalnih podintervala za problem iz Primjera 4.7. Za skup točaka prikazan na Slici 4.17a i za $\hat{L} = 5$ sve točke leže iznad pravca $l_1(x) = \hat{L}x + (f(c_1) - \hat{L}d_1) = 5x + (12 - 5 \cdot 6)$. Kao što se može vidjeti na Slici 4.17, točka $T_1 = (d_1, f(c_1))$ predstavlja potencijalno optimalan interval prema Definiciji 4.2.



Slika 4.17: Traženje potencijalno optimalnih intervala

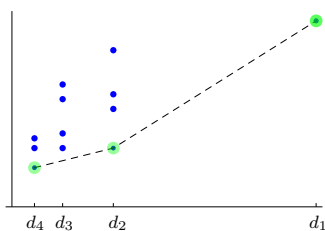
Međutim, kao što se može vidjeti, ima i boljih pravaca s manjim koeficijentom smjera, a koji imaju spomenuto svojstvo (Slika 4.17b):

$$\begin{aligned} f(c_1) - L_1 d_1 &\leq f(c_i) - L_1 d_i && \text{za sve } i, \\ f(c_2) - L_2 d_2 &\leq f(c_i) - L_2 d_i && \text{za sve } i, \\ f(c_3) - L_2 d_3 &\leq f(c_i) - L_2 d_i && \text{za sve } i. \end{aligned}$$

Svaki od ovih pravaca dobar je izbor: određuje barem jedan potencijalno optimalan interval (točku)! Primijetite da sve dobre točke leže na konveksnoj ljusci. Primijetite također da Lipschitzova konstanta na užem podintervalu u pravilu postaje manja, a time i algoritam efikasniji.

4.4.2 Traženje potencijalno optimalnih intervala

Geometrijski gledano, točka T_i na Slici 4.18 predstavljat će potencijalno optimalan interval u smislu Definicije 4.2 ako se kroz nju može provući pravac s nekim koeficijentom smjera $\hat{L} > 0$ iznad kojega (ili na kojemu) se nalaze sve druge točke.



Slika 4.18: Zelene točke odgovaraju potencijalno optimalnim intervalima

U nastavku ćemo operacionalizirati opisano načelo i sukladno Definiciji 4.2 identificirati potencijalno optimalne intervale bez korištenja konstante $\hat{L} > 0$.

Označimo $I := \{1, \dots, n\}$, $d_i := \frac{b_i - a_i}{2}$ i primijetimo najprije da se za izabrani $j \in I$ (a time i izabrani d_j) na Slici 4.18 može pojaviti jedna ili više točaka. Sve točke s apscisama d_j predstavljaju intervale poluširine d_j . Vrijedi sljedeća lema.

Lema 4.2. [14] *Neka je $f \in \text{Lip}_L[a, b]$ Lipschitz neprekidna funkcija s konstantom $L > 0$, $d_i, c_i, i \in I$ poluširine i centri intervala nastali dijeljenjem intervala $[a, b]$ u Algoritmu *DIRECT*.*

Interval $[a_j, b_j]$, $j \in I$ je potencijalno optimalan u smislu Definicije 4.2 onda i samo onda ako postoji $\hat{L} > 0$ takav da vrijedi

$$f(c_j) \leq f(c_i), \quad \forall i \in I_0, \quad (4.23)$$

$$\max_{k \in I_L} \frac{f(c_j) - f(c_k)}{d_j - d_k} \leq \hat{L} \leq \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}, \quad (4.24)$$

gdje je

- $I_0 = \{i \in I: d_i = d_j\}$ – indeksi točaka s apscisom d_j ;
- $I_L = \{i \in I: d_i < d_j\}$ – indeksi točaka koje se na grafikonu (4.18) nalaze lijevo od točaka s apscisom d_j ;
- $I_R = \{i \in I: d_i > d_j\}$ – indeksi točaka koje se na grafikonu (4.18) nalaze desno od točaka s apscisom d_j .

Primjedba 4.10. Za praktične primjene neće biti potrebno pronalaženje konstante $\hat{L} > 0$ iz (4.24).

Primjedba 4.11. Primjerice, ako je j indeks točke T_j sa Slike 4.18 koja ima apscisu $d_1 = \frac{b-a}{2}$, imamo: $|I_0| = 1$, $|I_L| = 11$, $|I_R| = 0$. Ako je j indeks točke T_j koja ima apscisu $d_2 = \frac{d_1}{3} = \frac{b-a}{6}$, imamo: $|I_0| = 4$, $|I_L| = 7$, $|I_R| = 1$. Ako je j indeks točke T_j koja ima apscisu $d_4 = \frac{d_3}{3} = \frac{b-a}{27}$, imamo: $|I_0| = 3$, $|I_L| = 0$, $|I_R| = 9$.

Važno je primijetiti da je skup I_L za točke iz prvog stupca (najmanji d_j) prazan i da je skup I_R za točke iz posljednjeg stupca (najveći d_j) prazan.

Dokaz. Zapišimo (4.22) kao

$$f(c_j) - f(c_i) \leq \hat{L}(d_j - d_i), \quad \forall i \in I, \quad (4.25)$$

Nužnost.

Za sve $i \in I_0$, $d_i = d_j$, iz čega slijedi (4.23). Uvjeti (4.23) jednostavno znače da točka T_j mora biti najniža između svih točaka apscise d_j .

Za $i \in I_L$ ($d_j - d_i > 0$) uvjete (4.25) možemo zapisati kao

$$\frac{f(c_j) - f(c_i)}{d_j - d_i} \leq \max_{k \in I_L} \frac{f(c_j) - f(c_k)}{d_j - d_k} \leq \hat{L}, \quad \forall i \in I_L. \quad (4.26)$$

Za $i \in I_R$ ($d_j - d_i < 0$) uvjete (4.25) možemo zapisati kao

$$\frac{f(c_j) - f(c_i)}{d_j - d_i} \geq \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k} \geq \hat{L}, \quad \forall i \in I_R. \quad (4.27)$$

Nejednakosti (4.26) i (4.27) zajedno možemo zapisati kao

$$\max_{k \in I_L} \frac{f(c_j) - f(c_k)}{d_j - d_k} \leq \hat{L} \leq \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}, \quad \text{odnosno} \quad (4.28)$$

$$\max_{k \in I_L} \frac{f(c_j) - f(c_k)}{d_j - d_k} \leq \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k} \quad (4.29)$$

Primijetite da smo na taj način eliminirali konstantu \hat{L} .

Specijalno, ako je $I_L = \emptyset$, uvjet (4.26), a onda i uvjet (4.29), trivijalno su ispunjeni. Analogno, ako je $I_R = \emptyset$, uvjet (4.27), a onda i uvjet (4.29), trivijalno su ispunjeni. Stoga, u ovim slučajevima uvjet (4.24) interpretiramo korištenjem konvencije da je $\max \emptyset = -\infty$ i $\min \emptyset = \infty$. Dakle, za $I_L = \emptyset$ ili $I_R = \emptyset$ uvjet (4.29), odnosno (4.24), ne treba provjeravati.

Zaključno, ako je interval $j \in I$ potencijalno optimalan u smislu Definicije 4.2, onda su ispunjeni uvjeti (4.23)-(4.24).

Dovoljnost.

Pretpostavimo da su za interval $[a_j, b_j]$, $j \in I$ ispunjeni uvjeti (4.23)-(4.24). Pokažimo da tada postoji $\hat{L} > 0$ takav da vrijedi (4.22).

Za $i \in I_0$ vrijedi ($d_i = d_j$), pa za $\hat{L} > 0$ možemo uzeti bilo koji pozitivan broj jer iz $f(c_j) \leq f(c_i)$ slijedi (4.25), odnosno (4.22).

Za $i \in I_L \cup I_R$, iz (4.29) zaključujemo da postoji $\hat{L} > 0$, takav da vrijedi (4.28).

Ako je $i \in I_L$, vrijedi (4.26), odnosno (4.25), a ako je $i \in I_R$, vrijedi (4.27), iz čega ponovo slijedi (4.25). \square

Zadatak 4.13. Odredite potencijalno optimalne intervale za skup intervala dobiven u Zadatku 4.11 nakon dvije iteracije *Algoritma DIRECT*.

Zadatak 4.14. Za funkciju $f: [-6, 12] \rightarrow \mathbb{R}$, $f(x) = \min\{|x-1|+1, |x-8|+2\}$ odredite Lipschitzovu konstantu te koristeći taj rezultat provedite prve tri iteracije *Algoritma DIRECT*. Za dobivene intervale grafički prikazite točke $T_i = (d_i, f(c_i))$, gdje je d_i poluširina a c_i centar i -tog intervala. Odredite koji su intervale potencijalno optimalni te za njih odredite dozvoljeni raspon konstante \hat{L} u Definiciji 4.2.

Dodatno sužavanje skupa potencijalno optimalnih intervala

Skup potencijalno optimalnih intervala dobiven sukladno Definiciji 4.2 može se još više suziti tako da ne razmatramo intervale čije se \mathcal{B} -vrijednosti relativno malo razlikuju od trenutno minimalne vrijednosti funkcije f_{min} .

Na Slici 4.19 prikazan je potencijalno optimalan podinterval $[a_j, b_j]$ s centrom u točki c_j , čija se \mathcal{B} -vrijednost od trenutnog minimuma funkcije f_{min} razlikuje za manje od ϵ . Takav interval ne bismo trebali dalje dijeliti jer ne daje nade da će se na njemu postići globalni minimum.

Ako je $[a_j, b_j]$ potencijalno optimalan interval sukladno Definiciji 4.2, onda vrijedi

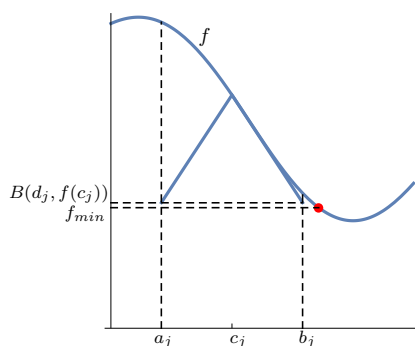
$$\mathcal{B}(d_j, f(c_j)) = f(c_j) - \hat{L}d_j \leq f_{min}.$$

Uz pretpostavku da je $f_{min} \neq 0$, to bi značilo da je relativno odstupanje aktualnog minimuma f_{min} od \mathcal{B} -vrijednosti j -tog podintervala nenegativno

$$\frac{f_{min} - (f(c_j) - \hat{L}d_j)}{|f_{min}|} \geq 0.$$

Ako je to odstupanje značajno (veće od nekog $\epsilon > 0$)

$$\frac{f_{min} - (f(c_j) - \hat{L}d_j)}{|f_{min}|} \geq \epsilon > 0, \quad (4.30)$$



Slika 4.19: Interval koji nije potencijalno optimalan

onda ćemo podinterval $[a_j, b_j]$ zadržati u skupu potencijalno optimalnih intervala. U protivnom nećemo ga smatrati potencijalno optimalnim. U literaturi [13, 14] preporuča se koristiti $\epsilon = 10^{-4}$.

Lema 4.3. [14] *Neka je $S \subset I$ skup indeksa potencijalno optimalnih intervala sukladno Definiciji 4.2, odnosno Lemi 4.2, i neka je $\epsilon > 0$. Relativno odstupanje aktualnog minimuma f_{min} od \mathcal{B} -vrijednosti j -tog podintervala $[a_j, b_j]$, $j \in S$ razlikuje se za više od ϵ onda i samo onda ako vrijedi*

$$\epsilon \leq \frac{f_{min} - f(c_j)}{|f_{min}|} + \frac{d_j}{|f_{min}|} \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}, \quad \text{ako je } f_{min} \neq 0, \quad (4.31)$$

odnosno

$$f(c_j) \leq d_j \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}, \quad \text{ako je } f_{min} = 0, \quad (4.32)$$

gdje je $I_R = \{i \in I : d_i > d_j\}$.

Dokaz. Nužnost.

Pretpostavimo da se potencijalno optimalni interval $[a_j, b_j]$, $j \in S$, razlikuje od aktualnog minimuma f_{min} za više od ϵ , tj. da za njega vrijedi (4.30).

Ako je $f_{min} \neq 0$ i $I_R \neq \emptyset$, iz (4.30) koristeći (4.27) dobivamo

$$\epsilon \leq \frac{f_{min} - f(c_j)}{|f_{min}|} + \hat{L} \frac{d_j}{|f_{min}|} \leq \frac{f_{min} - f(c_j)}{|f_{min}|} + \frac{d_j}{|f_{min}|} \min_{i \in I_R} \frac{f(c_j) - f(c_i)}{d_j - d_i}.$$

Nadalje (4.30) zapišimo kao

$$f(c_j) - \hat{L}d_j \leq f_{min} - \epsilon|f_{min}|. \quad (4.33)$$

Ako je $f_{min} = 0$ i $I_R \neq \emptyset$, iz (4.33) koristeći (4.27) dobivamo

$$f(c_j) \leq \hat{L}d_j \leq d_j \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}.$$

Zato uvjet (4.30) možemo zapisati pomoću (4.31)-(4.32). Dakle, na taj način i uvjete (4.30) zapisali smo bez korištenja konstante \hat{L} .

Dovoljnost.

Pretpostavimo da su za interval $[a_j, b_j]$, $j \in S$ ispunjeni uvjeti (4.31)-(4.32). Pokažimo da se tada \mathcal{B} -vrijednost potencijalno optimalnog intervala $[a_j, b_j]$, $j \in S$ razlikuje od aktualnog minimuma f_{min} za više od ϵ , odnosno da postoji $\hat{L} > 0$ takav da vrijedi (4.30). U tu svrhu dovoljno je uzeti $\hat{L} = \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}$. \square

Zadatak 4.15. Uočite zašto se u formuli (4.33) koristi apsolutna vrijednost trenutno minimalne vrijednosti funkcije f_{min} .

Primjedba 4.12. U programskom sustavu *Mathematica* ne treba posebno voditi računa o stupcima točaka za koje je $j = \operatorname{argmax}_{i \in I} d_i$ (najdesniji) i $j = \operatorname{argmin}_{i \in I} d_i$ (najlijeviji) jer u ovom programskom sustavu vrijedi:

$$\operatorname{Max}[\{\}] = -\infty \text{ i } \operatorname{Min}[\{\}] = \infty.$$

Sljedeći teorem objedinjuje Lemu 4.2 i Lemu 4.3 i daje karakterizaciju potencijalno optimalnih intervala čije se \mathcal{B} -vrijednosti značajnije razlikuju od aktualnog minimuma funkcije f . Takve intervale ima smisla dalje dijeliti u *Algoritmu DIRECT*.

Teorem 4.2. Neka je $f \in \operatorname{Lip}_L[a, b]$ Lipschitz neprekidna funkcija s konstantom $L > 0$, $\{d_i, c_i : i \in I\}$ poluširine i centri intervala nastali dijeljenjem intervala $[a, b]$ u *Algoritmu DIRECT*, $f_{min} = \min_{i \in I} f(c_i)$ aktualna vrijednost minimuma i $\epsilon > 0$.

Interval $[a_j, b_j]$, $j \in I$ je potencijalno optimalan u smislu Definicije 4.2, a relativno odstupanje aktualnog minimuma f_{min} od \mathcal{B} -vrijednosti j -tog podintervala $[a_j, b_j]$, $j \in S$ razlikuje se za više od ϵ onda i samo onda ako

vrijedi

$$f(c_j) \leq f(c_i), \quad \forall i \in I_0,$$

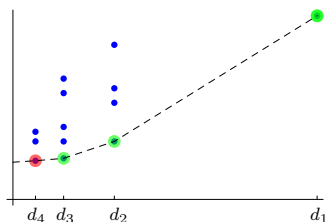
$$\max_{k \in I_L} \frac{f(c_j) - f(c_k)}{d_j - d_k} \leq \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k},$$

$$\begin{cases} \epsilon \leq \frac{f_{\min} - f(c_j)}{|f_{\min}|} + \frac{d_j}{|f_{\min}|} \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}, & \text{ako je } f_{\min} \neq 0, \\ f(c_j) \leq d_j \min_{k \in I_R} \frac{f(c_j) - f(c_k)}{d_j - d_k}, & \text{ako je } f_{\min} = 0, \end{cases}$$

gdje je

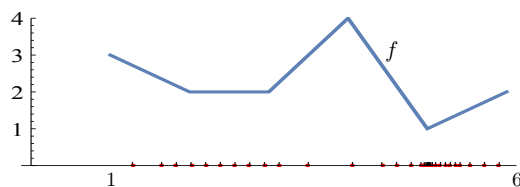
- $I_0 = \{i \in I: d_i = d_j\}$ – indeksi točaka s apscisom d_j ;
- $I_L = \{i \in I: d_i < d_j\}$ – indeksi točaka koje se na grafikonu (4.18) nalaze lijevo od točaka s apscisom d_j ;
- $I_R = \{i \in I: d_i > d_j\}$ – indeksi točaka koje se na grafikonu (4.18) nalaze desno od točaka s apscisom d_j .

Potencijalno optimalni intervali na Slici 4.20 prikazani su zelenim točkama. Nasuprot tome, interval koji je na Slici 4.20 predstavljen crvenom točkom zbog uvjeta (4.33) neće biti svrstan u potencijalno optimalne zato što se njegova \mathcal{B} -vrijednost gotovo ne razlikuje od trenutno minimalne vrijednosti funkcije f_{\min} . Geometrijsko objašnjenje uvjeta (4.33) može se također vidjeti na Slici 2.2 kod [13].



Slika 4.20: Zeleni točke odgovaraju potencijalno optimalnim intervalima, a crvena točka odgovara potencijalno optimalnom intervalu čija se \mathcal{B} -vrijednost ne razlikuje značajno od trenutnog minimuma f_{\min}

Primjer 4.9. Na Slici 4.21 prikazan je iterativni postupak Algoritma *DIRECT* s biranjem potencijalno optimalnih intervala u smislu Definicije 4.2 prilikom traženja globalnog minimuma funkcije iz Primjera 4.5, str. 72.

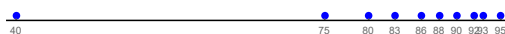


Slika 4.21: Primjena *Algoritma DIRECT* s izborom potencijalno optimalnih intervala za funkciju iz Primjera 4.5, str.72

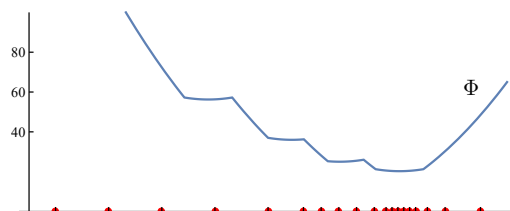
Primjer 4.10. *Kao reprezentant podataka $y = \{y_1, \dots, y_m\}$ možemo promatrati tzv. „Least Median of Squares” (LMS) (vidi [38, 54])*

$$LMS(y) = \operatorname{argmin}_{\lambda \in \mathbb{R}} \Phi(\lambda), \quad \Phi(\lambda) = \operatorname{med}_i (y_i - \lambda)^2. \quad (4.34)$$

Promatramo podatke koji predstavljaju 10 mjesečnih mjerenja dijastoličkog tlaka neke osobe: 90, 93, 86, 92, 95, 83, 75, 40, 88, 80 [54].



Graf funkcije Φ i prve 4 iteracije *Algoritma DIRECT* prikazane su na Slici 4.22. Pokušajte odrediti Lipschitzovu konstantu ove funkcije.



