Univerzitet u Nišu
Prirodno-matematički fakultet

Simbolička izračunavanja u višekriterijumskoj optimizaciji

Ivan P. Stanimirović

Niš, Februar 2015.

1. Uvod

Modeli za nalaženje optimuma jedne kriterijumske funkcije su obično samo aproksimacija realnih problema u kojima donosilac odluke mora da vodi računa o više ciljeva. Postoje matematički modeli i metodi u kojima donosilac odluke analizira i bira rešenja na osnovu više kriterijuma koji se istovremeno razmatraju. Pritom, kao i u slučaju jednokriterijumske optimizacije, donosilac odluke implicitno zadržava slobodu da prihvati, promeni ili odbaci rešenje dobijeno na osnovu matematičkog modela optimizacije. Metode koje od samog početka formiranja matematičkog modela za određeni realni problem vode računa o više ciljeva istovremeno razvijaju se u oblasti višekriterijumske optimizacije (VKO). Ovaj deo matematičkog programiranja svoj buran razvoj ima od kraja sedamdesetih godina dvadesetog veka.

Postoji više razloga koji utiču na to da su problemi VKO suštinski drugačiji u odnosu na probleme jednokriterijumske optimizacije. Osnovni je u tome što se svi faktori koji utiču na odluku, odnosno svi ishodi koje bi imalo eventualno rešenje, posmatraju kao kriterijumi čije bi vrednosti trebalo da budu optimalne. Dakle, potrebno je naći rešenje koje je najbolje po svim razmatranim kriterijumima istovremeno a činjenica je da su neki od njih u skoro svim problemima odlučivanja međusobno konfliktni. Pored toga, razmatrani kriterijumi mogu po svojoj prirodi biti veoma raznorodni i izraženi u različitim mernim jedinicama, od novčanih jedinica, preko jedinica fizičkih veličina, do verovatnoća ili subjektivnih procena datih po nekoj skali koja se formira za konkretni problem. Sve ovo ukazuje da konačno rešenje ne može da se odredi bez učešća donosioca odluke.

Zadatke višekriterijumske optimizacije u slučajevima kada se razmatraju važne odluke kao što su odluke u vezi sa kapitalnim ulaganjima u opremu, karakteriše relativno veliki broj kriterijuma. Što je broj kriterijuma veći, zadaci višekriterijumske analize su složeniji i teži. U ovakvim situacijama, u odlučivanju učestvuje veći broj pojedinaca ili grupa i svi oni favorizuju svoje sisteme vrednosti. Radi efikasnijeg analiziranja odluke i pronalaženja pogodnog rešenja vrši se grupisanje kriterijuma. Uobičajene su sledeće vrste kriterijuma: ekonomski, tehnički, tehnološki, socijalni i ekološki.

1.1 Preliminarije

Iako je linearno programiranje veoma primenljivo u praksi, mnoge probleme iz prakse je nemoguće adekvatno linearizovati a da se pritom drastično ne izgubi na tačnosti. U tom slučaju primenjuju se metodi nelinearnog programiranja. Osim nelinearnosti, u mnogim problemima je potrebno naći optimum više od jedne funkcije cilja. U tom slučaju, moramo rešavati ***problem višekriterijumske optimizacije***. Ukoliko sve funkcije cilja imaju optimum u istoj tački, problem je trivijalan i direktno se svodi na problem nelinearnog ili linearnog programiranja. U praksi je ova situacija veoma retka. Postoji više metoda za rešavanje problema višekriterijumske optimizacije [?]. Zajedničko svim tim metodima je da se polazni problem na odgovarajući način svodi na problem linearnog i nelinearnog programiranja.

Višekriterijumska optimizacija se može posmatrati kao nastavak istraživanja u klasičnoj (jednokriterijumskoj) optimizaciji, uz izvesna proširenja. Formalno, osnovno proširenje je uvođenje vektorske kriterijumske funkcije, što dovodi do problema vektorskog maksimuma. Suštinski, potrebno je da se proširi koncept optimalnosti. Razmatrajući problem vektorskog maksimuma koncept optimalnosti se zamenjuje konceptom neinferiornosti (Pareto optimalnosti). Može se uvesti pojam opšteg (jedinstvenog) kriterijuma optimizacije, koji uključuje kriterijumske funkcije i preferenciju donosioca odluke. Rešenje zadatka višekriterijumske optimizacije koje se dobija prema takvom kriterijumu je optimalno. U tom slučaju pojam optimalnog rešenja iz klasične optimizacije može se zadržati i u višekriterijumskoj. Međutim, teškoće se upravo javljaju pri pokušaju formalizacije takvog jedinstvenog kriterijuma. Zato se u višekriterijumskoj optimizaciji koriste dve faze ili etape. U prvoj fazi se određuje skup "boljih" rešenja na osnovu vektorske kriterijumske funkcije, a u drugoj se na osnovu preferencije donosioca odluke usvaja konačno rešenje, koje se može nazvati optimalnim. Skup rešenja koji se prezentira donosocu odluke treba da sadrži mali broj rešenja, koja su neinferiorna prema datim kriterijumskim funkcijama. Problem višekriterijumske optimizacije se najčešće javlja u planiranju složenih sistema; na primer, regionalni razvoj, razvoj vodoprivrednih ili elektroprivrednih sistema, urbano planiranje i očuvanje prirodne okoline [19]. Višekriterijumski problem se javlja u ekonomiji kao problem određivanja tržišne ravnoteže ili ekonomija blagostanja u decentralizovanoj ekonomiji [19]. Sličan problem se javlja i kao problem ravnoteže u teoriji igara [19]. U teoriji igara razmatraju se igre sa više igrača, što se u teoriji odlučivanja javlja kao "grupno odlučivanje" ili odlučivanje sa više donosioca odluke.

U ovoj glavi ćemo izložićemo problem višekriterijumske optimizacije, kao i metode za njeno rešavanje. Najpre ćemo dati definiciju problema višekriterijumske optimizacije kao i neophodnih pojmova za kasnija razmatranja. Teorijski ćemo obraditi i dati implementaciju nekoliko klasičnih metoda višekriterijumske optimizacije. Svaki od opisanih metoda biće ilustrovan na jednom ili više primera. Razmatranja vezana za implementaciju metoda su originalna i preuzeta su iz radova [22], [24] kao i iz monografije [23].

Opšta formulacija višekriterijumske optimizacije (VKO) poseduje opšti oblik:

$$
\begin{aligned}
\text{Max:} \quad & Q(\mathbf{x}) = Q_1(\mathbf{x}), \ldots, Q_l(\mathbf{x}), \quad \mathbf{x} \in \mathbf{R}^n \\
\text{p.o:} \quad & f_i(\mathbf{x}) \leq 0, \; i = 1, \ldots, m \\
& h_i(\mathbf{x}) = 0, \; i = 1, \ldots, k.
\end{aligned} \tag{1.1.0.1}
$$

gde su: $Q_1(\mathbf{x}), \ldots, Q_l(\mathbf{x})$, $f_1(x), \ldots, f_m(x), g_1(x), \ldots, g_m(x)$ realne funkcije od n promenljivih koje su sadržane u vektoru $\mathbf{x} = (x_1, \ldots, x_n)$.

U ovom zadatku traži se rešenje $\mathbf{x}$ koje maksimizira svih l funkcija cilja. Zato se zadatak višekriterijumske optimizacije (VKO) naziva i zadatak vektorske optimizacije. Radi jednostavnosti ovde se razmatraju samo problemi maksimizacije. Poznato je da se zadatak minimizacije jednostavno prevodi u zadatak maksimizacije množenjem kriterijumske funkcije sa -1. Sve nadalje izložene definicije i metode moguće je prilagoditi da važe za rešavanje zadatka minimizacije.

Kažemo da je $\mathbf{X} \subseteq R^n$ *skup dopustivih rešenja* ako važi

$$\mathbf{X} = \{\mathbf{x} | f_i(\mathbf{x}) \leq 0, \; i = 1, \ldots, m; \; h_i(\mathbf{x}) = 0, \; i = 1, \ldots, k\}.$$

Svakom dopustivom rešenju $\mathbf{x} \in \mathbf{X}$ odgovara skup vrednosti kriterijumskih funkcija, tj. vektor $Q(\mathbf{x}) = (Q_1(\mathbf{x}), Q_2(x), \ldots, Q_l(\mathbf{x}))$. Na taj način se skup dopustivih rešenja preslikava u *kriterijumski skup*, tj. $S = \{Q(\mathbf{x}) | \mathbf{x} \in \mathbf{X}\}$.

U daljem tekstu biće korišćeni sledeci pojmovi:

- *Marginalna rešenja* zadatka VKO se određuju optimizacijom svake od funkcija cilja pojedinačno nad zadatim dopustivim skupom, tj. rešavanjem l jednokriterijumskih zadataka:

$$
\begin{aligned}
\text{Max:} \quad & Q_j(\mathbf{x}), \quad \mathbf{x} \in \mathbf{R}^n \\
\text{p.o:} \quad & f_i(\mathbf{x}) \leq 0, \; i = 1, \ldots, m \\
& h_i(\mathbf{x}) = 0, \; i = 1, \ldots, k.
\end{aligned}
$$

Marginalna rešenja ćemo obeležavati sa $\mathbf{x}^{(j)^*} = (x_1^{(j)^*}, x_2^{(j)^*}, \ldots, x_n^{(j)^*})$, gde je $x^{(j)^*}$ optimalno rešenje dobijeno optimizacijom j-te funkcije cilja nad zadatiom dopustivim skupom $\mathbf{X}$.