Slika 4.22: Primjena *Algoritma DIRECT* s izborom potencijalno optimalnih intervala na funkciju kojom se definira „Least Median of Squares”

Iz podataka možemo zaključiti da podatak 40 predstavlja outlier. Ako promotrimo aritmetičku sredinu uzorka, ona iznosi 82.2 i privučena je outlierom. S druge strane, medijan uzorka je 87, a LMS procjena iznosi 90.5. LMS procjena leži u području veće koncentracije podataka gdje većina točaka leži blizu jedna drugoj. Kad bismo outlier pomaknuli još dalje od ostatka podataka, ni medijan niti LMS ne bi promijenili svoju vrijednost. S druge strane, pomicanjem samo jednog podatka

možemo proizvoljno mijenjati vrijednost aritmetičke sredine. U tom smislu medijan i LMS predstavljaju primjere robusnih procjenitelja reprezentanata skupa podataka. Nedostatak LMS procjenitelja veća je varijabilnost od nekih drugih robusnih procjenitelja položaja. Za više detalja vidjeti [54].

Primjedba 4.13. Ako se u *Algoritmu* DIRECT primijeni paralelno programiranje, onda se može postaviti pitanje je li traženje podintervala s najmanjom \mathcal{B} -vrijednosti efikasnije (brže) izdvajanjem potencijalno optimalnih intervala sukladno Definiciji 4.2 ili je efikasnije (brže) jednostavno ispitivati sve „donje” točke u klasterima točaka jednake apscise?

Općenito nije! To pokazuju brojni primjeri iz primjena.

Poglavlje 5

Višedimenzionalna globalna optimizacija

Kao što smo spomenuli ranije u uvodu, problem globalne optimizacije pojavljuje se u različitim primjenama u gotovo svim područjima znanosti. Pri tome funkcija cilja može biti linearna, nelinearna, konveksna, nekonveksna, diferencijabilna, nediferencijabilna, može imati jednu ili više nezavisnih varijabli. Područje na kojemu je definirana funkcija može biti čitav skup \mathbb{R}^n (tada govorimo o problemu globalne optimizacije bez ograničenja), a može biti neki podskup u \mathbb{R}^n (tada govorimo o problemu globalne optimizacije s ograničenjima). Općenito, problem globalne optimizacije definirat ćemo na sljedeći način:

Treba odrediti točku $u^* \in \mathcal{D} \subseteq \mathbb{R}^n$ u kojoj funkcija $g: \mathcal{D} \rightarrow \mathbb{R}$ postiže globalni minimum $v^* = g(u^*) \in \mathbb{R}$, odnosno, treba odrediti točku $u^* \in \mathcal{D} \subseteq \mathbb{R}^n$ takvu da je

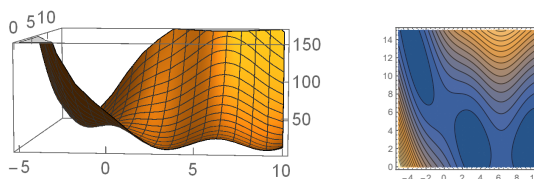
$$u^* \in \underset{u \in \mathcal{D}}{\operatorname{argmin}} g(u), \quad v^* = g(u^*). \quad (5.1)$$

Može se dogoditi da je skup $\underset{u \in \mathcal{D}}{\operatorname{argmin}} g(u)$ jednočlan (kao u Primjeru 1.2, str.7) ili višečlan (kao u Primjeru 5.1).

Primjer 5.1. Poznata Branin funkcija (koja se kao test primjer razmatra u brojnim radovima [13, 14, 26]) $g: [-5, 10] \times [0, 15] \rightarrow \mathbb{R}$,

$$g(x_1, x_2) = (x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6)^2 + 10(1 - \frac{1}{8\pi}) \cos x_1 + 10$$

postiže globalni minimum $g_{\min} = 0.398$ u tri različite točke. Jedna od njih je $u_1^* = (\pi, 2.275)$ (vidi Sliku 5.1).



Slika 5.1: Branin test funkcija [26]

U nastavku ćemo pretpostaviti da je domena funkcije g koju treba minimizirati hiperpravokutnik $\mathcal{D} = [a_1, b_1] \times \cdots \times [a_n, b_n]$ za neke $a_i < b_i$, $i = 1, \dots, n$. Prije pokretanja nekog optimizacijskog algoritma funkciju g najprije je korisno transformirati na jediničnu hiperkocku $[0, 1]^n$. U Primjeru 5.1 to znači da treba odrediti preslikavanje koje pravokutnik $[-5, 10] \times [0, 15]$ prevodi u kvadrat $[0, 1] \times [0, 1]$.

5.1 Transformacija problema na hiperkocku

U cilju transformacije optimizacijskog problema (5.1) na standardnu formu za funkciju $f: [0, 1]^n \rightarrow \mathbb{R}$

$$u^* \in \underset{u \in [0, 1]^n}{\operatorname{argmin}} f(u), \quad v^* = f(u^*), \quad (5.2)$$

potrebno je napraviti preslikavanje hiperpravokutnika $\mathcal{D} = \prod_{i=1}^n [a_i, b_i]$ na hiperkocku $[0, 1]^n$. Zbog jednostavnosti, u sljedećoj lemi pokazujemo kako se to može napraviti kada je $\mathcal{D} \subset \mathbb{R}^2$. Općeniti rezultat dobiva se analogno (vidi Zadatak 5.1).

Lema 5.1. *Neka su $a, b, c, d \in \mathbb{R}$ takvi da je $b > a$ i $d > c$. Jedinствena linearna bijekcija T koja pravokutnik $[a, b] \times [c, d]$ prevodi u jedinični kvadrat $[0, 1]^2$ i pri tome vrijedi $T(a, c) = (0, 0)$, $T(a, d) = (0, 1)$, $T(b, c) = (1, 0)$, $T(b, d) = (1, 1)$ zadana je s*

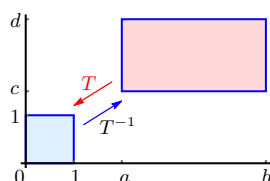
$$T(x) = A(x - u), \quad A = \begin{bmatrix} \frac{1}{b-a} & 0 \\ 0 & \frac{1}{d-c} \end{bmatrix}, \quad u = \begin{bmatrix} a \\ c \end{bmatrix}, \quad (5.3)$$

a inverzno preslikavanje $T^{-1}: [0, 1]^2 \rightarrow [a, b] \times [c, d]$ s

$$T^{-1}(x) = A^{-1}x + u. \quad (5.4)$$

Dokaz. Potražimo preslikavanje T u obliku $T(x) = Ax + b$, gdje je

$$A = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}, \quad b = \begin{bmatrix} \xi \\ \eta \end{bmatrix}.$$



Slika 5.2: Preslikavanje pravokutnika $[a, b] \times [c, d]$ na jedinični kvadrat $[0, 1]^2$

Po uvjetu treba biti

$$\begin{aligned} T(a, c)^T = (0, 0)^T &\Rightarrow \begin{cases} a\alpha + c\beta + \xi = 0 \\ a\gamma + c\delta + \eta = 0 \end{cases} \\ T(b, c)^T = (1, 0)^T &\Rightarrow \begin{cases} b\alpha + c\beta + \xi = 1 \\ b\gamma + c\delta + \eta = 0 \end{cases} \\ T(a, d)^T = (0, 1)^T &\Rightarrow \begin{cases} a\alpha + d\beta + \xi = 0 \\ a\gamma + d\delta + \eta = 1 \end{cases} \\ T(b, d)^T = (1, 1)^T &\Rightarrow \begin{cases} b\alpha + d\beta + \xi = 1 \\ b\gamma + d\delta + \eta = 1, \end{cases} \end{aligned}$$

iz čega slijedi

$$\alpha = \frac{1}{b-a}, \quad \beta = 0, \quad \gamma = 0, \quad \delta = \frac{1}{d-c}, \quad \xi = \frac{-a}{b-a}, \quad \eta = \frac{-c}{d-c},$$

odnosno

$$A = \begin{bmatrix} \frac{1}{b-a} & 0 \\ 0 & \frac{1}{d-c} \end{bmatrix}, \quad b = \begin{bmatrix} \frac{-a}{b-a} \\ \frac{-c}{d-c} \end{bmatrix}.$$

Kako je $b = -Au$, slijedi (5.3).

Formula (5.4) dokazuje se direktnom provjerom. □

Transformacija Branin funkcije iz Primjera 5.1 na kvadrat $[0, 1]^2$ daje funkciju $f: [0, 1]^2 \rightarrow \mathbb{R}$, gdje je $f = g \circ T^{-1}$, $T^{-1}(x) = A^{-1}(x + u)$, $A^{-1} = \begin{bmatrix} 15 & 0 \\ 0 & 15 \end{bmatrix}$, $u = [-5, 0]^T$.

Zadatak 5.1. Neka su $a_i, b_i, i = 1, \dots, n, (n \geq 2)$, realni brojevi takvi da je $b_i > a_i, \forall i = 1, \dots, n$. Definirajte preslikavanje $T: \mathbb{R}^n \rightarrow \mathbb{R}^n$, koje hiperpravokutnik $\prod_{i=1}^n [a_i, b_i]$ preslikava u jedinični hiperkvadrat $[0, 1]^n$.

Primjedba 5.1. Primijetite da se hiperpravokutnik u \mathbb{R}^n može zapisati na sljedeći način. Za dvije različite točke $a, b \in \mathbb{R}^n$ hiperpravokutnik $[a, b] \subset \mathbb{R}^n$ može se zapisati s $[a, b] = \{x \in \mathbb{R}^n: a_i \leq x_i \leq b_i, i = 1, \dots, n\}$ ili jednostavnije s $[a, b] = \{x \in \mathbb{R}^n: a \leq x \leq b\}$.

5.2 Optimizacijski algoritam DIRECT za funkciju dviju varijabli

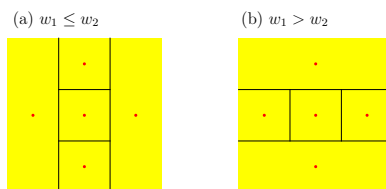
Zadana je Lipschitz neprekidna funkcija $g: [a, b] \times [c, d] \rightarrow \mathbb{R}$. Primjenom transformacije T zadane s (5.3), umjesto rješavanja optimizacijskog problema za funkciju g rješavat ćemo optimizacijski problem za funkciju

$$f = g \circ T^{-1}: [0, 1]^2 \rightarrow \mathbb{R}, \quad \text{gdje je } T^{-1}(x) = A^{-1}x + u. \quad (5.5)$$

Uočite da će i tako definirana funkcija f biti Lipschitz neprekidna (vidi Zadatak 4.7 i Zadatak 4.2 (e)).

5.2.1 Dijeljenje kvadrata $[0, 1]^2$

Slično kao što smo u t.4, str.69 definirali strategiju dijeljenja intervala, u ovom slučaju potrebno je razraditi strategiju dijeljenja kvadrata. Jedinični kvadrat $[0, 1]^2$ iterativnim postupkom dijelit ćemo na subpravokutnike prema niže navedenom postupku i pravilima.



Slika 5.3: Dijeljenje kvadrata $[0, 1] \times [0, 1]$

Postupak počinje dijeljenjem jediničnog kvadrata $[0, 1]^2$ s centrom u točki $c = \left(\frac{1}{2}, \frac{1}{2}\right)$ na način opisan na Slici 5.3a ako je

$$w_1 := \min\{f(c - \delta e_1), f(c + \delta e_1)\} \leq \min\{f(c - \delta e_2), f(c + \delta e_2)\} =: w_2, \quad (5.6)$$

gdje je $\delta = \frac{1}{3}$, a e_1 i e_2 standardni bazni vektori u \mathbb{R}^2 , odnosno na način opisan na Slici 5.3b ako je

$$w_1 > w_2. \quad (5.7)$$

Tako dobivamo skup od pet subpravokutnika $\mathcal{P} = \{P_1, \dots, P_5\}$ s centrima

$$\begin{aligned} c_1 &= \left(\frac{1}{2}, \frac{1}{2}\right), & c_2 &= \left(\frac{1}{2} - \delta, \frac{1}{2}\right), & c_3 &= \left(\frac{1}{2} + \delta, \frac{1}{2}\right), \\ c_4 &= \left(\frac{1}{2}, \frac{1}{2} - \delta\right), & c_5 &= \left(\frac{1}{2}, \frac{1}{2} + \delta\right). \end{aligned}$$

Također, svakom subpravokutniku pridružujemo duljine njegovih stranica u slučaju (5.6) (Slika 5.3a)

$$h^{(1)} = (\delta, \delta), \quad h^{(2)} = (\delta, 1), \quad h^{(3)} = (\delta, 1), \quad h^{(4)} = (\delta, \delta), \quad h^{(5)} = (\delta, \delta),$$

odnosno duljine njegovih stranica u slučaju (5.7) (Slika 5.3b)

$$h^{(1)} = (\delta, \delta), \quad h^{(2)} = (\delta, \delta), \quad h^{(3)} = (\delta, \delta), \quad h^{(4)} = (1, \delta), \quad h^{(5)} = (1, \delta).$$

Ako je $L > 0$ Lipschitzova konstanta funkcije f , \mathcal{B} -vrijednost pravokutnika P_i s centrom c_i i veličinom d_i definiramo s $\mathcal{B}_i = \mathcal{B}(c_i, d_i) = f(c_i) - Ld_i$, gdje d_i predstavlja neku mjeru veličine pravokutnika P_i . Budući da je cilj pronaći i dalje dijeliti subpravokutnik s najmanjom \mathcal{B} -vrijednosti, kao mjeru veličine subpravokutnika P_i sa stranicama $h_1^{(i)}, h_2^{(i)}$ treba uzeti

$$d_i = \max\{h_1^{(i)}, h_2^{(i)}\}, \quad i = 1, \dots, 5. \quad (5.8)$$

Naime, ako je primjerice $h_1^{(i)} > h_2^{(i)}$, onda se manja \mathcal{B} -vrijednost dobiva korištenjem dulje stranice

$$\mathcal{B}_1^{(i)} = f(c) - Lh_1^{(i)} < f(c) - Lh_2^{(i)} = \mathcal{B}_2^{(i)}.$$

Tako u slučaju (5.6) (Slika 5.3a) imamo

$$d_1 = \delta, \quad d_2 = 1, \quad d_3 = 1, \quad d_4 = \delta, \quad d_5 = \delta,$$

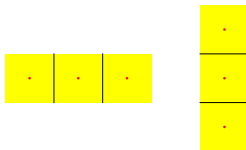
a u slučaju (5.7) (Slika 5.3b) imamo

$$d_1 = \delta, \quad d_2 = \delta, \quad d_3 = \delta, \quad d_4 = 1, \quad d_5 = 1.$$

Primjedba 5.2. Primijetite da je \mathcal{B} -vrijednost svakog pravokutnika P_i određena vrijednošću funkcije u centru c_i i duljinom d_i njegove veće stranice. Također je važno primijetiti da smo za mjeru veličine pravokutnika mogli uzeti i nešto drugo (primjerice, udaljenost centra do vrha, tj. duljinu poludijagonale) – to nam dozvoljavaju zapisi uvjeta u Lemi 4.2! Ova je činjenica važna ako područje dijelimo na trokute [49]. Tada bismo za d_i uzimali maksimalnu udaljenost od centra (težišta) trokuta do vrha.

Nadalje, svaki subkvadrat (jednake stranice) dijelit ćemo također prema prethodno opisanom postupku, a svaki subpravokutnik (različite stranice) na tri jednaka dijela po većoj stranici kao na Slici 5.4.

Na taj način u nekom trenutku raspoložemo skupom \mathcal{P} od m subpravokutnika. Za svaki od njih odredit ćemo njegov centar c_i i veličinu d_i kao poluduljinu veće stranice.



Slika 5.4: Dijeljenje subpravokutnika

Trenutnu aproksimaciju \hat{c} točke globalnog minimuma odredit ćemo kao

$$\hat{c} = \underset{i=1, \dots, m}{\operatorname{argmin}} f(c_i), \quad f_{\min} = f(\hat{c}). \quad (5.9)$$

Uočimo da problemu traženja subpravokutnika minimalne \mathcal{B} -vrijednosti možemo pristupiti jednako kao u t.4.4.1, str.89 i promatrati potencijalno optimalne subpravokutnike definirane analogno Definiciji 4.2, str.90. Korištenjem rezultata Teorema 4.2, str.96 možemo na taj način iz skupa \mathcal{P} izdvojiti *potencijalno optimalne subpravokutnike* i samo na njima provesti daljnji postupak dijeljenja. Time smo izbjegli korištenje Lipschitzove konstante funkcije f u algoritmu.

Tijekom iterativnog postupka skup \mathcal{P} proširuje se novim subpravokutnicima, a minimum f_{\min} korigira se uvijek samo na centrima novih subpravokutnika.

Problem zaustavljanja iterativnog postupka u literaturi nije egzaktno riješen. Jedna mogućnost za zaustavljanje iterativnog postupka je kontroliranje trenutne veličine d subpravokutnika na kojemu se postiže trenutni minimum.

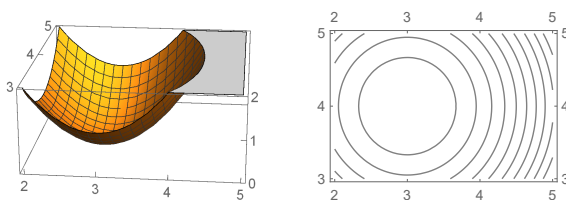
Nakon što odredimo prihvatljivu aproksimaciju $\{\hat{c}, f_{\min}\}$ globalnog minimuma funkcije f , primjenom inverznog preslikavanja T^{-1} iz Leme 5.1, str.102, aproksimaciju globalnog minimuma polazne funkcije g određujemo kao $T^{-1}(\hat{c})$.

Opisani algoritam formaliziran je u Algoritmu 13, str.111 za funkciju n nezavisnih varijabli.

5.2.2 Nekoliko primjera

Korištenjem programskog sustava *Mathematica* s nekoliko primjera ilustrirat ćemo Algoritam DIRECT za traženje globalnog minimuma funkcije dviju varijabli uz odgovarajući postupak dijeljenja jediničnog kvadrata $[0, 1]^2$.

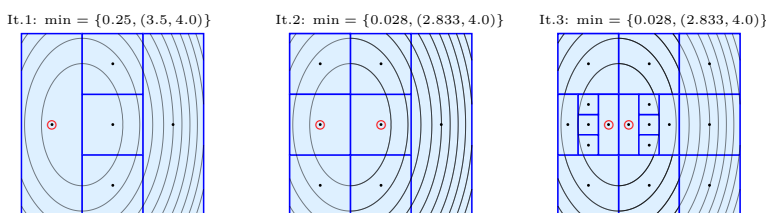
Primjer 5.2. Za početak promatrajmo sasvim jednostavnu kvadratnu funkciju $g: [2, 5] \times [3, 5] \rightarrow \mathbb{R}$, $g(x_1, x_2) = (x_1 - 3)^2 + (x_2 - 4)^2$, koja svoj globalni minimum postiže u točki $u^* = (3, 4)$, $g(u^*) = 0$.



Slika 5.5: Kvadratna funkcija $g: [2, 5] \times [3, 5] \rightarrow \mathbb{R}$, $g(x_1, x_2) = (x_1 - 3)^2 + (x_2 - 4)^2$

Ploha i odgovarajući ContourPlot ove funkcije prikazani su na Slici 5.5. Primjenom preslikavanja $T(x) = A(x - u)$, gdje je $A = \begin{bmatrix} 1/3 & 0 \\ 0 & 1/2 \end{bmatrix}$, $u = [2, 3]^T$, $T^{-1}(x) = A^{-1}(x) + u$, dobivamo funkciju $f: [0, 1]^2 \rightarrow \mathbb{R}$, $f(x) = (g \circ T^{-1})(x) = g(3x_1 + 2, 2x_2 + 3) = (3x_1 - 1)^2 + (2x_2 - 1)^2$, koja svoj globalni minimum postiže u točki $\hat{u} = (\frac{1}{3}, \frac{1}{2})$, $f(\hat{u}) = 0$.

Prve tri iteracije dijeljenja jediničnog kvadrata $[0, 1]^2$ za ovu funkciju te odgovarajući trenutni minimum prikazani su na Slici 5.6. Crvenim kružićima naznačeni su centri potencijalno optimalnih subpravokutnika.

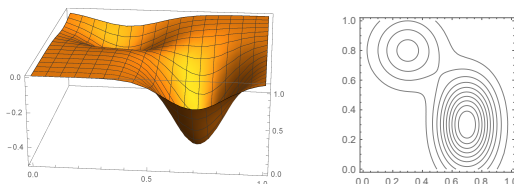


Slika 5.6: Dijeljenje kvadrata $[0, 1]^2 \subset \mathbb{R}^2$ u slučaju funkcije iz Primjera 5.2. Crvenim kružićima označeni su centri potencijalno optimalnih pravokutnika

Primjer 5.3. Konstruirajmo funkciju $f: [0, 1]^2 \rightarrow \mathbb{R}$,

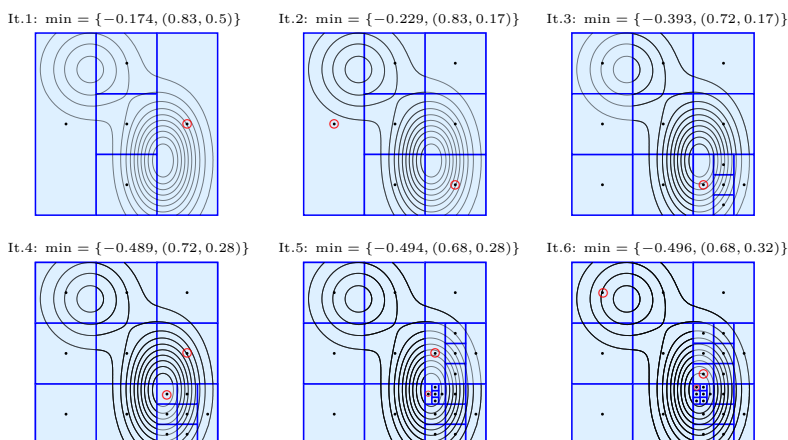
$$f(x_1, x_2) = -\frac{1}{2}e^{-10\|A([x_1, x_2]^T - [7, 3]^T)\|^2} - \frac{1}{5}e^{-20\| [x_1, x_2]^T - [0.3, 0.8]^T \|^2}, \quad (5.10)$$

gdje je $A = .8 \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$, koja će na $[0, 1]^2$ postići lokalni minimum u točki $u_1^* = (0.3, 0.8)$, $f(u_1^*) = -0.2$ i globalni minimum u točki $u_2^* = (0.7, 0.3)$, $f(u_2^*) = -0.5$.



Slika 5.7: $f(x_1, x_2) = -\frac{1}{2}e^{-10\|A([x_1, x_2]^T - [.7, .3]^T)\|^2} - \frac{1}{5}e^{-20\|[x_1, x_2]^T - [0.3, 0.8]^T\|^2}$

Ploha i odgovarajući `ContourPlot` prikazani su na Slici 5.7. Prve tri iteracije dijeljenja jediničnog kvadrata $[0, 1]^2$ za ovu funkciju te odgovarajući trenutni minimum prikazani su na Slici 5.8.



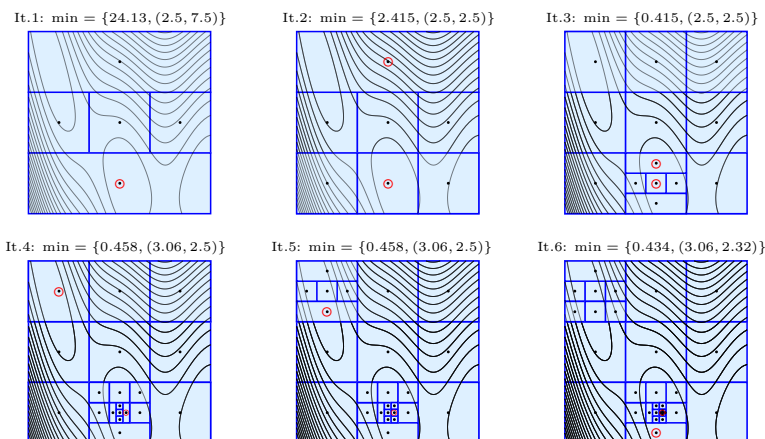
Slika 5.8: Dijeljenje kvadrata $[0, 1]^2 \subset \mathbb{R}^2$ u slučaju funkcije (5.10) iz Primjera 5.3. Crvenim kružićima označeni su centri potencijalno optimalnih pravokutnika

Zadatak 5.2. Pokažite da funkcija f zadana s (5.10) ima dvije stacionarne točke $u_1^* = (0.3, 0.8)$ i $u_2^* = (0.7, 0.3)$ i da je $f(u_1^*) = -0.2$, $f(u_2^*) = -0.5$.

Primjer 5.4. Pokažimo još primjenu optimizacijskog Algoritma *DIRECT* za Branin funkciju iz Primjera 5.1, koja svoj globalni minimum $g_{min} = 0.398$ postiže u trima različitim točkama. Jedna od njih je $u_1^* = (\pi, 2.275)$ (vidi

Sliku 5.1).

Prvih šest iteracija dijeljenja jediničnog kvadrata $[0, 1]^2$ za ovu funkciju te odgovarajući trenutni minimum prikazani su na Slici 5.9.



Slika 5.9: Dijeljenje kvadrata $[0, 1]^2 \subset \mathbb{R}^2$ u slučaju Branin funkcije iz Primjera 5.1. Crvenim kružićima označeni su centri potencijalno optimalnih pravokutnika

Zadatak 5.3. Zadana je funkcija $g: [-1, 1] \times [-1, 1] \rightarrow \mathbb{R}$,

$$g(x_1, x_2) = \max \{ |2x_1 - x_2 - 1|, |x_1 + x_2| \}.$$

- (a) Odredite globalni minimizator funkcije g .
- (b) Odredite funkciju f dobivenu transformacijom funkcije g na jedinični kvadrat. Što je globalni minimizator tako dobivene funkcije?
- (c) Napravite dvije iteracije *Algoritma* DIRECT te skicirajte kako bi se dalje dijelili pravokutnici.
- (d) Koja je aproksimacija globalnog minimizatora polazne funkcije g ?

5.3 Optimizacijski algoritam DIRECT za funkciju više varijabli

Zadana je Lipschitz neprekidna funkcija $g: \mathcal{D} \rightarrow \mathbb{R}$, gdje je $\mathcal{D} \subset \mathbb{R}^n$ hiperpravokutnik u \mathbb{R}^n . Primjenom transformacije T (Zadatak 5.1, str.104),

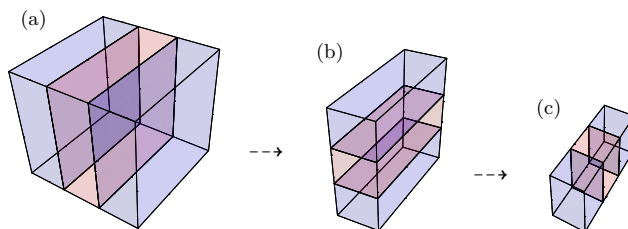
umjesto rješavanja optimizacijskog problema za funkciju g rješavat ćemo optimizacijski problem za funkciju

$$f = g \circ T^{-1}: [0, 1]^n \rightarrow \mathbb{R}, \quad \text{gdje je } T^{-1}(x) = A^{-1}x + u. \quad (5.11)$$

Slično kao što smo u t.4, str.69 definirali strategiju dijeljenja intervala, u ovom slučaju potrebno je razraditi strategiju dijeljenja hiperkocke. Jedinичnu hiperkocku $[0, 1]^n$ iterativnim ćemo postupkom dijeliti na subhiperpravokutnike prema niže navedenom postupku i pravilima.

Primjedba 5.3. Primijetite da u slučaju $n = 3$ kocku dijelimo na $2n + 1$ subparalelepipeda na sljedeći način (vidi [26], Slika 8, str. 169):

- po prvoj izabranoj koordinati (izbor ovisi o brojevima w_1, w_2, w_3 koji su definirani analogno (5.6)) kocku dijelimo na 3 jednaka subparalelepipeda (sloja) (vidi Sliku 5.10a);
- po drugoj izabranoj koordinati centralni subparalelepiped (sloj) dijelimo na 3 jednaka subparalelepipeda (sloja) (vidi Sliku 5.10b);
- po trećoj koordinati centralni subparalelepiped (sloj) dijelimo na 3 jednake subkocke (vidi Sliku 5.10c).



Slika 5.10: Dijeljenje kocke u \mathbb{R}^3

Nadalje, uvijek samo kocku dijelimo na ovaj način. Subparalelepipede, kakvi god bili, uvijek dijelimo na tri jednaka dijela po najdužoj stranici.

Niže navodimo pseudokod *Algoritma DIRECT*, a odgovarajući *Mathematica*-modul može se preuzeti na adresi <http://www.mathos.unios.hr/GOP-knjiga/GOPmoduli.rar>

Algoritam 13 (Algoritam DIRECT)**Input:** $f: \mathbb{R}^n \rightarrow \mathbb{R}$;**Input:** c_{min} ; $d = 0.5$; $0 < \eta < d$; $f_{min} = f(c_{min})$;

- 1: Neka je \mathcal{H} trenutni skup hiperpravokutnika;
- 2: Grupiraj sve hiperpravokutnike iz \mathcal{H} prema njihovoj veličini, a u svakoj grupi sortiraj ih prema vrijednosti funkcije f u njihovom centru;
- 3: Hiperpravokutnici iz svake grupe s najmanjom funkcijskom vrijednosti čine prošireni skup potencijalno optimalnih hiperpravokutnika \mathcal{E} ;
- 4: Primjenom Teorema 4.2 iz skupa \mathcal{E} , sagradi skup potencijalno optimalnih hiperpravokutnika \mathcal{P} ;
- 5: **for** $R \in \mathcal{P}$ **do**
- 6: Podijeli R na subhiperpravokutnike r_1, \dots, r_s ;
- 7: Izbriši hiperpravokutnik R
- 8: **for** $i = 1, \dots, s$ **do**
- 9: Odredi veličinu d_i , centar c_i i funkcijsku vrijednost $f(c_i)$;
- 10: Korigiraj minimalnu veličinu d i $(c_{min}, f(c_{min}))$
- 11: **end for**
- 12: **end for**
- 13: **if** $d > \eta$, **go to** Step 2; **else** STOP

Output: $(c_{min}, f(c_{min}))$

Primjer 5.5. U Primjeru 1.10, str.15 zadan je skup (konačni niz) $\mathcal{A} = \{a^i \in \mathbb{R} : i = 1, \dots, 262144\}$ realnih brojeva koji predstavljaju tonove (gray levels) točaka poznate crno-bijele slike „Elaine” veličine 512×512 .

Skup \mathcal{A} grupirali smo u $k = 2, 4, 8$ klastera minimizirajući funkciju općenito k varijabli

$$F(c_1, \dots, c_k) = \sum_{i=1}^m \min_{1 \leq j \leq k} d_{LS}(c_j, a^i), \quad (5.12)$$

gdje je $d_{LS}(x, y) = \|x - y\|_2^2$, LS-kvazimetrička funkcija. Problem $\operatorname{argmin}_{c_1, \dots, c_k \in \mathbb{R}} F(c_1, \dots, c_k)$ rješava se Algoritmom 13 korištenjem modula dostupnog na adresi <http://www.mathos.unios.hr/GOP-knjiga/GOPmoduli.rar>.

Brojne druge primjere problema globalne optimizacije i test funkcije za globalnu optimizaciju možemo pronaći u [13, 14, 16, 23, 50, 63].

5.4 Optimizacijski algoritam DIRECT za simetričnu funkciju

Funkcija $f: [a, b]^n \rightarrow \mathbb{R}$ je simetrična ako proizvoljna permutacija njenih nezavisnih varijabli ne mijenja vrijednost funkcije. Simetrične funkcije često se javljaju u različitim primijenjenim istraživanjima. Primjerice, funkcija

(5.12) simetrična je funkcija jer proizvoljna permutacija nezavisnih varijabli c_1, \dots, c_n znači samo drugačiji poredak centara klastera, ali ne mijenja vrijednost funkcije.

Primjer 5.6. *Funkcija $f: \mathbb{R}^2 \rightarrow \mathbb{R}$, $f(x_1, x_2) = x_1^2 - x_1x_2 + x_2^2$ simetrična je funkcija, ali funkcija $g: \mathbb{R}^2 \rightarrow \mathbb{R}$, $g(x_1, x_2) = x_1 - x_1x_2 + x_2^2$ nije.*

Zadatak 5.4. Primjenom programskog sustava *Mathematica* nacrtajte graf (plohu) i `ContourPlot` funkcije [16]

$$g: [0, 11]^2 \rightarrow \mathbb{R}, \quad g(x, y) = -.2(x^2 + y^2) + 2xy \cos x \cos y.$$

Je li ovo simetrična funkcija?

Primjer 5.7. *Neka je $\mathcal{A} \subset \mathbb{R}$ skup dobiven Mathematica-naredbom:*

```
In[1]:= m = 20; SeedRandom[13];
        A = RandomReal[{0, 1}, m]
```

```
Out[1]= {0.456535, 0.868230, 0.704274, 0.795001, 0.040520, 0.957827, 0.008372,
         0.251257, 0.014313, 0.743946, 0.066294, 0.783009, 0.907372, 0.081007,
         0.486618, 0.824774, 0.684515, 0.063848, 0.086283, 0.658425}
```

Elemente skupa \mathcal{A} označit ćemo s a^i , $i = 1, \dots, m$ i grupirati ih u dva klastera minimizacijom funkcije (5.12).

Područje definicije minimizirajuće funkcije (5.12) je jedinični kvadrat $[0, 1]^2$, a za traženje globalnog minimuma direktno se može primijeniti Algoritam 13 ili *Mathematica*-modul `NMinimize`. Dobivamo

```
In[1]:= Clear[x, y]; NMinimize[{F[x, y], 0 <= x <= 1, 0 <= y <= 1}, {x, y}]
```

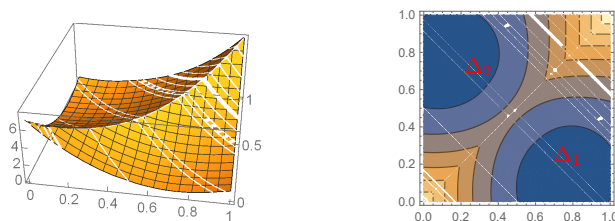
```
Out[2]= {0.300262, {x -> 0.739211, y -> 0.0764867}}
```

Graf (ploha) i `ContourPlot` odgovarajuće funkcije (5.12) prikazani su na Slici 5.11. Primijetite da je područje definicije simetrično podijeljeno na dva trokuta

$$\Delta_1 = \{(x_1, x_2) \in [0, 1]^2: 1 \geq x_1 \geq x_2 \geq 0\},$$

$$\Delta_2 = \{(x_1, x_2) \in [0, 1]^2: 0 \leq x_1 \leq x_2 \leq 1\},$$

a *Mathematica*-modul `NMinimize` pronašao je samo točku globalnog minimuma $u^* = (0.73921, 0.07648)$ u trokutu Δ_1 . Jasno je da postoji i druga točka globalnog minimuma $v^* = (0.07648, 0.73921) \in \Delta_2$.



Slika 5.11: $F(c_1, c_2) = \sum_{i=1}^m \min\{\|c_1 - a^i\|_2^2, \|c_2 - a^i\|_2^2\}$

Situacija koja se pojavila u prethodnom primjeru prirodno se generalizira i u slučaju simetrične Lipschitz-neprekidne funkcije više varijabli. Ako simetrična Lipschitz-neprekidna funkcija s n nezavisnih varijabli ima globalni minimum, tada se on može postići u barem $n!$ različitih točaka. Primjerice, ako bismo skup $\mathcal{A} \subset \mathbb{R}^n$ grupirali u 5 klastera, funkcija (5.12) postigla bi svoj globalni minimum u barem 120 različitih točaka, čije se koordinate razlikuju do na permutaciju. Pet koordinata bilo koje od ovih točaka predstavljaju centre pet klastera skupa \mathcal{A} . Algoritam DIRECT pronalazi svih 120 točaka i zbog toga je vrijeme izvođenja (CPU time) izuzetno veliko.

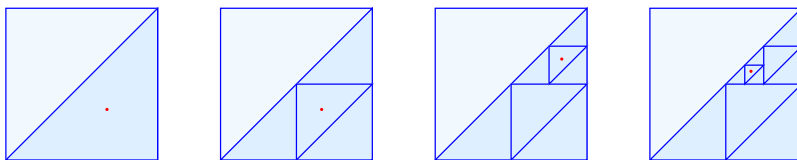
Zato se u literaturi [16] pojavila specijalna metoda i algoritam SymDIRECT kojim se efikasno rješava globalni optimizacijski problem (GOP) za simetričnu Lipschitz-neprekidnu funkciju. Metoda SymDIRECT jedna je modifikacija standardnog Algoritma DIRECT, a najkraće rečeno, umjesto traženja točke globalnog minimuma na hiperkocki $[0, 1]^n$, on se traži samo na jednom od $n!$ jednakih hipertetraedara. Zbog jednostavnosti izabran je hipertetraedar

$$\Delta = \{(x_1, \dots, x_n) \in [0, 1]^n : 1 \geq x_1 \geq \dots \geq x_n \geq 0\}.$$

U postupku dijeljenja hiperkocke $[0, 1]^n$ Algoritam SymDIRECT zadržava samo one subhiperpravokutnike koji su potpuno ili djelomično sadržani u hipertetraedru Δ . Metoda pokazuje visoku efikasnost kod rješavanja GOP za simetričnu Lipschitz-neprekidnu funkciju.

Godinu dana kasnije pojavila se nova, još efikasnija metoda i algoritam DISIMPL [49]. Ideja ove metode sastoji se u tome da umjesto dijeljenja hiperkocke $[0, 1]^n$ posebnim postupkom dijelimo hipertetraedar Δ . U najjednostavnijem slučaju ($n = 2$), hipertetraedar Δ donji je trokut jediničnog kvadrata $[0, 1]^2$ (Slika 5.12a), koji se u prvom koraku dijeli na 4 jednaka (sukladna) subtrokuta od kojih središnji ima centar (težište) u centru polaznog trokuta, a njegovi vrhovi podudaraju se s polovištima stranica polaznog tro-

kuta. Nakon određivanja \mathcal{B} -vrijednosti svih subtrokuta, dalje se dijeli onaj s najmanjom \mathcal{B} -vrijednosti. Naravno, i u ovom slučaju koriste se potencijalno optimalni subtrokuti. Jedno moguće dijeljenje trokuta Δ prikazano je na Slici 5.12.