- *Idealne vrednsti funkcija cilja*, označene sa Q_j^* jesu vrednosti funkcija cilja za marginalna rešenja:

$$Q_j^* = Q_j(\mathbf{x}^{(j)^*}), \; j = 1, \ldots, l$$

- Idealne vrednosti funkcija cilja određuju *idealnu tačku* u kriterijumskom prostoru, tj. idealnu vrednost vektorske funkcije

$$Q^* = (Q_1^*, Q_2^*, \ldots, Q_l^*).$$

- Ako postoji rešenje $\mathbf{x}^*$ koje istovremeno maksimizira sve funkcije cilja, tj.

$$\mathbf{x}^* = \{\mathbf{x} | Q_j(\mathbf{x}) = Q_j^*, \; j = 1, \ldots, l\},$$

onda se takvo rešenje naziva *savršeno resenje*.

U najvećem broju slučajeva marginalna rešenja se razlikuju i savršeno rešenje ne postoji. Kada savršeno rešenje postoji, tada se u suštini ne radi o problemu VKO.

Veoma je važno imati u vidu da su u realnim problemima ciljevi gotovo uvek u koliziji, što znači da ne mogu svi biti dostignuti u potpunosti. Zbog toga najčečešće nije moguće strogo definisati optimum niti za svaka dva rešenja formalno odrediti koje je bolje od drugog. Iz tog razloga proces dobijanja rešenja zahteva učešće donosioca odluke (u daljem tekstu DO). To je najčešće neko ko ima dublji uvid u problem i po čijem se zahtevu pristupa rešavanju. Donosilaca odluke može biti i više i tada se problem može dodatno iskomplikovati zbog njihovih različitih ciljeva, udela u odlučivanju i stepena odgovornosti koji su spremni da preuzmu.

2. Pareto optimalnost

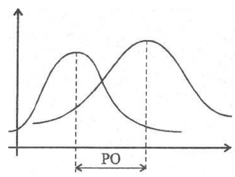
Činjenica da zadaci VKO po pravilu nemaju savršeno rešenje upućuje na preispitivanje koncepta optimalnosti i definicije optimalnog rešenja. Ključnu ulogu u tome ima koncept *Pareto optimalnosti.* To je proširenje poznatog koncepta optimalnosti koj se koristi u klasičnoj jednokriterijumskoj optimizaciji.

Pareto optimum se definiše na sledeći način:

- Dopustivo rešenje $\mathbf{x}^*$ predstavlja *Pareto optimum* zadatka VKO ako ne postoji neko drugo dopustivo rešenje $\mathbf{x}$ takvo da važi:

$$Q_j(\mathbf{x}) \geq Q_j(\mathbf{x}^*) \quad \forall j = 1, \ldots, l$$

pri čemu bar jedna od nejednakosti prelazi u strogu nejednakost $>$. Drugim rečima, $\mathbf{x}$ je Pareto optimum ako bi poboljšanje vrednosti bilo koje funkcije cilja prouzrokovalo pogoršanje vrednosti neke druge funkcije cilja. Za Pareto oprimum postoje sledeći sinonimi: *efikasno, dominantno i nedominirano rešenje.*



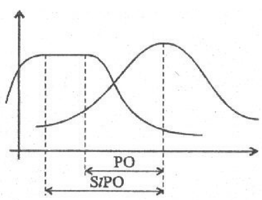
Slika 2.2.1. Geometrijska interpretacija Pareto optimalnih rešenja

Pored Pareto optimuma definišu se slabi i strogi (jaki) Pareto optimumi:

Dopustivo rešenje $\mathbf{x}^*$ je *slabi Pareto optimum* ako ne postoji neko drugo dopustivo rešenje x takvo da važi:

$$Q_j(\mathbf{x}) > Q_j(\mathbf{x}^*) \quad \forall j = 1, \ldots, l.$$

Drugim rečima $\mathbf{x}^*$ je slabi Pareto optimum ako nije moguće istovremeno poboljšati sve funkcije cilja.



Slika 2.2.2. Geometrijska interpretacija Pareto optimalnih i Slabo Pareto optimalnih rešenja

Pareto optimalno rešenje $\mathbf{x}^*$ je *strogi Pareto optimum* ako postoji broj $\beta > 0$ takav da za svaki indeks $j \in \{1, \dots, l\}$ i za svako $\mathbf{x}$ koje zadovoljava uslov

$$Q_j(\mathbf{x}) > Q_j(\mathbf{x}^*)$$

postoji bar jedno $i \in \{1, \dots, l\} \setminus \{j\}$ takvo da je

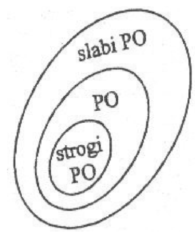
$$Q_i(\mathbf{x}) > Q_i(\mathbf{x}^*)$$

i da važi:

$$\frac{Q_j(\mathbf{x}) - Q_j(\mathbf{x}^*)}{Q_i(\mathbf{x}^*) - Q_i(\mathbf{x})} \geq \beta.$$

Strogi Pareto optimum izdvaja ona Pareto rešenja čije promene ne prouzrokuju prevelike relativne promene u funkcijama cilja.

Odnos između opisanih optimuma je takav da svaki skup strožijih Pareto optimuma predstavlja podskup slabijih optimuma, tj. svaki Pareto optimum je istovremeno i slabi Pareto optimum, a svaki strogi Pareto optimum je i Pareto optimum. Odnos svih skupova dat je na slici.



Slika 2.2.3. Odnos između Pareto optimalnih rešenja

Primer 2.0.1 Za sledeći matematički model naći rešenje koristeći se metodima višeciljnog odlučivanja.

$$\begin{array}{ll} \max & f_1(\mathbf{x}) = 2x_1 + 3x_2 \\ \max & f_2(\mathbf{x}) = x_1 - 5x_2 \\ \max & f_3(\mathbf{x}) = -3x_1 + 4x_2 \\ \text{p.o.} & g1(\mathbf{x}): \quad 2x_1 + 0x_2 \leq 10 \\ & g_2(\mathbf{x}): \quad 0x_1 + 3x_2 \leq 9 \\ & g_3(\mathbf{x}): \quad 4x_1 + 0x_2 \leq 32 \\ & g_4(\mathbf{x}): \quad 5x_1 + 0x_2 \geq 5. \end{array}$$

Rešenje: Najpre se određuju marginalna rešenja, koja predstavljaju optimalna rešenja za pojedine kriterijume, a potom i idealne vrednosti funkcija svakog od kriterijuma.

$$\begin{array}{rcl} 2x_1 + 0x_2 \leq 10 & \Rightarrow & x1 \leq 5 \\ 0x_1 + 3x_2 \leq 9 & \Rightarrow & x2 \leq 3 \\ 4x_1 + 0x_2 \leq 32 & \Rightarrow & x1 \leq 8 \\ 5x_1 + 0x_2 \geq 5 & \Rightarrow & x1 \geq 1 \\ x_1, x_2 \geq 0 & \Rightarrow & x_1, x_2 \geq 0 \end{array}$$

U koordinatnom sistemu $(\mathbf{x}) = (x_1, x_2)$ određene su tačke (1,0), (5,0), (5,3), (1,3), koje predstavljaju marginalna rešenja za funkcije kriterijuma.

Idealna vrednost funkcije kriterijuma $f_1(\mathbf{x})$

$$\begin{array}{rll} \max & f_1(\mathbf{x}) = 2x_1 + 3x_2 & \\ \text{p.o} & g_1(\mathbf{x}): & 2x_1 + 0x_2 \leq 10 \\ & g_2(\mathbf{x}): & 0x_1 + 3x_2 \leq 9 \\ & g_3(\mathbf{x}): & 4x_1 + 0x_2 \leq 32 \\ & g_4(\mathbf{x}): & 5x_1 + 0x_2 \geq 5 \\ & x_1, x_2 \geq 0 & \end{array}$$

$$\begin{aligned} f_1(1,0) &= 2 \cdot 1 + 3 \cdot 0 = 2 \\ f_1(5,0) &= 2 \cdot 5 + 3 \cdot 0 = 10 \\ f_1(5,3) &= 2 \cdot 5 + 3 \cdot 3 = 19 \\ f_1(1,3) &= 2 \cdot 1 + 3 \cdot 3 = 11 \end{aligned}$$

Idealna vrednost funkcije kriterijuma $f_1(\mathbf{x})$ jednaka je maksimalnoj vrednosti funkcije $f_1^*(\mathbf{x}) = 19$ za marginalno rešenje $\mathbf{x}^{(1)*} = (5, 3)$.

Idealna vrednost funkcije kriterijuma $f_2(\mathbf{x})$

$$\begin{array}{rll} \max & f_2(\mathbf{x}) = x_1 - 5x_2 & \\ \text{p.o} & g_1(\mathbf{x}): & 2x_1 + 0x_2 \leq 10 \\ & g_2(\mathbf{x}): & 0x_1 + 3x_2 \leq 9 \\ & g_3(\mathbf{x}): & 4x_1 + 0x_2 \leq 32 \\ & g_4(\mathbf{x}): & 5x_1 + 0x_2 \geq 5 \\ & x_1, x_2 \geq 0 & \end{array}$$

$$\begin{aligned} f_2(1,0) &= 1 \cdot 1 - 5 \cdot 0 = 1 \\ f_2(5,0) &= 1 \cdot 5 - 5 \cdot 0 = 5 \\ f_2(5,3) &= 1 \cdot 5 - 5 \cdot 3 = -10 \\ f_2(1,3) &= 1 \cdot 1 - 5 \cdot 3 = -14 \end{aligned}$$

Idealna vrednost funkcije kriterijuma $f_2(\mathbf{x})$ jednaka je maksimalnoj vrednosti funkcije $f_2^*(x) = 5$ za marginalno rešenje $\mathbf{x}^{(2)*} = (5, 0)$.

Idealna vrednost funkcije kriterijuma $f_3(\mathbf{x})$.

$$\begin{array}{rll} \max & f_3(\mathbf{x}) = -3x_1 + 4x_2 & \\ \text{p.o} & g_1(\mathbf{x}): & 2x_1 + 0x_2 \leq 10 \\ & g_2(\mathbf{x}): & 0x_1 + 3x_2 \leq 9 \\ & g_3(\mathbf{x}): & 4x_1 + 0x_2 \leq 32 \\ & g_4(\mathbf{x}): & 5x_1 + 0x_2 \geq 5 \\ & x_1, x_2 \geq 0 & \end{array}$$

$$\begin{aligned} f_3(1,0) &= -3 \cdot 1 + 4 \cdot 0 = -3 \\ f_3(5,0) &= -3 \cdot 5 + 4 \cdot 0 = -15 \\ f_3(5,3) &= -3 \cdot 5 + 4 \cdot 3 = -3 \\ f_3(1,3) &= -3 \cdot 1 + 4 \cdot 3 = 9 \end{aligned}$$

Idealna vrednost funkcije kriterijuma $f_3(\mathbf{x})$ jednaka je maksimalnoj vrednosti funkcije $f_3^*(x) = 9$ za marginalno rešenje $\mathbf{x}^{(3)*} = (1, 3)$.

Tabelarni prikaz posledice marginalnih rešenja na funkcije kriterijuma i skup ograničenja. Prve dve kolone tabele sadrže vrednosti promenljivih, zatim sledeće tri kolone sadrže ostvarene vrednosti kriterijumskih funkcija, dok poslednje četiri kolone sadrže ostvarene vrednosti ograničenja.

x_1	x_2	f_1	f_2	f_3	$g_1 \leq 10$	$g_2 \leq 9$	$g_3 \leq 32$	$g_4 \geq 5$
5	3	**19**	-10	-3	10	9	20	25
5	0	10	**5**	-15	10	0	20	25
1	3	11	-14	**9**	2	9	4	5

2.1 Metod globalnog kriterijuma

Metod globalnog kriterijuma izuzetno je jednostavan i ne zahteva preferencije o kriterijumima. Nakon određivanja idealnih vrednosti kriterijuma formira se pomoćni jednokriterijumski model sa ograničenjima kao u modelu i funkcijom kriterijuma:

$$\min f_r(\mathbf{x}) = \sum_k \left(\frac{f_k^*(\mathbf{x}) - f_k(\mathbf{x})}{f_k^*(\mathbf{x})} \right)^r, \ r \geq 1.$$

Rešenje problema jednako je minimalnoj vrednosti jednokriterijumskog modela, a predstavlja odabranu varijantu zbira normalizovanih odstojanja ostvarenih vrednosti od idealnih vrednosti kriterijuma.

Primer 2.1.1 Razmatra se napred prikazani primer i najjednostavnija varijanta formiranja funkcije kriterijuma u pomoćnom modelu metoda globalnog kriterijuma kada je $r = 1$.

$$\begin{aligned}
\max \quad & f_1(\mathbf{x}) = 2x_1 + 3x_2 \\
\max \quad & f_2(\mathbf{x}) = x_1 - 5x_2 \\
\max \quad & f_3(\mathbf{x}) = -3x_1 + 4x_2 \\
\text{p.o.} \quad & g1(\mathbf{x}): \quad 2x_1 + 0x_2 \leq 10 \\
& g_2(\mathbf{x}): \quad 0x_1 + 3x_2 \leq 9 \\
& g_3(\mathbf{x}): \quad 4x_1 + 0x_2 \leq 32 \\
& g_4(\mathbf{x}): \quad 5x_1 + 0x_2 \geq 5.
\end{aligned}$$

Rešenje: Idealne vrednosti kriterijuma:

$$f_1^*(\mathbf{x}) = 19, \ f_2^*(\mathbf{x}) = 5, \ f_3^*(\mathbf{x}) = 9.$$

Globalna funkcija (za $r = 1$) jednaka je:

$$\begin{aligned}
\min f_{r=1}(x) &= \frac{f_1^*(\mathbf{x}) - f_1(\mathbf{x})}{f_1^*(\mathbf{x})} + \frac{f_2^*(\mathbf{x}) - f_2(\mathbf{x})}{f_2^*(\mathbf{x})} + \frac{f_3^*(\mathbf{x}) - f_3(\mathbf{x})}{f_3^*(\mathbf{x})} \\
&= \frac{19 - (2x_1 + 3x_2)}{19} + \frac{5 - (x_1 - 5x_2)}{5} + \frac{9 - (-3x_1 + 4x_2)}{9}
\end{aligned}$$

$$\min f_{r=1}(x) = 3 + 0.03x_1 0.40x_2.$$

x_1	x_2	$\min f_{r=1}(\mathbf{x}) = 3 + 0.03x_1 + 0.40x_2$
5	3	4.3
5	0	**3.1**
1	3	4.2

Odatle se dobija $\min f_{r=1}(\mathbf{x}) = 3 + 0.03x_1 0.40x_2 = 3.1$ za $\mathbf{x}^* = (5, 0)$.

2.2 Metod težinskih koeficijenata

Metod težinskih koeficijenata je najstarija metoda za VKO-u koja je korišćena. Po ovoj metodi uvode se težinski koeficijenti w_i za sve kriterijumske funkcije $Q_i(\mathbf{x})$, $i = 1, \ldots, l$, pa se problem optimizacije svodi na sledeću skalarnu optimizaciju

$$\begin{aligned} \text{Max:} \quad & Q(\mathbf{x}) = \sum_{i=1}^{l} w_i Q_i(\mathbf{x}) \\ \text{p.o.:} \quad & \mathbf{x} \in \mathbf{X}, \end{aligned} \tag{2.2.0.1}$$

gde težine w_i, $i = 1, \ldots, l$ ispunjavaju sledeće uslove:

$$\sum_{i=1}^{l} w_i = 1, \quad w_i \geq 0, \ i = 1, \ldots, l.$$

Često se koristi metod težinskih koeficijenata tako što se zadaju vrednosti ovih koeficijenata. Međutim, to uvek izaziva određene teškoće i primedbe na ovakav postupak, jer se unosi subjektivan uticaj na konačno rešenje preko zadatih vrednosti težinskih koeficijenata.

Glavna ideja u metodu težinskih koeficijenata je da se odaberu težinski koeficijenti w_i koji odgovaraju ciljnim funkcijama $Q_i(\mathbf{x})$, $i = 1, \ldots, l$. Mnogi autori su razvili sistematske pristupe u selektovanju težina, čiji se pregled može naći u [10], [11] i [25]. Jedna od poteškoća ovog metoda je da variranje težina konzistentno i neprekidno ne mora uvek da rezultuje u tačnoj i kompletnoj reprezentaciji Pareto optimalnog skupa. Ovaj nedostatak je diskutovan u [8].

Teorema 2.2.1 *Ako su svi težinski koeficijenti w_i pozitivni, onda je rešenje problema* (2.2.0.1) *Pareto optimalno rešenje polaznog problema VKO.*

Dokaz. Neka je $\mathbf{x}^*$ rešenje problema (2.2.0.1), i neka su svi težinski koeficijenti strogo pozitivni. Pretpostavimo da ono nije Pareto optimalno, tj. da postoji $\mathbf{x} \in S$ tako da za $i = 1, \ldots, l$ važi $Q_i(\mathbf{x}) \geq Q_i(\mathbf{x}^*)$, pri čemu važi bar jedna stroga nejednakost (recimo za indeks j). Kako je $w_i > 0$ za svako i, važi

$$\sum_{i=1}^{l} w_i Q_i(\mathbf{x}) > \sum_{i=1}^{l} w_i Q_i(\mathbf{x}^*)$$

pa dobijamo kontradikciju sa pretpostavkom da je $\mathbf{x}^*$ rešenje težinskog problema (2.2.0.1). Sledi da je $\mathbf{x}^*$ Pareto optimalno. $\square$

Teorema 2.2.2 *Ako je za svako $i \in \{1, \dots, l\}$ ispunjen uslov $w_i \geq 0$, onda je rešenje problema* (2.2.0.1) *slabi Pareto optimum polaznog problema VKO.*

Dokaz. Neka je $\mathbf{x}^*$ rešenje problema (2.2.0.1) i da je ispunjen uslov $w_i \geq 0$. Pretpostavimo da ono nije slabo Pareto optimalno, tj. da postoji $\mathbf{x} \in S$ tako da za $i = 1, \dots, l$ važi $Q_i(\mathbf{x}) > Q_i(\mathbf{x}^*)$. Svi koeficijenti w_i su nenegativni i bar jedan je strogo veći od nule (zbog $\sum_{i=1}^{l} w_i = 1$), pa važi

$$\sum_{i=1}^{l} w_i Q_i(\mathbf{x}) > \sum_{i=1}^{l} w_i Q_i(\mathbf{x}^*)$$

pa dobijamo kontradikciju sa pretpostavkom da je $\mathbf{x}^*$ rešenje težinskog problema. Dakle, $\mathbf{x}^*$ je slabi Pareto optimum. □

Teorema 2.2.3 *Ako je rešenje problema* (2.2.0.1) *jedinstveno, ono je onda i Pareto optimalno.*

Dokaz. Neka je $\mathbf{x}^*$ jedinstveno rešenje problema (2.2.0.1). Pretpostavimo da ono nije Pareto optimalno rešenje problema VKO, tj. da postoji $\mathbf{x} \in S$ tako da za $i = 1, \dots, l$ važi $Q_i(x) \geq Q_i(x^*)$, pri čemu važi bar jedna stroga nejednakost (recimo za indeks j). Primetimo da to znači $\mathbf{x} \neq \mathbf{x}^*$.