Slika 5.12: Dijeljenje trokuta $\Delta = \{(x_1, x_2) \in [0, 1]^2 : 1 \geq x_1 \geq x_2 \geq 0\}$

Primjer 5.8. Neka je $\mathcal{A} = \{a^i \in [0, 1] : i = 1, \dots, m\} \subset \mathbb{R}$ skup konstruiran u Primjeru 5.7. Elemente skupa \mathcal{A} grupirat ćemo u tri klastera minimizacijom funkcije (5.12).

Primjenom *Mathematica*-modula `NMinimize` dobivamo

```
In[1]:= Clear[x, y, z];
        NMinimize[{F[x,y,z], 0<=x<=1, 0<=y<=1, 0<=z<=1}, {x,y,z}]

Out[2]= {0.125808, {x -> 0.792737, y -> 0.0515195, z -> 0.398137}}
```



Slika 5.13: Jedinična kocka $[0, 1]^3$ prirodno je podijeljena na 6 jednakih tetraedara

Na ovaj GOP primijenit ćemo i *Algoritam SymDIRECT*. Područje definicije minimizirajuće funkcije (5.12) u ovom je slučaju jedinična kocka $[0, 1]^3$ koja je prirodno podijeljena na 6 jednakih tetraedara (vidi Sliku 5.13¹). Sukladno *Metodi SymDIRECT*

¹Maketu su izradili Lidija Radan i Jan Valenta, Odjel za matematiku, Sveučilište u Osijeku.

dovoljno je promatrati samo jedan od 6 tetraedara i na njemu pronaći točke minimuma. Promatrajmo tetraedar

$$\Delta_1 = \{(x_1, x_2, x_3) \in [0, 1]^3 : 1 \geq x_1 \geq x_2 \geq x_3 \geq 0\}.$$

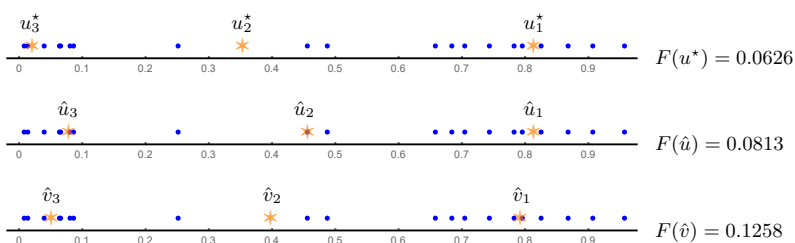
Algoritam SymDIRECT daje globalni minimum (vidi Sliku 5.14)

$$u^* = (0.81385, 0.35390, 0.02106) \in \Delta_1, \quad F(u^*) = 0.06258.$$

Mathematica-modul NMinimize pronašao je jednu od dvije točke lokalnog minimuma (vidi također Sliku 5.14).

$$\hat{u} = (0.81386, 0.45654, 0.07861) \in \Delta_1, \quad F(\hat{u}) = 0.08126,$$

$$\hat{v} = (0.79274, 0.05152, 0.39814) \in \Delta_1, \quad F(\hat{v}) = 0.12581.$$



Slika 5.14: Jedan globalni $(u^*, F(u^*))$ i dva lokalna minimuma $(\hat{u}, F(\hat{u}))$, $(\hat{v}, F(\hat{v}))$ iz Primjera 5.7

Zadatak 5.5. Neka je $\mathcal{A} \subset \mathbb{R}$. Konstruirajte algoritam po uzoru na *Algoritam DIRECT* koji će tražiti točku globalnog minimuma simetrične Lipschitz-neprekidne funkcije definirane na trokutu $\Delta = \{(x_1, x_2) \in [0, 1]^2 : 1 \geq x_1 \geq x_2 \geq 0\}$ primjenom prethodno spomenutog postupka dijeljenja trokuta.

Još jedna nova metoda [63] za traženje točke globalnog minimuma simetrične Lipschitz-neprekidne funkcije $g: [a, b]^n \rightarrow \mathbb{R}$ nakon transformacije $f = g \circ T^{-1}$ također pretražuje hipertetraedar Δ . Primjenom nove transformacije \mathcal{T} (vidi Lemu 5.2) problem traženja točke globalnog minimuma funkcije f na hipertetraedru Δ prevodi se na traženje točke globalnog minimuma nove funkcije $\Phi: [0, 1]^n \rightarrow \mathbb{R}$, $\Phi(\zeta) = f(\mathcal{T}(\zeta))$ na hiperkocki $[0, 1]^n$, a tu je moguće primijeniti standardni DIRECT ili neki drugi optimizacijski algoritam (vidi t.5.5.2, str.117).

Lema 5.2. Neka je $\mathcal{T}: \mathbb{R}^n \rightarrow \mathbb{R}^n$ linearni operator zadan s $\mathcal{T}(e_i) = \sum_{j=1}^i e_j$, gdje je $\{e_1, \dots, e_n\}$ standardna ortonormirana baza u \mathbb{R}^n . Tada vrijedi

$$c \in \Delta \quad \Leftrightarrow \quad c \in [0, 1]^n \quad \text{i} \quad \mathcal{T}^{-1}(c) \in [0, 1]^n \quad .$$

Zadatak 5.6. Dokažite Lemu 5.2.

5.5 Neke druge mogućnosti

5.5.1 Kratki povijesni pregled

Neka je $f: \mathcal{D} \rightarrow \mathbb{R}$, $\mathcal{D} \subseteq \mathbb{R}^n$, proizvoljna funkcija koja u točki $x^* = \operatorname{argmin}_{x \in \mathcal{D}} f(x)$ postiže globalni minimum $f^* = f(x^*)$. Kao što je već i ranije navedeno, problem određivanja takve točke $x^* \in \mathcal{D}$ zovemo *problem globalne optimizacije*. Metode za rješavanje problema globalne optimizacije tijekom povijesti² najviše su se razvijale pod utjecajem razvoja inženjerskih tehnologija. Prvi poznati znanstveni rad u literaturi povezan s globalnom optimizacijom potječe iz 1797. godine³. U tom je radu prvi puta sustavno analitički obrađen problem globalne optimizacije. Analitičke metode za rješavanje problema globalne optimizacije općenito su vrlo kompleksne, posebno u slučaju velikog broja varijabli te zbog toga prije pojave suvremenih računala njihova efikasna implementacija bila je gotovo nemoguća. U kasnijim 50.-tim godinama 20. stoljeća razvijene su prve numeričke metode za rješavanje problema globalne optimizacije koje su se zasnivale na tzv. načelu *podijeli-pa-vladaj*. Ugrubo govoreći, ovo se načelo sastoji od podjele domene \mathcal{D} na manje poddomene na kojima se mogla primijeniti neka lokalna metoda. Na taj način generira se iterativni postupak koji konvergira prema točki globalnog minimuma. Tipičan primjer ovakve metode je *Metoda grananja i ograđivanja* (eng. *Branch and Bound* - (BB)) [47], koja se u svom originalu koristila za rješavanje specifičnih problema globalne optimizacije povezane s linearnim odnosno cjelobrojnim programiranjem. *Metoda grananja i ograđivanja* primijenjena na općenitu funkciju prvi se puta koristi u radu [12]. U razdoblju od 1970. do 1980. najvećim su se dijelom razvijale metode za rješavanje mješovitih cjelobrojnih i nelinearnih problema [37]. U tom je razdoblju također razvijena metoda ovog tipa koja se temelji na intervalnoj

²Podaci o kratkom povijesnom pregledu metoda globalne optimizacije preuzeti su uglavnom iz [37].

³Iz povijesnih razloga navodimo originalnu referencu:

J. L. Lagrange, *Théorie des fonctions analytiques*, Impr. de la République, Paris, 1797.

aritmetici [18, 19]. Metoda se nije pokazala efikasnom zbog svoje općenito vrlo spore konvergencije.

Do kraja 80.-ih godina razvijene su različite metode globalne optimizacije koje su koristile specijalne strukture problema, pa i *stohastička optimizacija* (vidi primjerice [78]). Početkom 90.-ih godina, broj znanstvenih publikacija u području globalne optimizacije značajno se povećao. Također se razvija veliki broj softverskih paketa za rješavanje ovog problema. U tom razdoblju izdana je prva knjiga vezana uz globalnu optimizaciju [21].

U radu [2] *Metoda grananja i ograđivanja* pod nazivom αBB primijenjena je na općeniti nelinearni i nekonveksni problem uvjetne optimizacije. Nakon toga uslijedio je niz znanstvenih radova u kojima se razmatra ova metoda, a temeljna pretpostavka je da je funkcija cilja dva puta neprekidno diferencijabilna. Nešto kasnije razvijaju se stohastičke metode, kao što su *Simulated and Nested Annealing* [53], *Tabu Search* [8], *Multi-Level Single Linkage* [33], *Variable Neighbourhood Search* [42], *Differential Evolution*, itd.

5.5.2 Metoda grananja i ograđivanja

U nastavku ćemo ukratko proanalizirati prethodno spomenutu *Metoda grananja i ograđivanja*. Temeljna ideja ove metode sastoji se od definiranja liste \mathcal{L} konveksnih podskupova domene \mathcal{D} , pri čemu pretpostavljamo da barem jedan od tih podskupova sadrži točku globalnog minimuma funkcije f . Iz liste \mathcal{L} odabiremo podskup R , kojega obrišemo iz liste, te na tom podskupu određujemo gornju $\Phi_{UB}(R)$ i donju $\Phi_{LB}(R)$ među optimalne vrijednosti funkcije f^* tako da vrijedi

$$\Phi_{LB}(R) \leq \min_{x \in R} f(x) \leq \Phi_{UB}(R).$$

Ovaj postupak zovemo *ograđivanje*.

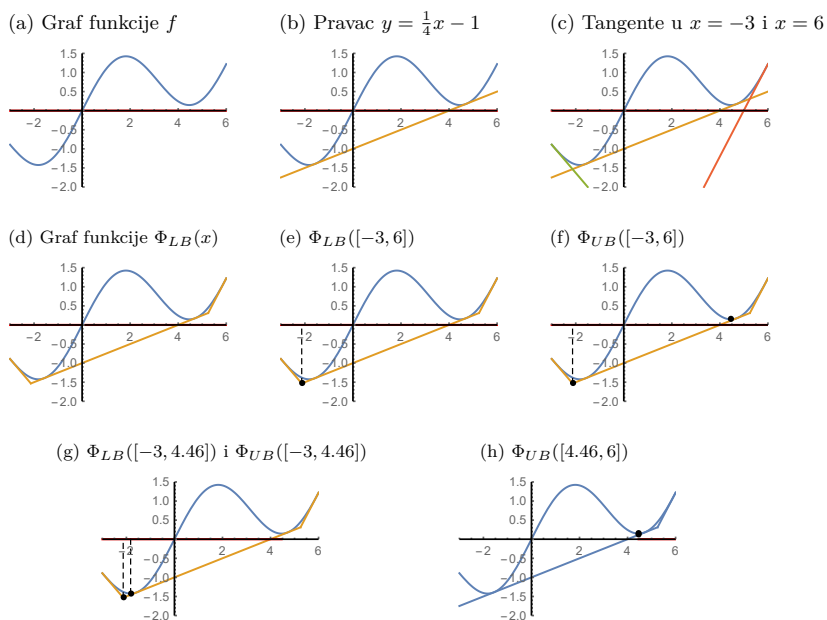
U cilju osiguravanja brzine konvergencije, gornju i donju među važno je odrediti na što je moguće jednostavniji način. Ako je pritom $\Phi_{UB}(R) - \Phi_{LB}(R) < \varepsilon$, gdje je $\varepsilon > 0$ unaprijed zadana točnost, podskup R zadržavamo te odgovarajuću vrijednost gornje međe $\Phi_{UB}(R)$ smatramo trenutnom vrijednošću globalnog minimuma. Također, iz liste \mathcal{L} uklonimo sve one podskupove kojima je vrijednost donje međe veća od $\Phi_{UB}(R)$.

U suprotnom, ako je $\Phi_{UB}(R) - \Phi_{LB}(R) > \varepsilon$, podskup R dalje dijelimo u disjunktne podskupove koje uvrštavamo u listu \mathcal{L} . U listi \mathcal{L} zadržavamo samo one podskupove kojima je vrijednost gornje međe manja od trenutne vrijednosti globalnog minimuma. Postupak daljnjeg dijeljenja nazivamo *grananje*.

U ovisnosti o načinu konstrukcije gornje i donje međe (ograđivanje), kao i načinu daljnjeg dijeljenja podskupa (grananja) mogu se konstruirati različite metode koje su zasnovane na ovoj ideji. Tipični primjer ove grupe metoda je već ranije spomenuta αBB metoda, kod koje se u svrhu definiranja gornje međe koristi pretpostavka da je funkcija f dva puta neprekidno diferencijabilna [2].

U nastavku kao ilustraciju navodimo primjer iz [37], u kojemu rješavamo problem jednodimenzionalne globalne optimizacije primjenom načela grananja i ograđivanja.

Primjer 5.9. Zadan je interval $Q = [-3, 6]$ te funkcija $f: Q \rightarrow \mathbb{R}$ formulom $f(x) = \frac{1}{4}x + \sin x$ (Slika 5.15a). Primjenom načela grananja i ograđivanja treba pronaći točku u kojoj se postiže globalni minimum funkcije f na intervalu Q s točnošću $\varepsilon = 0.15$



Slika 5.15: Metoda grananja i ograđivanja za funkciju $f: [-3, 6] \rightarrow \mathbb{R}$, $f(x) = \frac{1}{4}x + \sin x$

Na početku metode lista \mathcal{L} sastoji se samo od intervala $Q = [-3, 6]$, odnosno $\mathcal{L} = \{[-3, 6]\}$. Odredimo donju $\Phi_{LB}([-3, 6])$ i gornju među $\Phi_{UB}([-3, 6])$. Kako je

$-1 \leq \sin x \leq 1$, vrijedi (vidi Slika 5.15b)

$$f(x) \geq \frac{1}{4}x - 1.$$

U cilju profinjnjenja donje međe, u rubnim točkama $x = -3$ te $x = 6$, konstruiramo tangente na graf funkcije f . U točki $x = -3$, tangenta glasi: $y = -3.11 - 0.74x$, dok u točki $x = 6$, tangenta glasi: $y = -6.04 + 1.21x$ (vidi Sliku 5.15c). Sada je očigledno (Slika 5.15d)

$$f(x) \geq \Phi_{LB}(x) := \max\{\frac{1}{4}x - 1, -3.11 - 0.74x, -6.04 + 1.21x\},$$

odakle je (Slika 5.15e)

$$\Phi_{LB}([-3, 6]) = \min_{x \in [-3, 6]} \Phi_{LB}(x) = -1.53,$$

a postiže se u točki $\bar{x} = -2.13$.

Nadalje određujemo gornju među. Jedna je mogućnost primijeniti neku lokalnu minimizacijsku metodu te za gornju među odabrati vrijednost funkcije kojoj je ta metoda konvergirala. Kako je funkcija f dva puta neprekidno derivabilna, u tu svrhu primjenjujemo *Newtonovu metodu*, pri čemu za početnu aproksimaciju uzimamo $x_0 = 6$. *Newtonova metoda* konvergirala je prema $\tilde{x} = 4.46$ te je prema tome gornja međa (Slika 5.15f)

$$\Phi_{UB}([-3, 6]) = f(4.46) = 0.147.$$

Razlika gornje i donje međe iznosi

$$\Phi_{UB}([-3, 6]) - \Phi_{LB}([-3, 6]) = 1.67 > \varepsilon = 0.15,$$

što znači da interval $[-3, 6]$ treba podijeliti (granati) u dva podintervala. Pretpostavimo da interval dijelimo na podintevala $Q_1 = [-3, 4.46]$, $Q_2 = [4.46, 6]$. Točku $\tilde{x} = 4.46$ zovemo *točka grananja*. Lista \mathcal{L} sada glasi

$$\mathcal{L} = \{[-3, 4.46], [4.46, 6]\}.$$

Odabiremo interval $[-3, 4.46]$ iz liste, obrišemo ga iz liste te za njega određujemo donju i gornju među. Za donju među koristimo istu funkciju kao i za interval $[-3, 6]$ pa je

$$\Phi_{LB}([-3, 4.46]) = \min_{x \in [-3, 4.46]} \Phi_{LB}(x) = -1.53,$$

a postiže se u točki $\bar{x} = -2.13$. U svrhu određivanja gornje međe na skupu $[-3, 4.46]$, primjenjujemo *Newtonovu metodu* s početnom aproksimacijom $x_0 = 3$. *Newtonova metoda* konvergirala je prema $\tilde{x} = -1.82$ te je prema tome gornja međa (Slika 5.15g)

$$\Phi_{UB}([-3, 4.46]) = f(-1.82) = -1.42.$$

Razlika gornje i donje međe iznosi

$$\Phi_{UB}([-3, 4.46]) - \Phi_{LB}([-3, 4.46]) = 0.11 < \varepsilon = 0.15.$$

Ovaj interval ne dijelimo dalje te za aproksimaciju točke u kojoj se postiže globalni minimum uzimamo $\tilde{x} = -1.82$, dok je odgovarajuća vrijednost globalnog minimuma $U := -1.42$.

Nadalje, promatramo interval $[4.46, 6]$. Za konstrukciju donje međe koristimo funkciju $\Phi_{LB}(x)$ te je

$$\Phi_{LB}([4.46, 6]) = f(4.46) = 0.115.$$

U svrhu određivanja gornje međe na skupu $[4.46, 6]$, primjenjujemo *Newtonovu metodu* s početnom aproksimacijom $x_0 = 6$. *Newtonova metoda* konvergirala je prema $\tilde{x} = 4.46$, te je pri tome gornja međa (Slika 5.15h)

$$\Phi_{UB}([4.46, 6]) = f(4.46) = 0.147.$$

Razlika gornje i donje međe iznosi

$$\Phi_{UB}([4.46, 6]) - \Phi_{LB}([4.46, 6]) = 0.04 < \varepsilon = 0.15.$$

Ovaj interval ne dijelimo dalje. Kako za trenutnu aproksimaciju globalnog minimuma $U = -1.42$ vrijedi $U < \Phi_{LB}([4.46, 6]) = 0.147$, ovaj interval ne sadrži točku globalnog minimuma.

Budući da lista \mathcal{L} ne sadrži više podskupova, zaključujemo da je funkcija f postigla globalni minimum funkcije u točki $\tilde{x} = -1.82$ ta da on iznosi $U = -1.42$.

Poglavlje 6

Mathematica programi i moduli

Tijekom izvođenja nastave predviđeno je ilustrirati metode i algoritme te njihova svojstva primjenom programskog sustava *Mathematica* [76] za koji Sveučilište u Osijeku redovito obnavlja licencu. S jedne strane, na taj način demonstriraju se osnove korištenja ovog programskog sustava, a s druge strane, metode i algoritmi navedeni u ovom udžbeniku postaju bliži i operativniji. Svi niže navedeni programi slobodno se mogu preuzeti na adresi:

<http://www.mathos.unios.hr/GOP-knjiga/GOPmoduli.rar>

Treba napomenuti kako opisani *Mathematica* moduli imaju za cilj prvenstveno ilustrirati obrađene metode minimizacije te nije namjera postići najefikasniju moguću implemetaciju metoda. Za zahtjevniju uporabu svakako postoji prostor za brojna poboljšanja i ubrzanja trenutnih implemetacija.