Kako je $w_i \geq 0$ za svako i, važi

$$\sum_{i=1}^{l} w_i Q_i(\mathbf{x}) \geq \sum_{i=1}^{l} w_i Q_i(\mathbf{x}^*)$$

Ako bi važila stroga nejednakost, onda $\mathbf{x}^*$ ne bi bilo rešenje problema (2.2.0.1). Dakle, važi jednakost. To znači da postoje dva različita rešenja $\mathbf{x}$ i $\mathbf{x}^*$ problema (2.2.0.1), što je kontradikcija. □

Pokazaćemo sada jedno jače tvrđenje.

Teorema 2.2.4 *Ako su svi $w_i > 0$, $i \in \{1, \dots, l\}$, tada je rešenje problema* (2.2.0.1) *strogi Pareto optimum problema VKO.*

Dokaz. Neka je $\mathbf{x}^*$ rešenje težinskog problema. Pokazali smo da je ono Pareto optimalno. Dokažimo da je to rešenje takođe i strogi Pareto optimum sa konstantom

$$M = (k-1) \max_{i,j} \frac{w_j}{w_i}.$$

Pretpostavimo suprotno: da postoje $\mathbf{x} \in S$ i indeks i takvi da je $Q_i(\mathbf{x}) > Q_i(\mathbf{x}^*)$ pri čemu za svako j za koje je $Q_j(\mathbf{x}^*) > Q_j(\mathbf{x})$ važi $Q_i(\mathbf{x}^*) - Q_i(\mathbf{x}) < M(Q_j(\mathbf{x}) - Q_j(\mathbf{x}^*))$. Zamenom

$$M = \frac{(k-1) w_j}{w_i}$$

dobijamo

$$w_i \frac{Q_i(\mathbf{x}^*) - Q_i(\mathbf{x})}{k-1} < w_j\left(Q_j(\mathbf{x}) - Q_j(\mathbf{x}^*)\right) > 0 \ .$$

Dakle, za svako $j \neq i$ za koje je $Q_j(\mathbf{x}^*) > Q_j(\mathbf{x})$ važi prethodna nejednakost. Za one indekse $j \neq i$ za koje je $Q_j(\mathbf{x}^*) \leq Q_j(\mathbf{x})$ gornja nejednakost svakako važi. Dakle, za svako $j \neq i$ važi

$$w_i \frac{Q_i(\mathbf{x}^*) - Q_i(\mathbf{x})}{k-1} < w_j\left(Q_j(\mathbf{x}) - Q_j(\mathbf{x}^*)\right),$$

pa sabiranjem ovih nejednakosti za $j = 1, \ldots, i-1,\ i+1, \ldots, l$ dobijamo

$$w_i\left(Q_i(\mathbf{x}^*) - Q_i(\mathbf{x})\right) < \sum_{j=1, j\neq i}^{l} w_j\left(Q_j(\mathbf{x}) - Q_j(\mathbf{x}^*)\right)$$

tj.

$$\sum_{j=1}^{l} w_j Q_j(\mathbf{x}^*) < \sum_{j=1}^{l} w_j Q_j(\mathbf{x}) \ .$$

Dobijamo da $\mathbf{x}^*$ nije rešenje težinskog problema tj. dolazimo do kontradikcije. Zaključujemo da je $\mathbf{x}^*$ zaista strogi Pareto optimum polaznog problema. □

Težinski koeficijent na neki način treba da predstavi značaj kriterijumske funkcije kojoj je dodeljen. Da bismo to postigli najpre moramo da normalizujemo kriterijumske funkcije tj. da ih promenimo tako da imaju približno jednake vrednosti, a pri tome da zadrže sve bitne osobine. Na primer, ako je kriterijumska funkcija linearna $Q_j(\mathbf{x}) = \sum_{i=1}^{n} a_i x_i$, tada će normalizovan oblik ove funkcije biti

$$\frac{Q_j(\mathbf{x})}{\sum_{i=1}^{n} a_i} \ .$$

Ako je kriterijumska funkcija ograničena tada se za njen normalizovan oblik može uzeti sledeća funkcija

$$\overline{Q_i}(\mathbf{x}) = \frac{Q_i(\mathbf{x}) - Q_i^*}{\max_{\mathbf{x}\in S} Q_i(\mathbf{x}) - Q_i^*} \ ,$$

čiji je kodomen $[0, 1]$.

Ukoliko donosilac odluke sam definiše težinske koeficijente, onda ovaj metod spada u grupu a priori metoda. Međutim, najčešće se malo zna o tome kako koeficijente treba izabrati. Zato se obično težinski problem rešava za razne vrednosti vektora $(w_1, \ldots, w_l)$ i na taj način dobijaju različita rešenja među kojima DO bira ono koje mu najviše odgovara. Ako se koristi ovakav pristup, onda ova metoda postaje a posteriori metoda.

Glavna mana ove metode je teškoća određivanja težinskih koeficijenata kada nemamo dovoljno informacija o problemu.

Sledi opis implementacije metoda težinskih koeficijenata.

Funkcijom `Compositions[n,l]` možemo odrediti "l-dimenzionalne tačke", čiji zbir koordinata daje n. Kada tu listu podelimo sa brojem n, možemo dobiti tačke u intervalu $[0,1]$ čije koordinate mogu predstavljati koeficijente w_i. Na taj način smo obezbedili automatsko generisanje koeficijenata w_i potrebnih za realizaciju metoda. Ostavljena je i mogućnost da korisnik sam izabere koeficijente w_i. To se postiže tako što pri pozivu funkcije `MultiW`, kojom se implementira metoda težinskih koeficijenata, poslednji parametar zadamo kao nepraznu listu, tj. zadamo koeficijente w_i u obliku liste.

Sledi program kojim je implementiran metod težinskih koeficijenata [22].

Ulazne veličine:

q_,var_List - ciljna funkcija i lista njenih parametara;

constr_List - lista ograničenja;

*var_* - korak podele intervala [0,1];

w1_List - lista težinskih koeficijenata u intervalu [0,1]. Prazna lista kao vrednost parametra $w1$ označava da će težinski koeficijentu biti generisani pomoću funkcije *Compositions*. Inače, pretpostavlja se da je svaki element $w1[[i]], 1 \leq 1 \leq Length[w1]$ liste $w1$ jedan mogući skup koeficijenata w_j, $j = 1, \ldots, l$: $w[[j]] = w1[[i,j]], j = 1, \ldots, l$.

Lokalne promenljive:

fun - formirana funkcija za jednodimenzionalnu optimizaciju.

```
MultiW[q_, constr_List, var_List, w1_List] :=
 Module[{i=0,k,l=Length[q],res={},W={},fun,sk={},qres={},mxs={},m,ls={}},
   If[w1=={},
    k=Input["Initial sum of weighting coefficients?"]; W=Compositions[k,l]/k,
    W=w1;
   ];
   Print["Weighting Coefficients: "]; Print[W];
   Print["Single-objective problems: "];
   k = Length[W];
   For[i=1, i<=k, i++,
       fun = Simplify[Sum[W[[i,j]]*q[[j]], {j,l}]];  (* 3V *)
       AppendTo[res, Maximize[fun, constr, var]];    (* 1V *)
   ];
   Print["Solutions of single-objective problems: "];  Print[res];
   For[i=1, i<=k, i++,
      AppendTo[qres, q/.res[[i, 2]] ];     AppendTo[mxs, res[[i, 1]] ];
   ];
   Print["Choose the best solution: "];
   m=Max[mxs];
   For[i=1, i<=Length[mxs], i++,
      If[m==mxs[[i]],  AppendTo[ls, {qres[[i]], res[[i,2]]}]; ]
   ];
   Return[ls];
 ]
```

Za pozitivne težine i konveksni problem, optimalna rešenja jednokriterijumskog problem jesu Pareto optimalna [30] tj., minimiziranje odgovarajućeg jednokriterijumskog problema je dovoljno za Pareto optimalnost. Međutim, formulacija ne obezbeđuje neophodan uslov za Pareto optimalnost [31]. Kada je višekriterijumski problem konveksan, primena funkcije `W=Compositions[k,l]` produkuje b Pareto optimalnih rešenja, gde integer b ispunjava $1 \leq b \leq \binom{k+l-1}{l-1}$.

Primer 2.2.1 Maksimizirati funkcije koje se nalaze u listi:

$$Q = \{Q_1(x,y) = x + y, \ \ Q_2(x,y) = 2x - y\}$$

prema ograničenjima

$$-3x + 5y \leq 9, \ 3x + 2y \quad \leq \quad 12, \ 5x - 4y \leq 9, \ x \geq 0, \ y \geq 0.$$

Prvo se koristi prazna lista za parametar $w1$, i na taj način se koeficijenti w_i generišu pomoću funkcije *Compositions*:

In[1]:=`MultiW[{x+y,2x-y},{-3x+5y<=9,3x+2y<=12,5x-4y<=9,x>=0,y>=0},{x,y},{}]`

U slučaju $k = 5$, izraz `w=Compositions[k,l]/k` produkuje

w=$\{\{0,1\},\{\frac{1}{5},\frac{4}{5}\}, \{\frac{2}{5},\frac{3}{5}\},\{\frac{3}{5},\frac{2}{5}\}, \{\frac{4}{5},\frac{1}{5}\}, \{1,0\}\}$

Takođe, jednokriterijumski problemi optimizacije dati su sledećim unutrašnjim reprezentacijama.

Za $\{w_1, w_2\}$=$\{0,1\}$:
$\{2x\text{-}y, \{-3x + 5y <= 9, 3x + 2y <= 12, 5x - 4y <= 9, x >= 0, y >= 0\}, \{x, y\}\}$

Za $\{w_1, w_2\}$=$\{\frac{1}{5},\frac{4}{5}\}$:
$\{\frac{3}{5}(3x - y), \{-3x + 5y <= 9, 3x + 2y <= 12, 5x - 4y <= 9, x >= 0, y >= 0\}, \{x, y\}\}$

Za $\{w_1, w_2\}$=$\{\frac{2}{5},\frac{3}{5}\}$:
$\{\frac{1}{5}(8x - y), \{-3x + 5y <= 9, 3x + 2y <= 12, 5x - 4y <= 9, x >= 0, y >= 0\}, \{x, y\}\}$

Za $\{w_1, w_2\}$=$\{\frac{3}{5},\frac{2}{5}\}$:
$\{\frac{1}{5}(7x + y), \{-3x + 5y <= 9, 3x + 2y <= 12, 5x - 4y <= 9, x >= 0, y >= 0\}, \{x, y\}\}$

Za $\{w_1, w_2\}$=$\{\frac{4}{5},\frac{1}{5}\}$:
$\{\frac{3}{5}(2x + y), \{-3x + 5y <= 9, 3x + 2y <= 12, 5x - 4y <= 9, x >= 0, y >= 0\}, \{x, y\}\}$

Za $\{w_1, w_2\}$=$\{1,0\}$:
$\{x + y, \{-3x + 5y <= 9, 3x + 2y <= 12, 5x - 4y <= 9, x >= 0, y >= 0\}, \{x, y\}\}$

Elementi liste *res*, koji odgovaraju rešenjima jednokriterijumskih problema, jednaki su
$\{\{\frac{9}{2},\{x{-}{>}\ 3, y{-}{>}\ \frac{3}{2}\}\},\{\{\frac{9}{2},\{x{-}{>}\ 3, y{-}{>}\ \frac{3}{2}\}\},\{\{\frac{9}{2},\{x{-}{>}\ 3, y{-}{>}\ \frac{3}{2}\}\},$
$\{\{\frac{9}{2},\{x{-}{>}\ 3, y{-}{>}\ \frac{3}{2}\}\},\{\{\frac{9}{2},\{x{-}{>}\ 3, y{-}{>}\ \frac{3}{2}\}\},\{\{5,\{x{-}{>}\ 2, y{-}{>}\ 3\}\}\}$
a rezultat je lista koja sadrži vrednosti ciljnih funkcija u tački $\{x{-}{>}\ 2, y{-}{>}\ 3\}$, koja odgovara maksimalnoj vrednosti težinske funkcije (koja je jednaka 5):

Out[1]=$\{\{\{5,1\}, \ \{x{-}{>}\ 2, y{-}{>}\ 3\}\}$

Sada se vrednosti koeficijenata w_i definišu od strane korisnika. Vrednosti težinskih koeficijenata sadržane su u svakom elementu liste
$\{\{1,0\},\{0.9,0.1\},\{0.875,0.125\},\{0.8,0.2\},\{0,1\}\}$.

In[2]:=`MultiW[{x+y,2x-y},{-3x+5y<=12,5x-4y<=9,x>=0,y>=0},{x,y},`
`{{1,0},{0.9,0.1},{0.875,0.125},{0.8,0.2},{0,1}}]`

Sledeći problemi se rešavaju u ciklusu:

Za $\{w_1, w_2\}$=$\{1, 0\}$:
$\{x + y,\{-3x + 5y <= 9,3x + 2y <= 12,5x - 4y <= 9,x >= 0,y >= 0\},\{x,y\}\}$
Za $\{w_1, w_2\}$=$\{0.9, 0.1\}$:
$\{1.1x + 0.8y,\{-3x + 5y <= 9,3x + 2y <= 12,5x - 4y <= 9,x >= 0,y >= 0\},\{x,y\}\}$
Za $\{w_1, w_2\}$=$\{0.875, 0.125\}$:
$\{1.125x + 0.75y,\{-3x + 5y <= 9,3x + 2y <= 12,5x - 4y <= 9,x >= 0,y >= 0\},\{x,y\}\}$
Za $\{w_1, w_2\}$=$\{0.8, 0.2\}$:
$\{1.2x + 0.6y,\{-3x + 5y <= 9,3x + 2y <= 12,5x - 4y <= 9,x >= 0,y >= 0\},\{x,y\}\}$
Za $\{w_1, w_2\}$=$\{0, 1\}$:
$\{2x - y,\{-3x + 5y <= 9,3x + 2y <= 12,5x - 4y <= 9,x >= 0,y >= 0\},\{x,y\}\}$

Dobija se sledeća vrednost za listu *res* koja odgovara rešenjima ovih problema:

$\{\{5,\{x->2,y->3\}\},\{4.6,\{x->2.,y->3.\}\},\{4.5,\{x->3.,y->1.5\}\},$
$\{4.5,\{x->3.,y->1.5\}\},\{\frac{9}{2},\{x->3,y->\frac{3}{2}\}\}\}$

a rezultat je

Out[2]= $\{\{\{5,1\},\{x->2,y->3\}\}\}$

2.3 Metod sa funkcijom korisnosti

Za primenu ovog metoda neophodno je znati preferencije za kriterijume koje se unapred, apriori, uključuju u model. Preferencije se određuju na osnovu analize značajnosti kriterijuma za svaki konkretan slučaj, i to je najvažnija, kardinalna informacija o problemu. Najpre se formira pomoćni jednokriterijumski model u vidu funkcije korisnosti $U(f)$:

$$\max U(f) = U\{f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_l(\mathbf{x})\}$$

koja se rešava uz primenu postojećih ograničenja (p.o.):

$$g_i(\mathbf{x}) \leq 0, \; i \in I$$
$$x_j \geq 0, zasvakoj \in J.$$

Potom se nađeno optimalno rešenje uvodi u svaki od kriterijuma i određuju njihove konkretne vrednosti. Funkcija korisnosti zavisi od prirode rešavanog problema. Najčešće se koriste sledeće separabilne funkcije od zadatih funkcija modela višeciljnog odlučivanja:

$$\begin{aligned} U(f) &= \sum_k f_k(\mathbf{x}) \\ U(f) &= \prod_k f_k(\mathbf{x}) \\ U(f) &= \sum_k t_k f_k(\mathbf{x}). \end{aligned}$$

Definicija 2.3.1 Funkcija $U : \mathbf{R}^k \to \mathbf{R}$ kojoj je domen kriterijumski prostor i koja odražava težnje donosioca odluke zove se vrednosna funkcija ili funkcija korisnosti.

Boljoj odluci odgovara veća vrednost vrednosne funkcije. Iz tog razloga, vrednosna funkcija bi trebalo da bude strogo opadajuća po svim promenljivama. Dakle, problem se svodi na problem jednokriterijumske optimizacije:

$$\max \ U(f(\mathbf{x})) \ p.o. \ \mathbf{x} \in S. \tag{2.3.0.1}$$

Definicija 2.3.2 *Funkcija $f : \mathbf{R}^k \to \mathbf{R}$ je strogo rastuća (opadajuća) ako za proizvoljne $\mathbf{x} = (x_1, \ldots, x_k)$ i $\mathbf{y} = (y_1, \ldots, y_k)$ iz $\mathbf{R}^k$ važi, redom*

$$(x_i < y_i \text{ za svako } i = 1, \ldots, k) \Rightarrow f(\mathbf{x}) < f(\mathbf{y}).$$

$$(x_i \leq y_i \text{ za svako } i = 1, \ldots, k) \Rightarrow f(\mathbf{x}) > f(\mathbf{y})).$$

Definicija 2.3.3 *Funkcija $f : \mathbf{R}^k \to \mathbf{R}$ je jako rastuća (opadajuća) ako za proizvoljne $\mathbf{x} = (x_1, \ldots, x_k)$ i $\mathbf{y} = (y_1, \ldots, y_k)$ iz $\mathbf{R}^k$ važi, redom*

$$(x_i \leq y_i \text{ za svako } i = 1, \ldots, k \text{ i } x_j < y_j \text{ za neko } j) \Rightarrow f(\mathbf{x}) < f(\mathbf{y}).$$

$$(x_i \leq y_i \text{ za svako } i = 1, \ldots, k \text{ i } x_j < y_j \text{ za neko } j) \Rightarrow f(\mathbf{x}) > f(\mathbf{y}).$$

Teorema 2.3.1 *Za jako opadajuću funkciju U rešenje problema* (2.3.0.1) *je Pareto optimalno rešenje početnog višekriterijumskog problema.*

Dokaz. Pretpostavimo suprotno, tj. da neko rešenje problema (2.3.0.1) (označimo ga sa $\mathbf{x}^*$) nije Pareto optimalno. To znači da postoji $x \in S$ za koje važi da je $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$, pri čemu postoji indeks j za koji važi stroga nejednakost. Kako je funkcija U jako opadajuća, sledi $U(f(\mathbf{x})) > U(f(\mathbf{x}^*))$. Dobijamo kontradikciju sa pretpostavkom da se maksimum funkcije $U \circ f$ dostiže u tački $\mathbf{x}^*$. $\square$

Metod funkcije korisnosti je jednostavan i jako dobar u slučajevima kada ga je moguće primeniti. Međutim, u praksi su česti slučajevi kada je neprekidnom stabilnom funkcijom jako teško ili nemoguće izraziti želje donosioca odluke. Osim toga može se desiti da donosilac odluke nije u stanju da strogo matematički izrazi svoje težnje ili da ih delimično promeni nakon što bude upoznat sa rezultatima.