6.1 Uvod

Prije opisa konkretnih modula, navest ćemo neke njihove zajedničke karakteristike i sintaksu. Općeniti poziv modula ovdje ćemo navoditi u sljedećem obliku

```
ImeModula[arg1_, ... , argn_, oarg1_:v1, ... , oargm_:vm]
```

što znači da se modul *ImeModula* poziva s n argumenata *arg1*, ..., *argn*, te može imati m opcionalnih argumenata *oarg1*, ..., *oargm*. Opcionalni

argumenti mogu se zadati prilikom poziva modula, a ukoliko nisu zadani, za njihovu vrijednost uzima se vrijednost navedena nakon `:` u prethodnoj naredbi. Ova sintaksa odgovara načinu definiranja modula u *Mathematici* (vidjeti pripadnu temu u *Help-u*).

Moduli koje opisujemo u nastavku dijele sličnu strukturu ulaznih argumenata. Prvim argumentom modulu se predaje funkcija koju treba minimizirati. Zatim slijede parametri koji su specifični za pojedinu metodu. Argument `eps` standardno će označavati vrijednost koja se koristi pri provjeri kriterija zaustavljanja. Argument `It` u svim modulima označava maksimalan dozvoljen broj iteracija. Ukoliko se kriterij zaustavljanja ne ispuni unutar `It` iteracija, iterativni postupak se zaustavlja. Ostali standardni argumenti su indikator ispisa `Ind`, čija vrijednost različita od nula omogućava detaljan ispis iteracija algoritma, te indikator ispisa slika `S1`, čija vrijednost različita od nula omogućava odgovarajuće grafičke prikaze tamo gdje je to primjenjivo. U svim su modulima vrijednosti argumenata `Ind` i `S1` opcionalne, a postavljene su tako da se ispis iteracija, odnosno grafički prikazi prikazuju.

6.2 Klasične metode optimizacije

U ovoj točki ilustrirat ćemo uporabu *Mathematica*-modula koji implementiraju neke klasične metode optimizacije opisane u t.2, str.35. Karakteristika ovih metoda je eventualna garancija konvergencije iterativnog postupka prema lokalnom minimumu. Njihova je primjena stoga obično vezana uz pretpostavku konveksnosti minimizirajuće funkcije. Ovom pretpostavkom osigurano je da je pronađeni lokalni minimum ujedno i globalni (Lema 1.3, str.26).

6.2.1 Metode spusta za konveksne funkcije

Klasa metoda spusta opisana je u t.2.1, str.35. Pretpostavljamo da je $f: \mathbb{R}^n \rightarrow \mathbb{R}$ zadana dva puta neprekidno diferencijabilna i konveksna funkcija. Metode je moguće primijeniti i ako je funkcija zadana na nekom konveksnom skupu $\mathcal{D} \subset \mathbb{R}^n$. Iterativni postupak ovih metoda ima oblik $x^{k+1} = x^k + \alpha_k p^k$, gdje p^k predstavlja vektor smjera kretanja, a α_k duljinu koraka.

Koordinatna relaksacija

Metoda koordinatne relaksacije i izbor vektora smjera kretanja p^k opisan je u t.2.1.1, str.36. Izbor duljine koraka implementiran je sukladno Algo-

ritmu 2, str.39. Počevši s $\alpha = 1$, u svakoj se iteraciji duljina koraka smanjuje za zadani faktor $\nu \in \langle 0, 1 \rangle$ sve dok se ne ispuni uvjet $f(x^k + \alpha p^k) - f(x^k) \leq \tau \alpha \left(\nabla f(x^k) \right)^T p^k$ za zadanu vrijednost parametra $\tau \in \langle 0, 0.5 \rangle$. Prema Lemi 2.1, takav α uvijek je moguće pronaći, no iz praktičnih razloga proces smanjivanja parametra α zaustavit će se kada njegova vrijednost padne ispod vrijednosti α_{min} koju je moguće zadati kao ulazni argument modula. Modul je definiran naredbom

```
KoordinatnaRelaksacija[f_, x0_, It_, eps_,
    alfamin_:10^(-8), nu_:0.5, tau_:0.25, Ind_:1, Sl_:1]
```

gdje je x_0 početna aproksimacija, It maksimalan broj iteracija, dok $alfamin$, nu i tau označavaju redom parametre α_{min} , ν i τ kod izbora duljine koraka. Ulazna funkcija f može imati proizvoljan broj varijabli, no mora biti definirana tako da je moguće simbolički izračunati njezin gradijent. Kriterij zaustavljanja definiran je uvjetom $\|\nabla f(x^k)\| \leq \epsilon$, gdje se $\epsilon > 0$ predaje argumentom eps . Ostali argumenti definirani su kao što je objašnjeno u t.6.1, str.121. Grafički prikaz dostupan je za funkcije dvije varijable.

Gradijentna metoda

Gradijentna metoda razlikuje se od prethodno opisane *Metode koordinatne relaksacije* samo po načinu izbora vektora smjera kretanja. Odgovarajući modul definiran je s

```
Gradijentna[f_, x0_, It_, eps_,
    alfamin_:10^(-8), nu_:0.5, tau_:0.25, Ind_:1, Sl_:1]
```

gdje ulazni argumenti imaju isto značenje kao kod *Metode koordinatne relaksacije*.

Newtonova metoda minimizacije

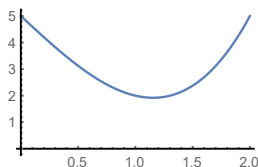
Algoritmom 1, str.38 definirana je *Newtonova metoda* bez izbora duljine koraka, dok je Algoritmom 2, str.39 definiran izbor duljine koraka. Odgovarajući moduli definirani su s

```
Newton[f_, x0_, It_, eps_, Ind_:1, Sl_:1]
```

```
NewtonKorak[f_, x0_, It_, eps_,
    lammin_:10^(-8), nu_:0.5, tau_:0.25, Ind_:1, Sl_:1]
```

Oznake argumenata i kriterij zaustavljanja jednaki su kao kod prethodnih dviju metoda spusta. Kod *Newtonove metode* dodatna je pretpostavka da je zadana funkcija definirana tako da je moguće simbolički izračunati njezin Hessian.

Primjer 6.1. *Treba odrediti aproksimaciju točke minimuma konveksne funkcije $f: [0, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 4x + 5$ (Slika 6.1).*



Slika 6.1: Graf funkcije $f: [0, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 4x + 5$

Osim gotovog modula `NMinimize` za numeričku minimizaciju, programski sustav *Mathematica* sadrži i funkciju `Minimize` koja na jednostavnijim funkcijama simbolički traži minimum. Tako za zadanu funkciju f sljedećom naredbom možemo egzaktno riješiti problem minimizacije.

```
In[1]:= f[x_]:= x^3-4x+5
In[2]:= Minimize[{f[x], 0 <= x <= 2}, x]
Out[2]:= {5 - 16/(3 Sqrt[3]), {x -> 2/Sqrt[3]}}
```

Korištenjem ugrađenog modula `NMinimize` dobivamo sljedeću aproksimaciju rješenja

```
In[3]:= NMinimize[{f[x], 0 <= x <= 2}, x]
Out[3]:= {1.9208, {x -> 1.1547}}
```

Na funkciju f iz Primjera 6.1 mogu se primijeniti sve prethodno navedene metode spusta. Sljedećim pozivom mogu se dobiti detaljni ispisi iteracija za *Metodu koordinatne relaksacije*, *Gradijentnu metodu*, *Newtonovu metodu* i *Newtonovu metodu s izborom duljine koraka*. Za opcionalne argumente koriste se zadane vrijednosti dok je za početnu aproksimaciju odabrana vrijednost 0.5. Izlazna vrijednost modula je lista sastavljena od aproksimacije točke minimuma i vrijednosti funkcije u toj točki.

```
In[4]:= KoordinatnaRelaksacija[f, {0.5}, 20, 0.00005]
Out[4]:= {1.9208, {x -> {1.15469}}}
```

```
In[5]:= Gradijentna[f, {0.5}, 20, 0.00005]
Out[5]:= {1.9208, {x -> {1.1547}}}
```



```
In[6]:= Newton[f, {0.5}, 20, 0.00005]
Out[6]:= {1.9208, {x -> {1.1547}}}
```

```
In[7]:= NewtonKorak[f, {0.5}, 20, 0.00005]
Out[7]:= {1.9208, {x -> {1.1547}}}
```

Može se uočiti da svi iterativni postupci konvergiraju prema točki minimuma.

Primjer 6.2. *Zadana je funkcija dviju varijabli, $f: \mathbb{R}^2 \rightarrow \mathbb{R}$, $f(x, y) = x^4 + y^4 + y^2$, za koju treba odrediti aproksimaciju točke globalnog minimuma.*

Lako je provjeriti da je f konveksna funkcija i da postiže jedinstveni globalni minimum u točki $(0, 0)$. Aproksimaciju točke minimuma pomoću *Metode koordinatne relaksacije*, *Gradijentne metode* odnosno *Newtonove metode* možemo odrediti pozivom sljedećih modula.

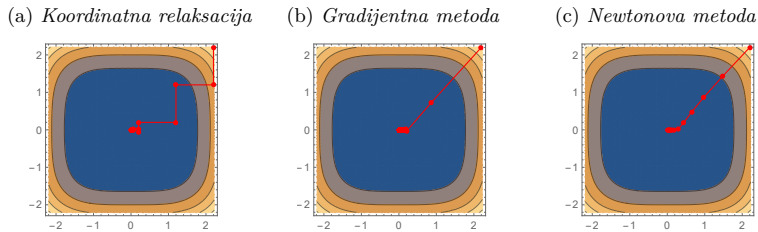
```
In[1]:= f[x_, y_] := x^4 + y^4 + y^2
```

```
In[2]:= KoordinatnaRelaksacija[f, {2.2, 2.2}, 1000, 0.00005]
Out[2]:= {2.45631*10^(-8), {x -> {0.0125, -0.000012207}}}
```

```
In[3]:= Gradijentna[f, {2.2, 2.2}, 1000, 0.00005]
Out[3]:= {2.09199*10^(-7), {x -> {0.0213865, -7.9294*10^(-14)}}}
```

```
In[4]:= Newton[f, {2.2, 2.2}, 1000, 0.00005]
Out[4]:= {8.26628*10^(-8), {x -> {0.0169562, 0.}}}
```

Kao početna aproksimacija izabrana je točka $(2.2, 2.2)$. Kriterij zaustavljanja u *Metodi koordinatne relaksacije* ispunjen je nakon 13 iteracija, kod *Gradijentne metode* tek nakon 270 iteracija, dok je *Newtonovoj metodi* bilo potrebno samo 12 iteracija. U konkretnom primjeru izbor duljine koraka kod *Newtonove metode* nije imao nikakav utjecaj jer je izlazna vrijednost algoritma izbora duljine koraka u svim iteracijama bila jednaka 1. Na Slici 6.2 grafički je prikazan niz aproksimacija dobiven promatranim metodama. Takav prikaz dobije se kao rezultat poziva modula ukoliko je argument `S1` postavljen na vrijednost različitu od nula.

Slika 6.2: Minimizacija funkcije $f: \mathbb{R}^2 \rightarrow \mathbb{R}$, $f(x, y) = x^4 + y^4 + y^2$

6.2.2 Nelder-Meadova metoda

Modul `NelderMead` implementira *Nelder-Meadovu metodu* opisanu u t.2.2, str.41 za funkcije dviju varijabli na osnovu Algoritma 3, str.45. Modul je prvenstveno namijenjen ilustraciji *Nelder-Meadove metode*, dok je puna implementacija metode za funkcije proizvoljnog broja varijabli dostupna ugrađenim modulom `NMinimize` uz opciju `Method->"NelderMead"`. Definicija modula je sljedeća:

```
NelderMead[f_, T1_, T2_, T3_, It_, Ind_:1, S1_:1, a_:0, b_:1, c_:0, d_:1]
```

Argumentom `f` modulu se predaje funkcija $f: \mathbb{R}^2 \rightarrow \mathbb{R}$ koju treba minimizirati, `T1`, `T2` i `T3` vrhovi su polaznog trokuta, a `It` broj iteracija. Argument `Ind` omogućuje detaljan ispis iteracija, a `S1` grafički prikaz, pri čemu argumenti `a`, `b`, `c` i `d` određuju područje crtanja kontura funkcije f .

Primjer 6.3. *Promatramo Branin funkciju (koja se također razmatra kao test primjer u [26]) $f: [-5, 10] \times [0, 15] \rightarrow \mathbb{R}$,*

$$f(x_1, x_2) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10,$$

koja globalni minimum postiže u trima različitim točkama: $(-\pi, 12.275)$, $(\pi, 2.275)$, $(3\pi, 2.475)$ u kojima vrijednost funkcije iznosi 0.398.

Pozivom modula

```
In[1]:= f[x_, y_]:= (y-5.1/(4*Pi^2)*x^2+5/Pi*x-6)^2+10*(1-1/(8*Pi))*Cos[x]+10
a=-5; b=10; c=0; d=15;
NelderMead[f, {0,10}, {-1,12}, {1,14}, 8, 1, 1, a, b, c, d]
NelderMead[f, {0,2}, {4,8}, {5,2}, 8, 1, 1, a, b, c, d]
NelderMead[f, {9,2}, {7,6}, {5,3}, 8, 1, 1, a, b, c, d]
```

dobivamo ispis prvih osam iteracija *Nelder-Meadove metode* uz početni trokut $\triangle T_1 T_2 T_3$ određen vrhovima:

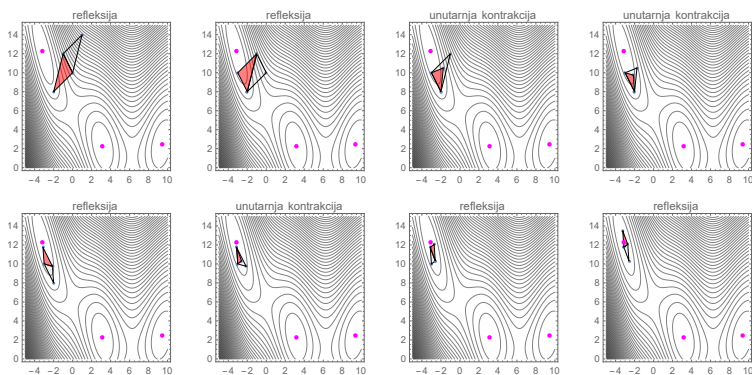
- $T_1 = (0, 10)$, $T_2 = (-1, 12)$, $T_3 = (1, 14)$,
- $T_1 = (0, 2)$, $T_2 = (4, 8)$, $T_3 = (5, 2)$,
- $T_1 = (9, 2)$, $T_2 = (7, 6)$, $T_3 = (5, 3)$.

Dobivene aproksimacije točke minimuma u ispisu računaju se kao težište trenutnog trokuta. Grafički prikaz tijeka algoritma uz odgovarajuće transformacije trokuta prikazan je na Slikama 6.3, 6.4 i 6.5. Pri tome crveni trokut označava transformaciju dobivenu u toj iteraciji dok su globalni minimumi označeni točkama. Početni trokuti odabrani su tako da postupak konvergira odgovarajućoj točki globalnog minimuma. To možemo vidjeti pozovemo li module bez detaljnog ispisa uz veći broj iteracija:

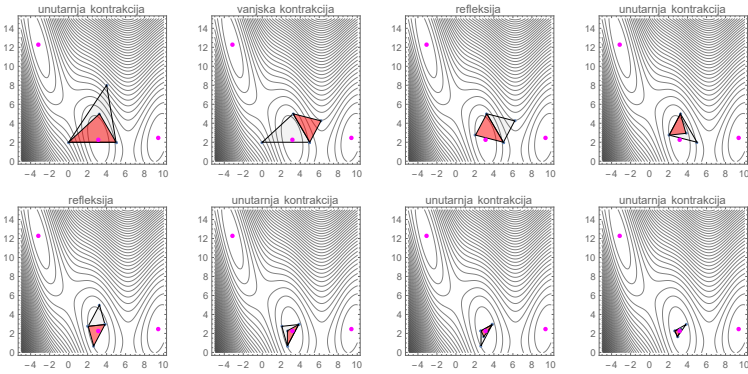
```
In[2]:= NelderMead[f, {0, 10}, {-1, 12}, {1, 14}, 50, 0, 0]
Out[2]:= {0.397887, {x -> {-3.14159, 12.275}}}
```

```
In[3]:= NelderMead[f, {0, 2}, {4, 8}, {5, 2}, 50, 0, 0]
Out[3]:= {0.397887, {x -> {3.14159, 2.275}}}
```

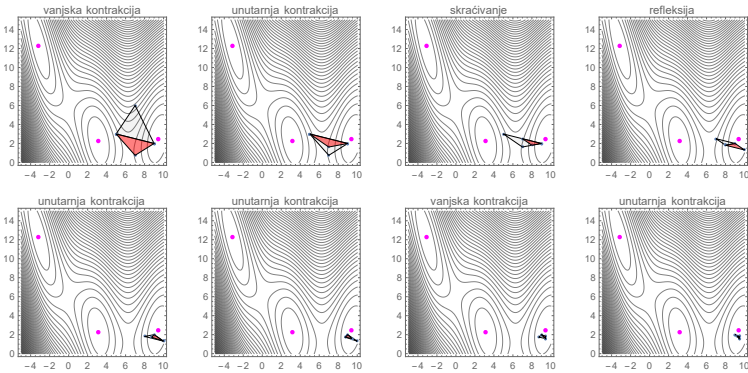
```
In[4]:= NelderMead[f, {9, 2}, {7, 6}, {5, 3}, 50, 0, 0]
Out[4]:= {0.397887, {x -> {9.42478, 2.475}}}
```



Slika 6.3: *Nelder-Meadova metoda* uz početni trokut $\triangle T_1 T_2 T_3$: $T_1 = (0, 10)$, $T_2 = (-1, 12)$, $T_3 = (1, 14)$



Slika 6.4: *Nelder-Meadova metoda* uz početni trokut $\triangle T_1 T_2 T_3$: $T_1 = (0, 2)$, $T_2 = (4, 8)$, $T_3 = (5, 2)$



Slika 6.5: *Nelder-Meadova metoda* uz početni trokut $\triangle T_1 T_2 T_3$: $T_1 = (9, 2)$, $T_2 = (7, 6)$, $T_3 = (5, 3)$

6.2.3 Kvadratne interpolacijske metode za jednodimenzionalnu minimizaciju

Klasa kvadratnih interpolacijskih metoda (t.2.3, str.41), odnosno *Metoda dvije točke I*, *Metoda dvije točke II* i *Metoda tri točke* implementirane su kao moduli

```
Inter2tockeI[f_, x0_, x1_, It_, eps_, Ind_:1, Sl_:1]
```

```
Inter2tockeII[f_, x0_, x1_, It_, eps_, Ind_:1, S1_:1]
```

```
Inter3tocke[f_, x0_, x1_, x2_, It_, eps_, Ind_:1, S1_:1]
```

Za *Metode dvije točke* ulazna funkcija $f: \mathbb{R} \rightarrow \mathbb{R}$ treba biti derivabilna i definirana tako da je derivaciju moguće simbolički izračunati. Argumenti x_0 , x_1 , x_2 označavaju početne aproksimacije, It je maksimalan broj iteracija, eps se koristi u provjeri kriterija zaustavljanja $|f'(x_k)| \leq eps$, dok Ind i $S1$ standardno označavaju indikatore ispisa i slika. Graf funkcije bit će nacrtan od najmanje do najveće početne točke.

Primjer 6.4. *Uporabu modula ilustrirat ćemo na funkciji $f: [0, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 4x + 5$ iz Primjera 6.1.*

Za početne točke u *Metodama dvije točke* uzet ćemo rubove domene $x_0 = 0$ i $x_1 = 2$, dok će dodatna točka za *Metodu tri točke* biti $x_2 = 0.5$. Module pozivamo sljedećim naredbama:

```
In[1]:= f[x_]:= x^3 - 4*x + 5
```

```
In[2]:= Inter2tockeI[f, 0, 2, 20, 0.0005]
```

```
Out[2]:= {1.9208, {x -> 1.15464}}
```

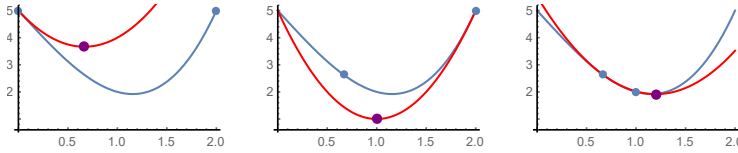
```
In[3]:= Inter2tockeII[f, 0, 2, 20, 0.0005]
```

```
Out[3]:= {1.9208, {x -> 1.1547}}
```

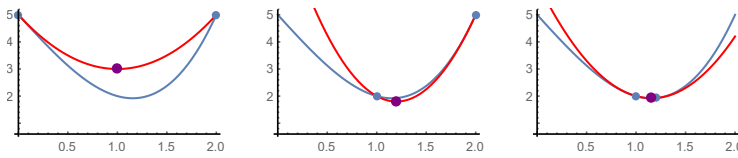
```
In[4]:= Inter3tocke[f, 0, 0.5, 2, 20, 0.0005]
```

```
Out[4]:= {1.9208, {x -> 1.15464}}
```

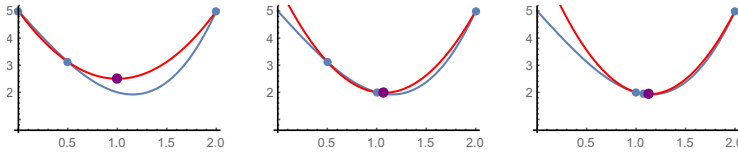
Sve tri metode postigle su kriterij zaustavljanja za manje od zadanog broja iteracija i to *Metode dvije točke* nakon 5 iteracija, a *Metoda tri točke* nakon 9 iteracija. Prateći grafički prikaz za prve tri iteracije prikazan je na Slikama 6.6, 6.7 i 6.8. Plavo obojani graf označava funkciju, dok crveni graf označava parabolu kojom se aproksimira funkcija na osnovu dviju odnosno triju točaka. Točke su označene plavom bojom, a veća ljubičasta točka označava minimum dobivene parabole. Kod *Metode dvije točke*, točka minimuma parabole ulazi u sljedeću iteraciju umjesto točke koja je prva dodana. Kod *Metode tri točke*, točka minimuma parabole zamijenit će jednu od tri točke na osnovu kriterija rasta i pada funkcije koja se minimizira.



Slika 6.6: Prve tri iteracije *Metode dvije točke I* za funkciju $f: [0, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 4x + 5$



Slika 6.7: Prve tri iteracije *Metode dvije točke II* za funkciju $f: [0, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 4x + 5$



Slika 6.8: Prve tri iteracije *Metode tri točke* za funkciju $f: [0, 2] \rightarrow \mathbb{R}$, $f(x) = x^3 - 4x + 5$

6.3 Jednodimenzionalna minimizacija kvazikonveksnih funkcija

U ovom poglavlju dajemo upute o korištenju modula za minimizaciju jednodimenzionalnih konveksnih ili kvazikonveksnih funkcija definiranih na intervalu $[a, b] \subset \mathbb{R}$. Moduli su izrađeni na osnovi algoritama za minimizaciju jednodimenzionalnih kvazikonveksnih funkcija (t.3, str.53). Navest ćemo upute za korištenje *Mathematica*-modula za traženje globalnog minimuma za konveksnu i kvazikonveksnu derivabilnu funkciju $f: [a, b] \rightarrow \mathbb{R}$:

- *Metoda tangenti* za konveksnu derivabilnu funkciju;
- *Metoda polovljenja* za kvazikonveksnu derivabilnu funkciju;

kao i *Mathematica*-modula za traženje globalnog minimuma kvazikonveksne nederivabilne funkcije $f: [a, b] \rightarrow \mathbb{R}$:

- *Metoda polovljenja* za kvazikonveksnu nederivabilnu funkciju;
- *Metoda zlatnog reza*.

Moduli će biti ilustrirani odgovarajućim primjerima i zadacima.

6.3.1 *Mathematica*-moduli za traženje globalnog minimuma konveksne i kvazikonveksne derivabilne funkcije

U ovom poglavlju pokazat ćemo korištenje *Mathematica*-modula za traženje globalnog minimuma konveksne, odnosno kvazikonveksne derivabilne funkcije $f: [a, b] \rightarrow \mathbb{R}$, kojima su operacionalizirane sljedeće metode:

- *Metoda tangenti* za konveksnu derivabilnu funkciju;
- *Metoda polovljenja* za kvazikonveksnu derivabilnu funkciju.

Metoda tangenti za konveksnu derivabilnu funkciju

Za zadanu konveksnu derivabilnu funkciju $f: [a, b] \rightarrow \mathbb{R}$ modul `MeTang` operacionalizira Algoritam 6, str.55 i daje aproksimaciju globalnog minimuma $(\hat{u}, f(\hat{u}))$ i apsolutnu vrijednost derivacije funkcije f' u točki \hat{u} kao jednu indikaciju točnosti aproksimacije. Modul je definiran s

```
MeTang[f_, a_, b_, u0_, n_, Ind_:1]
```

gdje je `f` minimizirajuća funkcija, `a` lijevi rub intervala, `b` desni rub intervala, `u0` $\in [a, b]$ početna aproksimacija, a `n` broj iteracija.

U modulu `MeTang` najprije definiramo tangentu Λ u točki u_0 . Sljedeću aproksimaciju u_1 određujemo kao točku u kojoj tangenta Λ postiže najmanju vrijednost na intervalu $[a, b]$ (lijevi ili desni rub intervala $[a, b]$). Nakon toga definiramo odgovarajući slomljeni pravac P_1 i sljedeću aproksimaciju kao točku minimuma funkcije P_1 . Postupak se iterativno nastavlja do n -te aproksimacije. Poziv i output modula koji ne ispisuje međurezultate zadan je s:

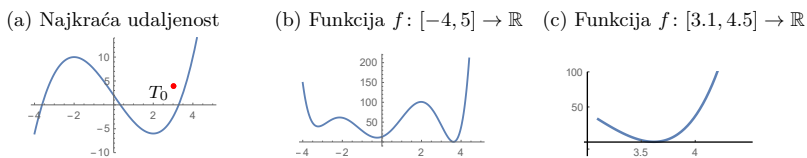
```
In[1]:= MeTang[f, a, b, u0, n, 0]
```

```
Out[2]:= {up, f[up], Abs[f'[up]]}
```

gdje je up n -ta aproksimacija točke minimuma.

Izvršenje programa prati se primjenom naredbe `Monitor`.

Primjer 6.5. Promatramo problem određivanja najkraće LS-udaljenosti točke $T_0 = (3, 4)$ do kubne parabole zadane funkcijom $q: [-4, 5] \rightarrow \mathbb{R}$, $q(x) = \frac{1}{2}x^3 - 6x + 2$ (vidi Sliku 6.9a i također Primjer 1.2, str.7 i Primjer 3.1, str.54).



Slika 6.9: Problem najkraće LS-udaljenosti točke do kubne parabole

LS-udaljenost točke $T_0 = (3, 4)$ do kubne parabole q zadana je funkcijom $f: [-4, 5] \rightarrow \mathbb{R}_+$, $f(x) = (x - 3)^2 + (q(x) - 4)^2$ (vidi Sliku 6.9b). Ova funkcija nije konveksna na intervalu $[-4, 5]$, pa zato promatramo njenu restrikciju na interval $[3.1, 4.5]$, gdje postiže i svoj globalni minimum (vidi Sliku 6.9c). Pokažite da je funkcija f konveksna na $[3.1, 4.5]$.

Za traženje globalnog minimuma funkcije f na intervalu $[3.1, 4.5]$ najprije ćemo primijeniti gotovi *Mathematica*-modul `NMinimize`.

```
In[3]:= q[x_] := x^3/2 - 6 x + 2; {x0, y0} = {3, 4}; a = -4; b = 5;
        f[x_] := (x - x0)^2 + (q[x] - y0)^2;
        NMinimize[{f[x], a <= x <= b}, x]
```

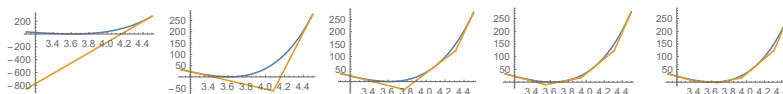
```
Out[4]= {10.8186, {x -> -0.243094}}
```

Uz početnu aproksimaciju $u_0 = 4.5$, nakon prvih $n = 5$ iteracija i opciju kontrole vremena izvršenja (u sec.) modul `MeTang` daje

```
In[5]:= MeTang[f, 3.1, 4.5, 4.5, 5, 0] // Timing
```

```
Out[6]= {0.562500, {3.66835, 0.898341, 20.4029}}
```

Pokretanjem modula `MeTang` uz `Ind=1` ispisuju se sve iteracije i prikazuje odgovarajuća animacija iterativnog postupka.



Niže navodimo rezultate izvođenja modula **MeTang** uz **Ind=0** i kontrolu vremena izvršenja za $n = 10$ i $n = 20$.

```
Out[7]:= {3.171875, {3.61388, 0.383975, 1.0669}}
Out[8]:= {14.937500, {3.61676, 0.382438, 0.00010046}}
```

Zadatak 6.1. Izradite *Mathematica*-modul koji će *Metodom tangenti* pronalaziti globalni minimum konveksne derivabilne funkcije $f: [a, b] \rightarrow \mathbb{R}$, a iterativni postupak zaustavite nakon ispunjavanja uvjeta $|u_n - u_{n-1}| < \epsilon$ & $|f'(u_n)| < \epsilon$, gdje je u_n n -ta aproksimacija točke minimuma.

Zadatak 6.2. Izradite *Mathematica*-modul za *Metodu tangenti* tako da traženje minimuma funkcije P_n implementirate pretraživanjem vrhova slomljenog pravca P_n . Usporedite vrijeme izvršavanja s modulom **MeTang**.

Metoda polovljenja za kvazikonveksnu derivabilnu funkciju

Za zadanu kvazikonveksnu derivabilnu funkciju $f: [a, b] \rightarrow \mathbb{R}$ modul **MePolDer** operacionalizira Algoritam 7, str.58 i daje aproksimaciju globalnog minimuma $(\hat{u}, f(\hat{u}))$ s točnošću $\epsilon > 0$. Modul je definiran s:

```
MePolDer[f_, a_, b_, It_, eps_, Ind_:1]
```

gdje je **f** minimizirajuća funkcija, **a** lijevi rub intervala, **b** desni rub intervala, a **eps**>0 tražena točnost. Poziv i output modula koji ne ispisuje međurezultate zadan je s

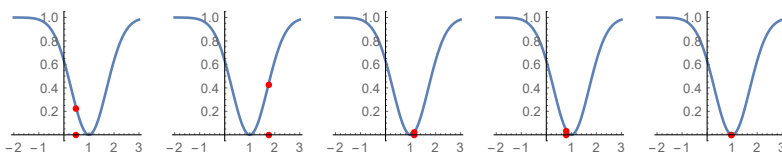
```
In[1]:= MePolDer[f, a, b, 50, eps, 0]
```

```
Out[2]:= {f[xp], {x -> xp}}
```

gdje je **xp** dobivena aproksimacija točke minimuma.

Primjer 6.6. *Potražimo globalni minimum kvazikonveksne derivabilne funkcije $f: [-2, 3] \rightarrow \mathbb{R}$, $f(x) = 1 - e^{-(x-1)^2}$.*

Uz traženu točnost $\epsilon = 0.05$ (jedna točna decimala), nakon $k = 6$ iteracija modul **MePolDer**[f, -2, 3., 50, .05, 0] daje {0.0000610333, {x -> 1.00781}}. Uz opciju **Ind=1** dobivamo ispis svih iteracija i odgovarajuću grafičku animaciju.



6.3.2 *Mathematica*-moduli za traženje globalnog minimuma kvazikonveksne nederivabilne funkcije

U ovom poglavlju pokazat ćemo korištenje *Mathematica*-modula kojima su operacionalizirane sljedeće metode za traženje globalnog minimuma kvazikonveksne nederivabilne funkcije $f: [a, b] \rightarrow \mathbb{R}$:

- *Metoda polovljenja* za kvazikonveksnu nederivabilnu funkciju;
- *Metoda zlatnog reza*.

Metoda polovljenja za kvazikonveksnu nederivabilnu funkciju

Za kvazikonveksnu funkciju $f: [a, b] \rightarrow \mathbb{R}$, koja ne mora biti derivabilna, modul `MePolNe` operacionalizira Algoritam 8, str.62 i daje aproksimaciju globalnog minimuma $(\hat{u}, f(\hat{u}))$ s točnošću $\epsilon > 0$. Modul je definiran s:

```
MePolNe[f_, a_, b_, It_, eps_, del_, Ind_:1]
```

gdje je `f` minimizirajuća funkcija, `a` lijevi rub intervala, `b` desni rub intervala, `eps`>0 tražena točnost, a `del`>0 maleni broj. Poziv i output modula koji ne ispisuje međurezultate zadan je s:

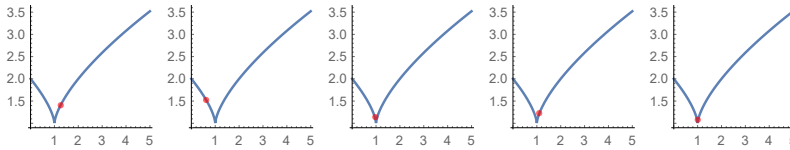
```
In[1]:= MePolNe[f, a, b, It, eps, del, 0]
```

```
Out[2]:={f[xp], {x -> xp}}
```

gdje je `xp` aproksimacija točke globalnog minimuma.

Primjer 6.7. *Potražimo globalni minimum kvazikonveksne nederivabilne funkcije $f: [0, 5] \rightarrow \mathbb{R}$, $f(x) = 1 + \sqrt[3]{(x-1)^2}$.*

Uz traženu točnost $\epsilon = 0.005$ (dvije točne decimale) i $\delta = 0.0025$, nakon $k = 10$ iteracija modul `MePolNe[f, 0, 5, 50, .005, .0025, 0]` daje rješenje `{1.00635, {x -> 1.00051}}`. Uz opciju `Ind=1` dobivamo ispis svih iteracija i odgovarajuću grafičku animaciju.



Metoda zlatnog reza

Za kvazikonveksnu funkciju $f: [a, b] \rightarrow \mathbb{R}$, koja ne mora biti derivabilna, modul `MeZlatni` operacionalizira Algoritam 9, str.67 i daje aproksimaciju globalnog minimuma $(\hat{u}, f(\hat{u}))$ s točnošću $\epsilon > 0$. Modul je definiran s:

```
MeZlatni[f_, a_, b_, It_, eps_, Ind_:1]
```

gdje je `f` minimizirajuća funkcija, `a` lijevi rub intervala, `b` desni rub intervala, a `eps>0` tražena točnost. Poziv i output modula koji ne ispisuje međurezultate zadan je s:

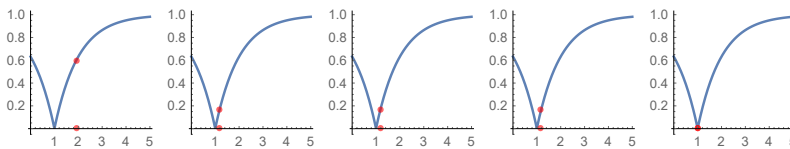
```
In[1]:= MeZlatni[f, a, b, 50, eps, 0]
```

```
Out[2]:= {f[xp], {x -> xp}}
```

gdje je `xp` aproksimacija točke globalnog minimuma.

Primjer 6.8. *Potražimo globalni minimum kvazikonveksne nederivabilne funkcije $f: [0, 5] \rightarrow \mathbb{R}$, $f(x) = 1 - e^{-|x-1|}$.*

Uz traženu točnost $\epsilon = 0.005$ (dvije točne decimale) nakon `k=14` iteracija modul `MeZlatni[f, 0, 5, 50, .005, 0]` daje `{0.0000661048, {x -> 0.999934}}`. Uz opciju `Ind=1` dobivamo ispis svih iteracija i odgovarajuću grafičku animaciju.



6.3.3 Numerički eksperimenti

U ovom poglavlju razmatramo problem globalne optimizacije za jednu klasu funkcija koje se često pojavljuju u različitim primijenjenim istraživanjima [29, 54, 55, 61, 62], a već smo ga ranije spomenuli u Primjeru 1.20, str.23.

Budući da praktični primjeri mogu uključivati izuzetno veliki broj podataka, ovakvi problemi pokazuju se pogodnima kod numeričkog eksperimentiranja i testiranja različitih numeričkih metoda.

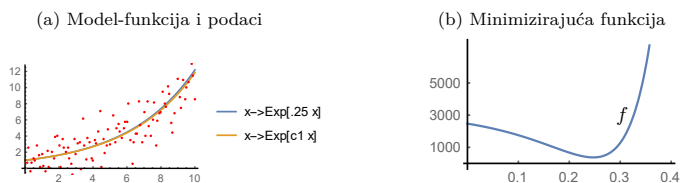
Primjer 6.9. Zadana je model-funkcija $g: [0, 10] \rightarrow \mathbb{R}$, $g(x; c) = e^{cx}$ s jednim parametrom $c \in \mathbb{R}$ i podaci (vidi Sliku 6.10a) (t_i, y_i) , $i = 1, \dots, m$

$$t_i = i \frac{10}{m}, \quad i = 1, \dots, m,$$

$$y_i = g(t_i; .25) + \varepsilon_i, \quad i = 1, \dots, m,$$

gdje je ε_i pseudoslučajan broj generiran iz normalne distribucije s očekivanjem 0 i varijancom 2. LS-optimalnu vrijednost \hat{c} parametra c odredit ćemo minimizacijom funkcije (vidi Primjer 1.20, str.23)

$$f(c) = \sum_{i=1}^m (y_i - g(t_i; c))^2.$$



Slika 6.10: Traženje LS-optimalnog parametra model-funkcije $f(x; c) = e^{cx}$

Graf funkcije f prikazan je na Slici 6.10b. Primjenom gotovog *Mathematica*-modula `NMinimize` dobivamo

```
In[1]:= NMinimize[{f[c], c]
```

```
Out[2]= {363.888, {c -> 0.251902}}
```

Sve prethodno navedene metode za jednodimenzionalnu minimizaciju također se uspješno i efikasno mogu primijeniti za traženje globalnog minimuma ove funkcije. Jedino kod *Metode tangenti* područje minimizacije treba suziti tako da na njemu funkcija f bude konveksna.

Zadatak 6.3. Definirajte $m = 1000$ podataka (t_i, y_i) , $i = 1, \dots, m$ slično kao u prethodnom Primjeru 6.9 te za jednoparametarsku model-funkciju $g: [0, 10] \rightarrow \mathbb{R}$, $g(x; c) = e^{cx}$ odredite najbolji LS i najbolji LAD parametar \hat{c} . Za traženje rješenja koristite ranije spomenute metode. Usporedite njihovu efikasnost mjerenjem CPU vremena izvršenja iterativnog postupka i potrebnim brojem izračunavanja minimizirajuće funkcije. Za analizu rezultata iskoristite grafičke mogućnosti programskog sustava *Mathematica*.