U većini a posteriori metoda leži funkcija kriterijuma sa ciljem da ona bude sto tačnije izražena.

Primer 2.3.1 Rešava se problem

$$\begin{array}{ll}
\max & f_1(\mathbf{x}) = 2x_1 + 3x_2 \\
\max & f_2(\mathbf{x}) = x_1 - 5x_2 \\
\max & f_3(\mathbf{x}) = -3x_1 + 4x_2 \\
\text{p.o.} & g1(\mathbf{x}) : \quad 2x_1 + 0x_2 \leq 10 \\
 & g_2(\mathbf{x}) : \quad 0x_1 + 3x_2 \leq 9 \\
 & g_3(\mathbf{x}) : \quad 4x_1 + 0x_2 \leq 32 \\
 & g_4(\mathbf{x}) : \quad 5x_1 + 0x_2 \geq 5.
\end{array}$$

sa sledećim težinama za kriterijume: $t_1 = 0.6, t_2 = 0.3$ i $t_3 = 0.1$.

Uvođenjem vrednosti težinskih koeficijenata u funkciju korisnosti dobija se:

$$\max U(f) = t_1 \cdot f_1(\mathbf{x}) + t_2 \cdot f_2(x) + t_3 \cdot f_3(x)$$
$$\max U(f) = 0.6(2x_1 + 3x_2) + 0.3(x_1 - 5x_2) + 0.1(-3x_1 + 4x_2),$$

ili posle sređivanja

$$maxU(f) = 0.9x_1 0.7x_2.$$

Maksimum se određuje iz sledeće tabele

x_1	x_2	$\max U(f) = 0.9x_1 0.7x_2$	f_1	f_2	f_3
5	3	2.4			
5	0	**4.5**	5	10	-15
1	3	-1.2			

Rešenje: $\max f(\mathbf{x}) = 0.9x_1 0.7x_2 = 4.5$ za $\mathbf{x}* = (5, 0)$.

2.4 Metod ograničavanja kriterijuma

Metod ograničavanja kriterijuma jedan je od najstarijih metoda rešavanja modela višeciljnog odlučivanja, čiji se elementi direktno ili indirektno uključuju u procedure drugih metoda. Po ovom metodu vrši se optimizacija najznačajnijeg kriterijuma, neka je to s-ti kriterijum, dok se ostali prevode u ograničenja sa zahtevima da se ostvare željene vrednosti tih kriterijuma. Kao i kod metoda sa funkcijom korisnosti, i u ovom metodu se preferencije o kriterijumima zadaju unapred, apriori, i to je najvažnija, kardinalna informacija. Prvo se definiše jednokriterijumski pomoćni model koji odgovara polaznom modelu višeciljnog odlučivanja:

$$\begin{array}{ll} \max & f_s(\mathbf{x}) \\ \text{p.o.} & g_i(\mathbf{x}) \leq 0,\ i = 1, 2, ..., l \\ & f_k(\mathbf{x}) \left\{ \begin{array}{l} \geq L_k \\ \leq H_k \end{array} \right. , k \neq s, k = 1, 2, ..., l \\ & x_j \geq 0,\ j = 1, 2, ..., n. \end{array}$$

gde veličine L_k i H_k predstavljaju donju i gornju granicu, respektivno, za k-te kriterijume prevedene u ograničenja, $k \neq s$, $k = 1, 2, ..., l$. Određivanjem optimalnog rešenja modela dobija se uslovljena optimalna vrednost s-tog kriterijuma, koja se uvodi u ostale kriterijume radi proračuna njihovih vrednosti. U tu svrhu mogu se koristiti odgovarajuće granice i izravnavajuće promenljive pripradajućih ograničenja. Za razliku od prethodnih metoda višeciljnog odlučivanja koji zahtevaju određivanje maksimalnih vrednosti, ovaj metod se sprovodi uvođenjem samo donjih granica za k-te kriterijume u ograničenja. Pri tome se mora imati na umu koje su moguće vrednosti tih kriterijuma, što se određuje na osnovu analize tabele sa idealnim vrednostima kriterijuma i posledicama marginalnih rešenja na kriterijume.

Primer 2.4.1 Neka se u napred razmatranom primeru smatra da je prvi kriterijum ($s = 1$) od najvećeg značaja. Potrebno je odrediti maksimalnu vrednost tog kriterijuma sa zahtevom da se ostalim kriterijumima ostvare najmanje 60% idealnih vrednosti.

$$0.60 f_2^*(\mathbf{x}) = 0.60 \cdot 5 = 3.0$$
$$0.60 f_3^*(\mathbf{x}) = 0.60 \cdot 9 = 5.4$$

x_1	x_2 f_1	$g_1 \leq 10$	$g_2 \leq 9$	$g_3 \leq 32$	$g_4 \geq 5$	$g_5 \geq 3$	$g_6 \geq 5.4$	
5	3	**19**	10	9	20	25	-10	-3
5	0	10	10	0	20	25	5	-15
1	3	11	2	9	4	5	-14	9

Optimalno Rešenje je $f_1(\mathbf{x}) = 19$ za $\mathbf{x}^{(1)*} = (5, 3)$.

2.5 Leksikografska višekriterijumska optimizacija

Često se do optimalnog rešenja dolazi posle uzastopnog donošenja odluka. Prvo se nađe optimalno rešenje za najvažniju funkciju cilja. Ako je optimalno rešenje jedinstveno, tada je problem rešen. Međutim, ako optimalno rešenje nije jedinstveno, tada se na skupu svih optimalnih rešenja optimizuje funkcija cilja koja je druga po važnosti. Ako je optimalno rešenje sada jedinstveno, problem je rešen; ako nije, optimizuje se funkcija cilja treća po važnosti na skupu optimalnih rešenja prve i druge funkcije cilja itd.

Dakle posmatrajmo problem minimizacije date uređene sekvence ciljnih funkcija:

$$Q_1(x), \ldots, Q_l(x)$$

i skup ograničenja

$$\mathbf{x} \in \mathbf{X}.$$

Trebalo bi da se reši sledeći skup konveksnih uslovnih nelinearnih programa

$$\begin{aligned} \text{Max:} \quad & Q_i(\mathbf{x}) \\ \text{p.o.:} \quad & \mathbf{x} \in \mathbf{X} \\ & Q_j(\mathbf{x}) \geq a_j, \quad j = 1, \ldots, i-1,\ i \geq 2, \end{aligned}$$

gde su $a_j = Q_j(\mathbf{x_j}^*)$, $j = 1, \ldots, i-1$ optimalne vrednosti prethdono postavljenih jednokriterijumskih problema na nivoima prioriteta $j = 1, \ldots, i-1$, $i \geq 2$.

Nejednakosti u poslednjem problemu mogu biti zamenjene jednakostima [21]. leksikografski metod spada u grupu apriornih metoda.

Pretpostavimo da su kriterijumske funkcije $Q_1, \ldots, Q_l$ poređane od najvažnije do najmanje važne. Polazni problem višekriterijumske optimizacije se rešava sledećim algoritmom:

1. $S_0 := S,\ i := 1$

2. Rešava se problem:

$$\text{Minimizirati } Q_i(\mathbf{x}) \text{ pod uslovom } \mathbf{x} \in S_{i-1}$$

   Neka je rešenje ovog problema $\mathbf{x}^{(i)^*}$.

3. Ako je $\mathbf{x}^{(i)^*}$, dobijeno u koraku 2, jedinstveno rešenje, ono se proglašava za rešenje višekriterijumskog problema i algoritam se završava.

4. Formira se skup $S_i := \{\mathbf{x} \in S_{i-1} \mid Q_i(\mathbf{x}) \geq Q_i(\mathbf{x}^{(i)^*})\}$

5. Ako je $i = l$ skup S_l se proglašava za skup rešenje problema višekriterijumske optimizacije, i algoritam se završava.

6. Stavlja se $i := i + 1$ i vraćamo se na korak 2.

Jedan od načina za implementaciju višekriterijumske optimizacije dat je funkcijom *MultiLex* [22].

```
MultiLex[q_List,constr_List,var_List]:=
  Module[{res={},s,f=constr,l=Length[q] ,i},
     For[i=1,i<=l,i++,
         s=Maximize[q[[i]],f,var];       (* 2V *)
         If[i<l, AppendTo[f,q[[i]]>=First[s]]  ];   (* 3V,2V *)
         AppendTo[res,{q/.Last[s],Last[s]}];
     ];
     Print[res]; Return[{q/.Last[s],Last[s]}];
  ]
```

Primer 2.5.1 Primer. Problem maksimizacije funkcija koje su sadržane u listi
$\{8x1 + 12x2, 14x1 + 10x2, x1 + x2\}$
prema ograničenjima
$\{8x1 + 4x2 <= 600, 2x1 + 3x2 <= 300, 4x1 + 3x2 <= 360, 5x1 + 10x2 >= 600,$
$x1 >= 0, x2 >= 0\}$
može biti rešen koristeći izraz

```
In[3]:=MultiLex[{8x1+12x2,14x1+10x2,x1+x2},
   {8x1+4x2<=600,2x1+3x2<=300,4x1+3x2<=360,5x1+10x2>=600,x1>=0,x2>=0},
   {x1,x2}]
```

Unutrašnje reprezentacije jednokriterijumskih problema su:

za $i = 1$:

$8x1 + 12x2,$
$\{8x1 + 4x2 <= 600,\ 2x1 + 3x2 <= 300,\ 4x1 + 3x2 <= 360,\ 5x1 + 10x2 >= 600,$
$x1 >= 0, x2 >= 0\},$
$\{x1, x2\}$

za $i = 2$:

$14x1 + 10x2,$
$\{8x1 + 4x2 <= 600,\ 2x1 + 3x2 <= 300,\ 4x1 + 3x2 <= 360,\ 5x1 + 10x2 >= 600,$
$x1 >= 0,\ x2 >= 0,\ 8x1 + 12x2 >= 1200\},$
$\{x1, x2\}$

za $i = 3$:

$x1 + x2$,
$\{8x1 + 4x2 <= 600,\ 2x1 + 3x2 <= 300,\ 4x1 + 3x2 <= 360,\ 5x1 + 10x2 >= 600,$
$x1 >= 0,\ x2 >= 0,\ 8x1 + 12x2 >= 1200,\ 14x1 + 10x2 >= 1220\}$,
$\{x1, x2\}$

Rešenja uzastopnih optimizacionih problema su zapamćena u listi res, koja je jednaka

```
{{1200,{x1->0,x2->100}},{1220,{x1->30,x2->80}},{110,{x1->30,x2->80}}}
```

Teorema 2.5.1 *Rešenje dobijeno leksikografskom metodom je Pareto optimalno rešenje problema višekriterijumske optimizacije.*

Dokaz. Označimo sa $\mathbf{x}^*$ rešenje dobijeno leksikografskim metodom. Pretpostavimo da ono nije Pareto optimalno tj. da postoji neko $x \in S$ tako da je $Q_i(x) \geq Q_i(x^*)$ za svako $i = 1, \ldots, l$ pri čemu za neko k vazi $Q_k(\mathbf{x}) > Q_k(\mathbf{x}^*)$. Mogu nastati dva slučaja. Prva mogućnost je da je $\mathbf{x}^*$ jedinstveno rešenje i da su iskorišćeni svi kriterijumi do j-tog (to znači da je na kraju prethodno opisanog algoritma $i = j$). Kako je $\mathbf{x} \in S = S_0$ i $Q_1(\mathbf{x}) \geq Q_1(x^*)$ sledi da je $Q_1(\mathbf{x}) = Q_1(\mathbf{x}^*)$, kao i $x \in S_1$. Kako je sada $x \in S_1$ i $Q_2(x) \geq Q_2(x^*)$ sledi $Q_2(x) = Q_2(x^*)$ i $\mathbf{x} \in S_2$. Nastavljajući ovakvo rezonovanje zaključujemo $Q_{j-1}(\mathbf{x}) = Q_{j-1}(\mathbf{x}^*)$ i $\mathbf{x} \in S_{j-1}$. Kako je $\mathbf{x}^*$ jedinstveno rešenje problema

$$\text{Minimizirati } Q_j(\mathbf{x}) \text{ pod uslovom } \mathbf{x} \in S_{j-1}$$

a pri tome važi $Q_j(\mathbf{x}) \geq Q_j(\mathbf{x}^*)$ i $\mathbf{x} \in S_{j-1}$ zaključujemo da mora biti $\mathbf{x} = \mathbf{x}^*$ pa je $Q_k(\mathbf{x}) = Q_k(\mathbf{x}^*)$ odakle dobijamo kontradikciju sa pretpostavkom $Q_k(\mathbf{x}) > Q_k(\mathbf{x}^*)$.

Druga mogućnost je da su iskorišćeni svi kriterijumi (tj. u prethodnom algoritmu je $i = l$). Potpuno analogno kao u prvom slučaju se pokazuje da $\mathbf{x} \in S_k$ i da za svako $i = 1, \ldots, k$ važi $Q_i(\mathbf{x}) = Q_i(\mathbf{x}^*)$, što je kontradikcija sa $Q_k(\mathbf{x}) > Q_k(\mathbf{x}^*)$. $\square$

Iz ovog dokaza se može zaključiti da se ograničenje oblika $Q_j(x) \leq Q_j(\mathbf{x}^{(i)*})$ može zameniti ograničenjem $Q_j(\mathbf{x}) = Q_j(\mathbf{x}^{(i)*})$.

Mane leksikografske metode su očigledne:

- U praksi je često teško odrediti koji je kriterijum važniji od drugog.
- Najčešće se dešava da se jedinstveno rešenje dobije pre nego što se iskoriste svi kriterijumi, ili čak nakon korišćenja samo jednog kriterijuma. Na taj način, neki kriterijumi uopšte ne učestvuju u donošenju odluke, što je krajnje nepoželjno.

Zbog toga klasična leksikografski medod ima jako ograničenu primenu.

2.6 Relaksirani leksikografski metod

Hijerarhijski metod je modifikacija leksikografskog metoda, u kome se koriste ograničenja oblika [20]:

$$Q_j(\mathbf{x}) \geq \left(1 + \frac{\delta_j}{100}\right) Q_j(\mathbf{x_j}^*).$$

Relaksacija se sastoji u povećanju desne strane ograničenja za procenat $Q_j(\mathbf{x_j}^*) \cdot \delta_j$. Variranjem parametra δ_j mogu se generisati različite Pareto optimalne tačke [17]. Još jedna varijacija leksikografskog metoda je uvedena u [27], gde su ograničenja jednaka

$$Q_j(\mathbf{x}) \geq Q_j(\mathbf{x_j}^*) - \delta_j.$$

U ovom slučaju, $\delta_j \leq 100$ jesu pozitivne tolerancije definisane od strane DO.

Relaksirani leksikografski metod je iterativni postupak u kome se rešavaju jednokriterijumski zadaci optimizacije. Pretpostavlja se da su od strane DO dati prioriteti kriterijuma i da su u skladu sa njima dodeljeni indeksi kriterijumima. U relaksiranoj leksikografskoj metodi se po svakom od p kriterijuma rešava jednokriterijumski zadatak optimizacije. Pri tome se u narednoj iteracijine postavlja kao ograničenje zahtev da rešenje bude optimalno po kriterijumu višeg prioriteta, več se ono relaksira tako da se zahteva da rešenje bude u okolini optimalnog rešenja dobijenog u prethodnoj iteraciji. Nataj način, svaki kriterijum utiče na konačno rešenje.

DO zadaje redosled kriterijuma po značajnosti. Pored toga, svakom kriterijumu, uzimajući poslednji, dodeljuje se vrednost δ_k , $k = 1, \ldots, l-1$, za koju kriterijum višeg prioriteta sme da odstupi od svoje optimalne vrednosti. Metoda obuhvata izvršavanje sledećih l koraka:

Korak 1. Rešiti problem

$$\begin{array}{rl} \text{Max:} & Q_1(\mathbf{x}) \\ \text{p.o.:} & \mathbf{x} \in \mathbf{X} \end{array}$$

Rešenje ovog problema označimo sa Q_1^*.

Korak 2. Rešiti problem

$$\begin{array}{rl} \text{Max:} & Q_2(\mathbf{x}) \\ \text{p.o.:} & \mathbf{x} \in \mathbf{X} \\ & Q_1(\mathbf{x}) \geq Q_1^* - \delta_1 \end{array}$$

Korak p. Za svako $3 \leq p \leq l$ rešiti jednokriterijumski problem

$$\begin{array}{rl} \text{Max:} & Q_p(\mathbf{x}) \\ \text{p.o.:} & \mathbf{x} \in \mathbf{X} \\ & Q_j(\mathbf{x}) \geq Q_j^* - \delta_j,\ j = 1, \ldots, l-1 \end{array}$$

Za rešenje polaznog modela se usvaja rezultat dobijen u poslednjem koraku, a vrednosti funkcija cilja za dobijeno rešenje se moraju posebno računati.

Rešenje dobijeno ovim metodom obezbeđuje slabi Pareto optimum, a ako je rešenje jedinstveno, ono je i Pareto optimalno. Ako je dopustiva oblast konveksna, podešavanjem parametara $\delta_k, k = 1, \ldots, l-1$, može se dobiti bilo koje Pareto optimalno rešenje.

Teorema 2.6.1 *Rešenje dobijeno relaksiranim leksikografskim metodom je slabi Pareto optimum problema višekriterijumske optimizacije.*

Dokaz. Pretpostavimo suprotno da rešenje $\mathbf{x}^*$ dobijeno relaksiranim leksikografskim metodom nije slabo Pareto optimalno. To znači da postoji neko $\mathbf{x} \in S$ tako da za svako $i = 1, \ldots, l$ važi $Q_i(\mathbf{x}) \geq Q_i(x^*)$. Kako je $Q_i(\mathbf{x}^*) = Q_i^*$ tim pre važi $Q_i(\mathbf{x}) \geq Q_i^* - \delta_i$, $i = 1, \ldots, k-1$, pa kako je $Q_k(\mathbf{x}) > Q_k(\mathbf{x}^*)$ dobijamo kontradikciju sa pretpostavkom da je $\mathbf{x}^*$ rešenje dobijeno relaksiranim leksikografskom metodom. □

Teorema 2.6.2 *Ako se relaksiranim leksikografskim metodom dobija jedinstveno rešenje, ono je Pareto optimalno.*

Dokaz. Pretpostavimo suprotno, da jedinstveno rešenje $\mathbf{x}^*$ dobijeno relaksiranim leksikografskim metodom nije Pareto optimalno. To znači da postoji neko $\mathbf{x} \in S$ tako da za svako $i = 1, \ldots, l$ važi $Q_i(\mathbf{x}) \geq Q_i(\mathbf{x}^*)$ pri čemu za neko $j \in \{1, \ldots, l\}$ važi stroga nejednakost. Razlikujemo dva slučaja:

1) $Q_k(\mathbf{x}) > Q_k(\mathbf{x}^*)$. Kako iz pretpostavljenog sledi da je $Q_i(\mathbf{x}) \geq Q_i^* - \delta_i$, $i = 1, \ldots, k-1$, zaključujemo da $\mathbf{x}^*$ nije rešenje dobijeno u koraku k gore opisanog algoritma tj. da $\mathbf{x}^*$ nije rešenje dobijeno relaksiranim leksikografskim metodom, što je kontradikcija.