Primjer 6.10. Zadana je model-funkcija $g: [0, 10] \rightarrow \mathbb{R}$, $g(x; b, c) = be^{cx}$ s dva parametra $b, c \in \mathbb{R}$ i podaci (vidi Sliku 6.11a) (t_i, y_i) , $i = 1, \dots, m$

$$t_i = i \frac{10}{m}, \quad i = 1, \dots, m,$$

$$y_i = g(t_i; 1, .25) + \varepsilon_i, \quad i = 1, \dots, m,$$

gdje je ε_i pseudoslučajan broj generiran iz normalne distribucije s očekivanjem 0 i varijancom 2. LS-optimalne vrijednosti \hat{b}, \hat{c} parametara b, c odredit ćemo minimizacijom funkcije (vidi Primjer 1.20, str.23)

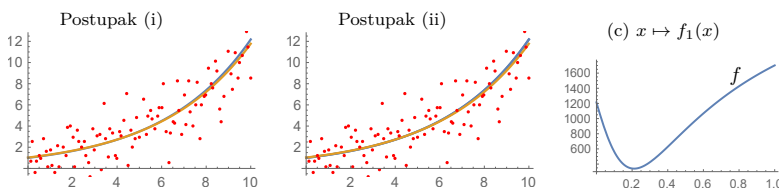
$$f(b, c) = \sum_{i=1}^m (y_i - be^{ct_i})^2. \quad (6.1)$$

- Postupak (i).
LS-optimalne vrijednosti \hat{b}, \hat{c} parametara b, c odredit ćemo primjenom *Mathematica*-modula `NMinimize[]` na funkciju f

```
In[1]:= NMinimize[{f[b,c], {b,c]}
```

```
Out[2]= {373.419, {b -> 1.02741, c -> 0.243914}}
```

Grafovi originalne funkcije g (plavo) i procijenjene \hat{g} (narančasto) prikazani su na Slici 6.11a.



Slika 6.11: Traženje LS-optimalnih parametara model-funkcije $f(x; b, c) = be^{cx}$

- Postupak (ii).
Budući da gradijent funkcije f u točki minimuma mora iščeznuti, iz $\frac{\partial f}{\partial b} = 0$ dobivamo

$$\sum_{k=1}^m y_k e^{ct_k} = b \sum_{k=1}^m e^{2ct_k}. \quad (6.2)$$

Iz ove jednadžbe izrazimo b i uvrstimo u (6.1). Tako dobivamo funkciju jedne varijable (Slika 6.11c)

$$f_1(c) = \sum_{i=1}^m \left(y_i - y_i - \frac{\sum y_k e^{ct_k}}{\sum e^{2ct_k}} e^{ct_i} \right)^2. \quad (6.3)$$

Na funkciju f_1 primijenimo neku od metoda za jednodimenzionalnu minimizaciju i dobivamo \hat{c} . Uvrštavanjem \hat{c} u (6.2) dobivamo \hat{b} .

Poznavajući procijenjene vrijednosti parametara $(\hat{b}, \hat{c}) = (1.0274, 0.2439)$ dobivamo model-funkciju $g(x) = 1.02741e^{0.2439x}$ (vidi Sliku 6.11b).

Zadatak 6.4. Definirajte $m = 1000$ podataka (t_i, y_i) , $i = 1, \dots, m$ slično kao u Primjeru 6.9 te za dvoparametarsku model-funkciju $g: [0, 10] \rightarrow \mathbb{R}$, $g(x; b, c) = be^{cx}$ odredite najbolje LS parametre \hat{b}, \hat{c} . Za traženje rješenja primijenite postupak analogan Postupku (ii) iz Primjera 6.10 i koristite ranije spomenute metode jednodimenzionalne minimizacije. Usporedite efikasnost spomenutih metoda mjerenjem CPU vremena izvršenja iterativnog postupka i potrebnim brojem izračunavanja vrijednosti minimizirajuće funkcije. Za analizu rezultata iskoristite grafičke mogućnosti programskog sustava *Mathematica*.

Zadatak 6.5. Definirajte $m = 1000$ podataka (t_i, y_i) , $i = 1, \dots, m$ slično kao u Primjeru 6.9 te za troparametarsku model-funkciju $g: [0, 10] \rightarrow \mathbb{R}$, $g(x; b, c) = a + be^{cx}$ odredite najbolje LS parametre $\hat{a}, \hat{b}, \hat{c}$. Za traženje rješenja primijenite postupak analogan Postupku (ii) iz Primjera 6.10 i koristite ranije spomenute metode jednodimenzionalne minimizacije. Usporedite efikasnost spomenutih metoda mjerenjem CPU vremena izvršenja iterativnog postupka i potrebnim brojem izračunavanja vrijednosti minimizirajuće funkcije. Za analizu rezultata iskoristite grafičke mogućnosti programskog sustava *Mathematica*.

6.4 Jednodimenzionalna globalna optimizacija

Metode globalne optimizacije za Lipschitz-neprekidnu funkciju $f: [a, b] \rightarrow \mathbb{R}$ opisane u t.4, str.69 pronalaze aproksimacije svih točaka globalnog minimuma funkcije f . U nastavku su opisani moduli koji implementiraju ove metode.

6.4.1 Pijavskijeva metoda slomljenih pravaca

Metoda Pijavskog (t.4.2, str.74) implementirana je modulom

```
Pijavskij[f_, a_, b_, L_, u0_, It_, Ind_:1, Sl_:1]
```

gdje je f funkcija koju treba minimizirati, a i b su rubovi intervala, L vrijednost Lipschitzove konstante, u_0 početna točka, It broj iteracija, a Ind i $S1$ standardno označavaju indikatore ispisa i slika. Modul `Pijavskij` implementiran je na osnovu Algoritma 10, str.76. Pri tome, traženje minimuma po dijelovima linearne funkcije P_n (korak 6. u Algoritmu 10) implementirano je pretraživanjem lokalnih minimuma funkcije P_n kako je opisano u Primjedbi 4.3, str.80.

6.4.2 Shubertova metoda

Shubertova metoda (t.4.3, str.84) implementirana je modulom

```
Shubert[f_, a_, b_, L_, It_, Ind_:1, S1_:1]
```

gdje su argumenti jednako označeni kao kod *Metode Pijavskog* (t.6.4.1). Za razliku od *Metode Pijavskog*, ovdje nije potrebno odabrati početnu točku: početna točka jedan je od rubova intervala $[a, b]$. Modul *Shubertove metode* održava listu podintervala i pripadnu \mathcal{B} -vrijednost (vidi Algoritam 11, str.85). U svakoj iteraciji podinterval najmanje \mathcal{B} -vrijednosti izbacuje se iz liste, a dodaju se dva novonastala intervala. Treba uočiti da je ovaj proces potpuno analogan održavanju liste vrhova donje ograde kao kod *Metode Pijavskog*. Naime, usporedimo li (4.11) i (4.15)-(4.16), jasno se vidi da računanje novih vrhova u *Metodi Pijavskog* odgovara računanju vrijednosti U i \mathcal{B} u *Shubertovoj metodi*.

6.4.3 Optimizacijski algoritam DIRECT – poznata Lipschitzova konstanta

Za *Algoritam DIRECT* za jednodimenzionalnu minimizaciju (t.4.4, str.86) napravljena su dva modula s obzirom na to je li Lipschitzova konstanta poznata ili ne. *Algoritam DIRECT* uz poznatu Lipschitzovu konstantu (Algoritam 12, str.88) zapravo je samo jedna varijacija *Metode slomljenih pravaca*. Odgovarajući modul definiran je s:

```
DIRECT1DL[f_, a_, b_, L_, It_, Ind_:1, S1_:1]
```

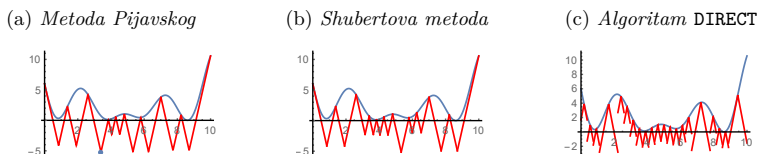
gdje su argumenti jednako označeni kao kod *Metode Pijavskog* (t.6.4.1), odnosno *Shubertove metode* (t.6.4.2). Implementacija metode vrlo je slična kao kod *Shubertove metode* uz razliku što je ovdje svaki interval predstavljen centrom i poluširinom koje se koriste za izračunavanje \mathcal{B} -vrijednosti.

Primjer 6.11. Uporabu modula za tri navedene metode ilustrirat ćemo na problemu minimizacije funkcije $f: [0, 10] \rightarrow \mathbb{R}$, $f(x) = (x - 5)\sin(2x) + 0.2(x - 5)^2 + 1$. Funkcija f na promatranom intervalu uz jedan globalni minimum koji se postiže za $x \approx 8.60851$ ima i tri lokalna minimuma.

Lako je provjeriti da je zadana funkcija Lipschitz-neprekidna te da se Lipschitzova konstanta može izračunati kao $L = |f'(0)| = 12$. Module koji će provesti deset iteracija uz detaljan ispis i grafički prikaz pozivamo s

```
In[1]:= f[x_] := (x - 5)*Sin[2 x] + 0.2*(x - 5)^2 + 1
        a = 0; b = 10;
        L = Abs[f'[a]];
        Pijavskij[f, a, b, L, a, 10]
        Shubert[f, a, b, L, 10]
        DIRECT1DL[f, a, b, L, 10]
```

pri čemu je u *Metodi Pijavskog* početna točka lijevi rub intervala. Na Slici 6.12 prikazana je funkcija f i odgovarajuće donje ograde nakon 10 iteracija za svaku metodu posebno.



Slika 6.12: Funkcija $f: [0, 10] \rightarrow \mathbb{R}$, $f(x) = (x - 5)\sin(2x) + 0.2(x - 5)^2 + 1$ i odgovarajuće donje ograde nakon 10 iteracija

Povećamo li broj iteracija, niz dobivenih aproksimacija konvergirat će prema točki globalnog minimuma funkcije f . Dobivene vrijednosti nakon 1000 iteracija možemo usporediti s aproksimacijom dobivenom ugrađenim modulom `NMinimize`. Zbog brzine izvršavanja isključen je grafički prikaz i ispis:

```
In[2]:= Pijavskij[f, a, b, L, a, 1000, 0, 0]
Out[2]:= {0.00278399, {x -> 8.61304}}

In[3]:= Shubert[f, a, b, L, 1000, 0, 0]
Out[3]:= {0.0026467, {x -> 8.60719}}

In[4]:= DIRECT1DL[f, a, b, L, 1000, 0, 0]
Out[4]:= {0.00263412, {x -> 8.60851}}

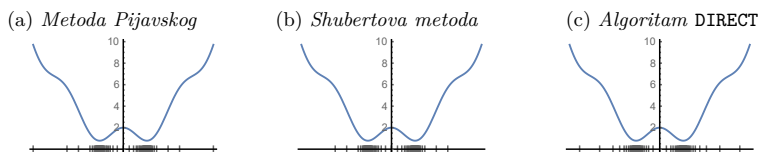
In[5]:= NMinimize[{f[x], a <= x <= b}, x]
Out[5]:= {0.125391, {x -> 3.81931}}
```


Ugrađeni modul `NMinimize` uspio je pronaći samo lokalni minimum dok su opisane metode dale dobru aproksimaciju točke globalnog minimuma.

Primjer 6.12. Promotrimo funkciju $f: [-2, 2] \rightarrow \mathbb{R}$, $f(x) = \cos(5x) + \frac{15x^2}{2\pi} + 1$ koja globalni minimum postiže u dvije točke $x_1^* = -\pi/6$ i $x_2^* = \pi/6$.

Zadana funkcija je Lipschitz-neprekidna te Lipschitzova konstanta iznosi $L = |f'(2)| \approx 12.269$. Na Slici 6.13 prikazan je graf funkcije f dok je crticama na apscisi označen niz aproksimacija dobiven *Metodom Pijavskog* i *Shubertovom metodom*, te niz centara dobiven *Algoritmom DIRECT*. Pripadni niz aproksimacija može se dobiti pozivom modula uz ispis.

```
In[1]:= f[x_] := Cos[5*x] + 15*x^2/(2*Pi) + 1
a=-2; b=2;
L = Abs[f'[a]];
Pijavskij[f, a, b, L, a, 1000, 1, 0]
Shubert[f, a, b, L, 1000, 1, 0]
DIRECT1DL[f, a, b, L, 1000, 1, 0]
```



Slika 6.13: Funkcija $f: [-2, 2] \rightarrow \mathbb{R}$, $f(x) = \cos(5x) + \frac{15x^2}{2\pi} + 1$ s prikazom niza aproksimacija

Na Slici 6.13 može se jasno uočiti da niz aproksimacija ima dva gomilišta u točkama globalnog minimuma promatrane funkcije. Isto možemo primijetiti promotrimo li posljednjih deset od ukupno 1000 aproksimacija dobivenih gornjim pozivima modula:

```
Out[1]:={-0.520803,0.520745,-0.525971,0.526023,-0.526023,
0.525971,-0.526076,0.526129,-0.526129,0.526076}
```

```
Out[2]:={-0.520803,-0.525971,0.526023,-0.526023,0.525971,
-0.526076,0.526129,-0.526129,0.526076,0.520688}
```

```
Out[3]:={-0.528442,0.525191,-0.525191,0.522007,-0.522007,
0.525168,-0.525168,0.52203,-0.52203,-0.518756}
```

6.4.4 Optimizacijski algoritam DIRECT – nepoznata Lipschitzova konstanta

Ukoliko Lipschitzova konstanta nije poznata, što je obično slučaj u praktičnim primjerima, tada u *Algoritmu* DIRECT treba uključiti izbor potencijalno optimalnih intervala kao što je opisano u t.4.4.1, str.89. Koristit ćemo Definiciju 4.2, str.90, potencijalno optimalnog intervala, odnosno uvjete dane Lemom 4.2, str.92. Odgovarajući modul definiran je s

DIRECT1D[f_, a_, b_, It_, Ind_:1, Sl_:1]

gdje su argumenti definirani kao kod ostalih modula iz ovog odjeljka. Za razliku od modula DIRECT1DL (t.6.4.3), modul DIRECT1D ne zahtijeva Lipschitzovu konstantu L. Osim ispisa aproksimacija po iteracijama, modul omogućava i jednostavni grafički prikaz funkcije i centara intervala označenih crticama na apscisi.

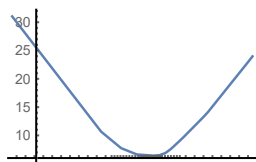
Primjer 6.13. *Promatramo problem određivanja najboljeg LAD pravca kroz ishodište na osnovu skupa $\mathcal{A} = \{(x_i, y_i) : i = 1, \dots, m\}$ točaka u ravnini (vidi Primjer 1.15, str.20). Za odrediti takav pravac treba odrediti minimizator funkcije*

$$F_{LAD}(k) = \sum_{i=1}^m |y_i - kx_i|.$$

Za konkretan primjer uzet ćemo $m = 10$ podataka generiranih oko pravca $y = 2x$ na sljedeći način

$$\begin{aligned} x_i &= i, & i &= 1, \dots, 100, \\ y_i &= 2x_i + \varepsilon_i, & i &= 1, \dots, 100, \end{aligned}$$

gdje je ε_i pseudoslučajan broj generiran iz normalne distribucije s očekivanjem 0 i varijancom 1.



Slika 6.14: Graf funkcije F_{LAD} s prikazom centara intervala

Funkcija F_{LAD} je konveksna (vidi Zadatak 1.27, str.26), no nije diferencijabilna, stoga primjerice metode spusta iz t.2, str.35 nisu primjenjive. Lipschitzovu

konstantu funkcije F_{LAD} nije jednostavno odrediti. Crtanjem grafa funkcije lako je lokalizirati interval u kojemu se nalazi globalni minimum, primjerice [1.5, 2.5]. Primjenom *Algoritma* DIRECT u malom broju iteracija može se odrediti vrlo dobra aproksimacija točke globalnog minimuma.

```
In[1]:= m = 10;
      xx = Table[i, {i, 1, m}];
      yy = 2*xx + RandomVariate[NormalDistribution[0, 1], m];
      FLAD[k_] := Sum[Abs[yy[[i]] - k*xx[[i]]], {i, 1, m}]
```

```
In[2]:= DIRECT1D[FLAD, 1.5, 2.5, 20]
Out[2]:= {6.43202, {x -> 2.08729}}
```

```
In[3]:= NMinimize[{FLAD[x], 1.5 <= x <= 2.5}, x]
Out[3]:= {6.43202, {x -> 2.08729}}
```

Grafički prikaz dobiven pozivom modula može se vidjeti na Slici 6.14.

6.5 Višedimenzionalna globalna optimizacija

Metode globalne optimizacije za Lipschitz-neprekidnu funkciju $f: \mathcal{D} \rightarrow \mathbb{R}$, gdje je $\mathcal{D} \subset \mathbb{R}^n$ hiperpravokutnik u \mathbb{R}^n , opisane u t.5, str.101 pronalaze aproksimaciju globalnog minimuma funkcije f . U nastavku su opisani moduli koji implementiraju ove metode.

6.5.1 Optimizacijski algoritam DIRECT za funkciju više varijabli

Algoritam DIRECT (t.5.3, str.109) pronalazi aproksimacije svih točaka globalnog minimuma Lipschitz-neprekidnu funkciju $f: \mathcal{D} \rightarrow \mathbb{R}$, gdje je $\mathcal{D} \subset \mathbb{R}^n$ hiperpravokutnik u \mathbb{R}^n , a implementirana je modulom¹

```
DIRECT[f_, dom_, DMin -> 10^(-2), MaxIter -> 50, Eps -> 10^(-4)]
```

gdje je f Lipschitz neprekidna funkcija čiji se globalni minimum traži, dom je domena funkcije f , parametar $DMin$ najmanja je dozvoljena veličina hiperpravokutnika u postupku dijeljenja, $MaxIter$ je maksimalno dopušteni broj iteracija, a Eps predstavlja parametar $\epsilon > 0$ iz Leme 4.3, str.95. Preporučena vrijednost parametra Eps je 10^{-4} [13].

Primjer 6.14. *Naka je $\mathcal{A} \subset \mathbb{R}$ skup iz Primjera 5.7, str.112. Potražiti ćemo 2-optimalnu partitciju skupa \mathcal{A} primjenom Algoritma DIRECT.*

¹Modul je izradio doc.dr.sc. Ivan Vazler, Odjel za fiziku, Sveučilište u Osijeku

Najprije definiramo skup \mathcal{A} , minimizacijsku funkciju i njenu domenu, a zatim pozovemo modul

```
In[1]:= A = Flatten[podaci]; m = Length[A];
      F2[c1_, c2_] := Sum[Min[{(c1 - A[[i]])^2, (c2 - A[[i]])^2}], {i, m}];
      dom2 = {{0, 1}, {0, 1}};
      Timing[
        DIRECT[F2, dom2, DMin -> 10^(-2), MaxIter -> 50, Eps -> 10^(-4)]
      ]
```

Kod imlementacije modula uključili smo mjerenje potrebnog CPU vremena. Dobivamo centre klastera optimalne particije $c_1^* = 0.6893$, $c_2^* = 0.3930$, vrijednost minimizacijske funkcije (suma kvadrata rasipanja podataka oko centara) $F_2(c_1^*, c_2^*) = 2855.35$ i CPU vrijeme 153.05 sec.

```
Out[2]:= {153.046875, {{0.6893004115, 0.3930041152}, 2855.35}}
```

6.5.2 Optimizacijski algoritam SymDIRECT za funkciju više varijabli

Algoritam SymDIRECT (t.5.4, str.111) znatno efikasnije od *Algoritma* DIRECT pronalazi aproksimaciju jedne točke globalnog minimuma simetrične Lipschitz neprekidne funkcije $f: [a, b]^n \rightarrow \mathbb{R}$, a implementirana je modulom²

```
SymDIRECT[f_, dom_, DMin -> 10^(-2), MaxIter -> 50, Eps -> 10^(-4)]
```

gdje je f Lipschitz neprekidna funkcija čiji se globalni minimum traži, dom je domena funkcije f , parametar $DMin$ najmanja je dozvoljena veličina hiperpravokutnika u postupku dijeljenja, $MaxIter$ je maksimalno dopušteni broj iteracija, a Eps predstavlja parametar $\epsilon > 0$ iz Leme 4.3, str.95. Preporučena vrijednost parametra Eps je 10^{-4} [13].

Primjer 6.15. *Naka je $\mathcal{A} \subset \mathbb{R}$ skup iz Primjera 5.7, str.112. Potržit ćemo 2-optimalnu particiju skupa \mathcal{A} primjenom Algoritma SymDIRECT.*

Najprije definiramo skup \mathcal{A} , minimizacijsku funkciju i njenu domenu, a zatim pozovemo modul

```
In[1]:= A = Flatten[podaci]; m = Length[A];
      F2[c1_, c2_] := Sum[Min[{(c1 - A[[i]])^2, (c2 - A[[i]])^2}], {i, m}];
      dom2 = {{0, 1}, {0, 1}};
      Timing[
        SymDIRECT[F2, dom2, DMin -> 10^(-2), MaxIter -> 50, Eps -> 10^(-4)]
      ]
```

²Modul je izradio doc.dr.sc. Ivan Vazler, Odjel za fiziku, Sveučilište u Osijeku

Kod imlementacije modula uključili smo mjerenje potrebnog CPU vremena. Dobivamo centre klastera optimalne particije i vrijednost minimizacijske funkcije kao i kod primjene *Algoritma DIRECT* te CPU vrijeme 87.47 sec.

```
Out [2] := {87.406250, {{0.6893004115, 0.3930041152}, 2855.35}}
```

Kod složenijih primjera kao što je Primjer 5.5, str.111 (vidi također Primjer 1.10, str.15) efikasnost *Algoritma SymDIRECT* dolazi znatno više do izražaja [16, 63].

Literatura

- [1] C. ADJIMAN, S. DALLWIG, C. FLOUDAS, A. NEUMAIER, *A global optimization method, αBB , for general twice-differentiable constrained NLPs – I. Theoretical advances*, Computers and Chemical Engineering, **22**(1998) 1137–1158.
- [2] I. P. ANDROULAKIS, C. D. MARANAS, C. A. FLOUDAS, *αBB : A global optimization method for general constrained nonconvex problems*, Journal of Global Optimization, **7**(1995) 337–363.
- [3] M. AVRIEL, *Nonlinear Programming: Analysis and Methods.*, Dover Publications, Inc. Mineola, New York, 2006.
- [4] A. M. BAGIROV, J. UGON, D. WEBB, *Fast modified global k-means algorithm for incremental cluster construction*, Pattern Recognition, **44**(2011) 866–876.
- [5] J. F. BONNANS, J. C. GILBERT, C. LEMARECHAL, C. A. SAGASTIZABAL, *Numerical Optimization. Theoretical and Practical Aspects*, Springer, Berlin, 2006.
- [6] R. J. BOSCOVICH, *De litteraria expeditione per pontificiam ditionem, et synopsis amplioris operis, ac habentur plura eius ex exemplaria etiam sensorum impressa*, Bononienci Scientiarum et Artium Znstituto Atque Academia Commentarrii, **4**(1757) 353–396.
- [7] D. L. BOYD, L. VANDENBERGHE, *Convex Optimization*, Cambridge University Press, Cambridge, 2004.
- [8] R. CHELOUAH, P. SIARRY, *Tabu search applied to global optimization*, European Journal of Operational Research, **123**(2000) 256–270.
- [9] R. CUPEC, R. GRBIĆ, K. SABO, R. SCITOVSKI, *Three points method for searching the best least absolute deviations plane*, Applied Mathematics and Computation, **215**(2009) 983–994.

- [10] J. J. DENNIS, R. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, Philadelphia, 1996.
- [11] Z. DREZNER, H. W. HAMACHER, *Facility Location: Applications and Theory*, Springer, 2004.
- [12] J. E. FALK, R. M. SOLAND, *An algorithm for separable nonconvex programming problems*, *Management Science*, **15**(1969) 55–569.
- [13] D. E. FINKEL, *DIRECT Optimization Algorithm User Guide*, Center for Research in Scientific Computation. North Carolina State University, 2003, <http://www4.ncsu.edu/definkel/research/index.html>.
- [14] J. M. GABLONSKY, *Direct version 2.0*, Technical report, Center for Research in Scientific Computation. North Carolina State University, 2001.
- [15] R. GRBIĆ, D. GRAHOVAC, R. SCITOVSKI, *A method for solving the multiple ellipses detection problem*, *Pattern Recognition*, **60**(2016) 824–834.
- [16] R. GRBIĆ, E. K. NYARKO, R. SCITOVSKI, *A modification of the DIRECT method for Lipschitz global optimization for a symmetric function*, *Journal of Global Optimization*, **57**(2013) 1193–1212.
- [17] C. GURWITZ, *Weighted median algorithms for l_1 approximation*, *BIT*, **30**(1990) 301–310.
- [18] E. R. HANSEN, *Global optimization using interval analysis: The one-dimensional case*, *Journal of Optimization Theory and Applications*, **29**(1979) 331–344.
- [19] E. R. HANSEN, G. W. WALSTER, *Global Optimization Using Interval Analysis*, Marcel Dekker, INC., New York-Basel, 1992.
- [20] E. M. T. HENDRIX, B. G. TÓTH, *Introduction to Nonlinear and Global Optimization*, Springer, 2010.
- [21] R. HORST, H. TUY, *Global Optimization: Deterministic Approaches*, Springer, Berlin, third edition, 1996.
- [22] C. IYIGUN, A. BEN-ISRAEL, *A generalized Weiszfeld method for the multi-facility location problem*, *Operations Research Letters*, **38**(2010) 207–214.

- [23] M. JAMIL, X.-S. YANG, *A literature survey of benchmark functions for global optimization problems*, International Journal of Mathematical Modelling and Numerical Optimisation, **4**(2013) 150–194.
- [24] D. JANKOV MAŠIREVIĆ, S. MIODRAGOVIĆ, *Geometric median in the plane*, Elemente der Mathematik, **70**(2015) 21–32.
- [25] F. JARRE, J. STOER, *Optimierung*, Springer, Berlin, Heidelberg, 2004.
- [26] D. R. JONES, C. D. PERTTUNEN, B. E. STUCKMAN, *Lipschitzian optimization without the Lipschitz constant*, Journal of Optimization Theory and Applications, **79**(1993) 157–181.
- [27] A. JOVIČIĆ, K. SABO, *Formule za udaljenost točke do pravca u ravnini, u smislu l_p -udaljenosti, $1 \leq p \leq \infty$* , math.e, Hrvatski matematički elektronski časopis, **29**(2016).
- [28] D. JUKIĆ, *Konveksni skupovi*, Odjel za matematiku, Sveučilište u Osijeku, 2015.
- [29] D. JUKIĆ, R. SCITOVSKI, *Existence of optimal solution for exponential model by least squares*, J. Comput. Appl. Math., **78**(1997) 317–328.
- [30] D. JUKIĆ, R. SCITOVSKI, *Matematika I*, Odjel za matematiku, Sveučilište u Osijeku, 2004.
- [31] D. JUKIĆ, R. SCITOVSKI, Š. UNGAR, *The best total least squares line in r^3* , In: I. AGANOVIĆ, T. HUNJAK, R. SCITOVSKI, editors, *Proceedings of the 7th International Conference on Operational Research KOI98*, volume 13, 1993, 311–316.
- [32] J. KOGAN, *Introduction to Clustering Large and High-dimensional Data*, Cambridge University Press, New York, 2007.
- [33] S. KUCHERENKO, Y. SYTSKO, *Application of deterministic low-discrepancy sequences in global optimization*, Computational Optimization and Applications, **30**(2004) 297–318.
- [34] S. KUREPA, *Uvod u linearnu algebru*, Školska knjiga, Zagreb, 1985.
- [35] I. KUZMANOVIĆ, K. SABO, *Linearno programiranje*, Odjel za matematiku, Sveučilište u Osijeku, 2016.

- [36] J. C. LAGARIAS, J. A. REEDS, M. H. WRIGHT, M. H. P. WRIGHT, *Convergence properties of the Nelder-Mead simplex algorithm in low dimensions*, SIAM J. Optim., **9**(1998) 112–147.
- [37] L. LIBERTI, *Introduction to Global Optimization*, LIX, École Polytechnique, 2008.
- [38] Y.-S. LIU, K. RAMANI, *Robust principal axes determination for point-based shapes using least median of squares*, Computer-Aided Design, **41**(2009) 293–305.
- [39] G. LIUZZI, S. LUCIDI, V. PICCIALLI, *A direct-based approach for large-scale global optimization problems*, Computational Optimization and Applications, **45**(2010) 353–375.
- [40] D. MATIJEVIĆ, N. TRUHAR, *Uvod u računarstvo*, Odjel za matematiku, Sveučilište u Osijeku, 2012.
- [41] E. MELACHRINOUDIS, *An analytical solution to the minimum l_p -norm of a hyperplane*, Journal of Mathematical Analysis and Applications, **211**(1997) 172–189.
- [42] N. MLADENOVIĆ, J. PETROVIĆ, V. KOVAČEVIĆ-VUJČIĆ, M. ČANGALLOVIĆ, *Solving a spread spectrum radar polyphase code design problem by tabu search and variable neighbourhood search*, European Journal of Operations Research, **15**(2003) 389–399.
- [43] A. MORALES-ESTEBAN, F. MARTÍNEZ-ÁLVAREZ, S. SCITOVSKI, R. SCITOVSKI, *A fast partitioning algorithm using adaptive Mahalanobis clustering with application to seismic zoning*, Computers & Geosciences, **73**(2014) 132–141.
- [44] J. A. NELDER, R. MEAD, *A simplex method for function minimization*, The Computer Journal, **7**(1965) 308–313.
- [45] Y. NIEVERGELT, *Total least squares: state-of-the-art regression in numerical analysis*, SIAM Review, **36**(1994) 258–264.
- [46] J. M. ORTEGA, W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, SIAM, Philadelphia, 2000.
- [47] C. H. PAPADIMITRIOU, K. STEIGLITZ, *Combinatorial Optimization: Algorithms and Complexity*, Dover, New York, 1998.

- [48] J. PARAJKA, S. KOHNOVÁ, G. BÁLINT, M. BARBUC, M. BORGA, P. CLAPS, S. C. A. DUMITRESCU, E. GAUME, K. HLAVČOVÁ, R. MERZ, M. PFAUNDLER, G. STANCALIE, J. SZOLGAY, G. BLÖSCHL, *Seasonal characteristics of flood regimes across the Alpine-Carpathian range*, Journal of Hydrology, **394**(2010) 78–89.
- [49] R. PAULAVIČIUS, J. ŽILINSKAS, *Simplicial Global Optimization*, volume X of *Series: Springer Briefs in Optimization*, Springer, Berlin, 2014.
- [50] J. D. PINTÉR, *Global Optimization in Action (Continuous and Lipschitz Optimization: Algorithms, Implementations and Applications)*, Kluwer Academic Publishers, Dordrecht, 1996.
- [51] W. H. PRESS, B. P. FLANNERY, S. A. TEUKOLSKY, W. T. VETTERLING, *Numerical Recipes*, Cambridge University Press, Cambridge, 1992.
- [52] K. V. PRICE, R. M. STORN, J. A. LAMPINEN, *Differential Evolution: A Practical Approach to Global Optimization*, Springer, Berlin, Heidelberg, 2005.
- [53] S. RAJASEKARAN, *On simulated annealing and nested annealing*, Journal of Global Optimization, **16**(2000) 43–56.
- [54] P. J. ROUSSEEUW, A. M. LEROY, *Robust Regression and Outlier Detection*, Wiley, New York, 2003.
- [55] K. SABO, R. SCITOVSKI, *The best least absolute deviations line – properties and two efficient methods*, ANZIAM Journal, **50**(2008) 185–198.
- [56] K. SABO, R. SCITOVSKI, *An approach to cluster separability in a partition*, Information Sciences, **305**(2015) 208–218.
- [57] K. SABO, R. SCITOVSKI, I. VAZLER, *Grupiranje podataka - klasteri*, Osječki matematički list, **10**(2010) 149–178.
- [58] K. SABO, R. SCITOVSKI, I. VAZLER, *One-dimensional center-based l_1 -clustering method*, Optimization Letters, **7**(2013) 5–22.
- [59] K. SABO, R. SCITOVSKI, I. VAZLER, M. ZEKIĆ-SUŠAC, *Mathematical models of natural gas consumption*, Energy Conversion and Management, **52**(2011) 1721–1727.

- [60] K. SABO, P. TALER, Z. BERTIĆ, *Mathematics and politics: How to determine optimal constituencies in Republic of Croatia*, Croatian Operational Research News, **2(1)**(2015) 10–12.
- [61] R. SCITOVSKI, *Problemi najmanjih kvadrata. Financijska matematika*, Ekonomski fakultet, Elektrotehnički fakultet, Sveučilište u Osijeku, 1993.
- [62] R. SCITOVSKI, *Numerička matematika*, Odjel za matematiku, Sveučilište u Osijeku, 3, izdanje, 2015.
- [63] R. SCITOVSKI, *A new global optimization method for a symmetric lipschitz continuous function and application to searching for a globally optimal partition of a one-dimensional set*, Journal of Global Optimization, (2017), DOI: 10.1007/s10898-017-0510-4.
- [64] R. SCITOVSKI, M. B. ALIĆ, *Grupiranje podataka*, Odjel za matematiku, Sveučilište u Osijeku, 2016.
- [65] R. SCITOVSKI, N. TRUHAR, Z. TOMLJANOVIĆ, *Metode optimizacije*, Odjel za matematiku, Sveučilište u Osijeku, 2014.
- [66] Y. D. SERGEYEV, D. E. KVASOV, *Diagonal Global Optimization Methods*, FizMatLit, Moscow, 2008, (in Russian).
- [67] S. SINGER, J. NELDER, *Nelder-Mead algorithm*, Scholarpedia, **4**(2009) 2928.
- [68] G. W. STEWART, *Afternotes on Numerical Analysis*, SIAM, Philadelphia, 1996.
- [69] J. STOER, R. BULIRSCH, *Introduction to Numerical Analysis*, Springer, New York, 2002.
- [70] R. G. STRONGIN, Y. D. SERGEYEV, *Global Optimization with Non-Convex Constraints: Sequential and Parallel Algorithms*, Kluwer Academic Publishers, Dordrecht, 2000.
- [71] E. SÜLI, D. MAYERS, *An Introduction to Numerical Analysis, 2nd Ed.*, Cambridge University Press, Second printing, 2006.
- [72] W. SUN, Y. YUAN, *Optimization theory and methods. Nonlinear programming*, Springer, New York, 2006.

- [73] S. THEODORIDIS, K. KOUTROUMBAS, *Pattern Recognition*, Academic Press, Burlington, 2009, 4th edition.
- [74] N. TRUHAR, *Numerička linearna algebra*, Odjel za matematiku, Sveučilište u Osijeku, 2010.
- [75] ŠIME UNGAR, *Matematička analiza 3*, Sveučilište u Zagrebu, Prirodoslovno–matematički fakultet, Matematički odjel, 2002.
- [76] I. WOLFRAM RESEARCH, *Mathematica*, Wolfram Research, Inc., Champaign, Illinois, 2016, version 11.0 edition.
- [77] G. R. WOOD, B. P. ZHANG, *Estimation of the Lipschitz constant of a function*, *Journal of Global Optimization*, **8**(1996) 91–103.
- [78] Y. YAO, *Dynamic tunnelling algorithm for global optimization*, *IEEE Transactions On Systems Man and Cybernetics*, **19**(1989) 1222–1230.

Indeks

- \mathcal{B} -vrijednost, 87, 105
- Algoritam
 - DIRECT za funkciju dviju varijabli, 104, 143
 - DIRECT za funkciju jedne varijable, 86, 88, 139
 - DIRECT za funkciju jedne varijable bez poznavanja Lipschitzove konstante, 89, 142
 - DIRECT za funkciju više varijabli, 109, 111, 143
 - DIRECT za simetričnu funkciju, 111, 144
 - izbora duljine koraka, 39, 123
 - Metoda dvije točke, 48, 128
 - Metoda polovljenja za kvazikonveksnu derivabilnu funkciju, 58, 133
 - Metoda polovljenja za nederivabilnu strogo kvazikonveksnu funkciju, 62, 134
 - Metoda slomljenih pravaca, 76, 138
 - Metoda tangenti, 55, 131
 - Metoda tri točke, 51, 128
 - Metoda zlatnog reza, 67, 135
 - Nelder-Meadova metoda, 45, 126
 - Newtonova metoda, 38, 123
 - Shubertova metoda, 85, 139
- Branin funkcija, 101, 126
- Centroid, 9
- Geometrijski medijan, 11
- Globalni
 - maksimum, 2
 - minimum, 2, 26, 29
- Gradijent, 33
- Gradijentna nejednakost, 25, 54
- Hessian, 34
- Klasteri, 14, 112
- Konveksan skup, 24
- Konveksna funkcija, 24
- Kvazikonveksna funkcija, 27
- Lipschitz-neprekidna funkcija, 69, 72
- Lokalni minimum, 2, 26, 29
- Medijan, 10
- Metoda
 - dvije točke, 47, 128
 - gradijentna, 37, 123
 - grananja i ograđivanja, 117
 - koordinatne relaksacije, 36, 122
 - kvazi-Newtonova, 41
 - Nelder-Meadova, 41, 126
 - Newtonova s regulacijom koraka, 39, 123
 - Newtonova za minimizaciju bez ograničenja, 37, 46, 123
 - polovljenja za kvazikonveksnu derivabilnu funkciju, 57, 133

- polovljenja za nederivabilnu strogo kvazikonveksnu funkciju, 60, 134
- Shubertova, 84, 139
- slomljenih pravaca (Pijavskij), 74, 138
- spusta za konveksne funkcije, 35, 122
- tangenti, 53, 131
- tri točke, 49, 128
- zlatnog reza, 63, 135
- Najbolji
 - LAD-pravac, 20
 - LS-pravac, 19
 - OD-pravac, 22
 - TLS-pravac, 22
- Optimizacijski problem, 2
- Outlier, 12, 98
- Potencijalno optimalan
 - interval, 90, 92, 95
 - subpravokutnik, 106
 - subtrokut, 114
- Segmentacija
 - crno-bijele slike, 15, 111
 - slike u boji (HSI zapis), 18
 - slike u boji (RGB zapis), 15
- Simetrična funkcija, 111
- Stacionarna točka, 34
- Stohastička optimizacija, 117
- Taylorova formula, 34
- Udaljenost
 - točke do grafa funkcije, 8
 - točke do parabole, 7, 8
 - točke do pravca, 3
- Unimodalna funkcija, 32
- Zlatni rez, 64

ISBN 978-953-8154-06-5



9 789538 154065