2) $Q_k(\mathbf{x}) = Q_k(\mathbf{x}^*)$. Kao i pod 1), važi $Q_i(\mathbf{x}) \geq Q_i^* - \delta_i$, $i = 1, \ldots, k-1$, pa dobijamo da je $\mathbf{x}$ rešenje dobijemo relaksiranim leksikografskim metodom. Međutim, kako je $\mathbf{x}^*$ jedinstveno rešenje zaključujemo da mora biti $\mathbf{x} = \mathbf{x}^*$. Tada je i $Q_j(\mathbf{x}) = Q_j(\mathbf{x}^*)$, što je kontradikcija.

Dakle, $\mathbf{x}^*$ je zaista Pareto optimalno rešenje. □

U praktičnoj primeni leksikografskog metoda vrednosti δ_i moraju biti pažljivo odabrane. Ukoliko se ovim metodom dobije više rešenja bilo bi dobro smanjiti te vrednosti.

Primer 2.6.1 Primenom relaksiranog leksikografskog metoda rešiti sledeći zadatak VKO (funkcije cilja su relaksirane po prioritetu):

$$\begin{array}{rl} \text{Max:} & \{Q_1(\mathbf{x}), Q_2(\mathbf{x}), Q_3(\mathbf{x})\} \\ \text{p.o.:} & 2x_1 + x_2 \leq 6,\ x_1 + 3x_2 \leq 6,\ x_1 \geq 0,\ x_2 \geq 0. \end{array}$$

gde je

$$\begin{aligned} Q_1(\mathbf{x}) &= x_1 + 4x_2, \quad \delta_1 = 1 \\ Q_2(\mathbf{x}) &= x_1, \quad\quad\quad \delta_2 = 1 \\ Q_3(\mathbf{x}) &= x_1 + x_2. \end{aligned}$$

Rešenje.
Korak 1. Rešiti problem

$$\begin{array}{ll} \text{Max:} & Q_1(x) = x_1 + 4x_2 \\ \text{p.o.:} & 2x_1 + x_2 \leq 6,\ x_1 + 3x_2 \leq 6,\ x_1, x_2 \geq 0. \end{array}$$

Ovaj zadatak ima jedinstveno optimalno rešenje u tački $x^{1^*} = (0, 2)$, pri čemu je $Q_1^* = 8$.
Korak 2. Rešiti problem

$$\begin{array}{ll} \text{Max:} & Q_2(\mathbf{x}) = x_1 \\ \text{p.o.:} & 2x_1 + x_2 \leq 6,\ x_1 + 3x_2 \leq 6,\ x_1, x_2 \geq 0,\ x_1 + 4x_2 \geq 7. \end{array}$$

Dobija se rešenje: $x^{2^*} = (2.43, 1.14), Q_2^* = 2.43$
Korak 3. Rešiti problem

$$\begin{array}{ll} \text{Max:} & Q_3(\mathbf{x}) = x_1 + 4x_2 \\ \text{p.o.:} & 2x_1 + x_2 \leq 6,\ x_1 + 3x_2 \leq 6,\ x_1, x_2 \geq 0,\ x_1 + 4x_2 \geq 7, x_1 \geq 1.43. \end{array}$$

Rešenje ovog zadatka se usvaja kao konačno. To je rešenje: $x_1^* = 3.6,\ x_2^* = 1.2$. Vrednosti funkcija cilja u ovoj tački su: $Q^* = (7.2, 2.4, 3.6)$.

Relaksirani leksikografski metod je veoma osetljiv na izbor koeficijenata δ_j a u toj meri se "pogrešnim" izborom mogu dobiti neprihvatljiva rešenja. Tako u prethodnom primeru imamo slučaj da se konačno resenje poklapa sa marginalnim rešenjem funkcije cilja koja ima najniži prioritet, dok je vrednost najznačajnijeg kriterijuma smanjena. Kod primene ove metode se preporučuje da DO kritički preispita dobijena rešenja, uporedi ih sa marginalnim i da po potrebi koriguje zadate koeficijente.

Primer 2.6.2 Vlasnik hotela pre početka sezone odlučuje o opremanju soba nameštajem. Hotel raspolaže sa ukupno 58 soba, od kojih su 16 male, tako da mogu da budu samo jednokrevetne, dok ostale mogu biti jednokrevetne, dvokrevetne ili trokrevetne. Podaci o ceni opremanja soba i sezonskoj zaradi po sobi su dati u sledećoj tabeli:

	Jednokrevetna	Dvokrevetna	Trokrevetna
Cena opremanja	20000	40000	60000
Sezonska zarada	15000	25000	30000

Tabela 4.6.1.

Za opremanje soba, vlasnik može da izdvoji 2.5 miliona dinara. On želi da postigne dva cilja:
1. da ostvari što veću sezonsku zaradu hotela i
2. da omogući primanje što većeg broja gostiju kako ne bi došlo do toga da gostima bude uskraćeno gostoprimstvo.

a) Formulisati matematički model za određivanje optimalnog opremanja soba potrebnim nameštajem (broj kreveta) ako se žele ostvariti oba postavljena kriterijuma.

b) Rešiti zadatak ako je prvi kriterijum prioritetan i ako je vlasnik spreman da "žrtvuje" 50.000 din. svoje sezonske zarade da bi poboljšao drugi kriterijum.

Rešenje.

a) Promenljive x_1, x_2 i x_3 će predstavljati broj jednokrevetnih, dvokrevetnih i trokrevetnih soba u hotelu, respektivno. Tada matematički model ima sledeći oblik:

$$\begin{array}{rl} \text{Max:} & [Q_1(\mathbf{x}), Q_2(\mathbf{x})] \\ \text{p.o.:} & x_1 + x_2 + x_3 \leq 58, \\ & 20.000x_1 + 40.000x_2 + 60.000x_3 \leq 2.500.000, \\ & x_1 \geq 16,\ x_2 \geq 0,\ x_3 \geq 0. \end{array}$$

gde su:

$$Q_1(\mathbf{x}) = 15.000x_1 + 25.000x_2 + 30.000x_3$$
$$Q_2(\mathbf{x}) = x_1 + x_2 + 3x_3$$

b) Iako nije eksplicitno naglašeno, na osnovu zahteva zadatka je očigledno da se traži rešenje dobijeno relaksiranim leksikografskim metodom, s tim da je najvažniji prvi kritarijum, a zadati koeficijent $\alpha_1 = 50.000$. U prvom koraku rešavamo model:

$$\begin{array}{rl} \text{Max:} & Q_1(\mathbf{x}) = 15.000x_1 + 25.000x_2 + 30.000x_3 \\ \text{p.o.:} & \mathbf{x} \in \mathbf{X}, \end{array}$$

čije je rešenje: $Q_1^* = 1.415.000$ din, $x_1^{1^*} = 16$, $x_2^{1^*} = 17$, $x_3^{1^*} = 25$. U drugoj iteraciji rešavamo model u koji smo dodali ograničenje koji obezbeđujemo da se prvi kriterijum ne "pokvari" za više od 50.000: $(1.415.000 - 50.000 = 1.365.000)$:

$$\begin{array}{rl} \text{Max:} & Q_2(\mathbf{x}) = x_1 + 2x_2 + 3x_3 \\ \text{p.o.:} & \mathbf{x} \in \mathbf{X},\ 15.000x_1 + 25.000x_2 + 30.000x_3 \geq 1.365.000. \end{array}$$

Rešenje ovog zadatka je konačno rešenje problema. Ono, međutim, nije jedinstveno. Štaviše, simpleks metodom se mogu generisati četiri rešenja:

$$x^{*I} = (23, 0, 34),\ x^{*II} = (16, 7, 31, 67),\ x^{*III} = (24, 5, 0, 33, 5),\ x^{*IV} = (16, 17, 25).$$

Sva optimalna rešenja se mogu dobiti konveksnom kombinacijom ova četiri. Grafički posmatrano, u trodimenzionalnom prostoru se dobija da su optimalna rešenja deo ravni oivičen trapezoidom čija su temena date četiri tačke. U svim tačkama koje se nalaze unutar tog trapezoida, vrednost druge funkcije cilja iznosi 125, dok vrednost prve varira između 1.365.000 i 1.415.000. Dakle, dobili smo skup tačaka koje predstavljaju slaba Pareto rešenja. Jedino tačka x^{*IV}, koja je optimalna i po prvom kriterijumu, je Pareto rešenje.

Zaključak Da bi vlasnik hotela na što bolji način zadovoljio oba kriterijuma, najbolje je da opremi 16 jednokrevetnih, 17 dvokrevetnih i 25 trokrevetnih soba i tako ostvari zaradu od 1.415.000 din. po sezoni. Pri tome će hotel moći da prima 125 gostiju.

Jedan od načina za implementaciju relaksiranog leksikografske meto da dat je funkcijom **MultiRelax**[] [22]:

Ulazne veličine:
q - lista ciljnih funkcija;
parcf - parametri ciljnih funkcija;
constr - lista ograničenja.

```
MultiRelax[q_List, constr_List, var_List, delta_List] :=
   Module[{res={}, s, f=constr, l=Length[q], i},
```

```
    For[i=1, i<=l, i++,
       s = Maximize[q[[i]], f, var];      (* 2V *)
       If[i<l, AppendTo[f, q[[i]]>=First[s]-delta[[i]] ]  ]; (* 3V,2V *)
       AppendTo[res, s];
    ];
    Print[res]; Return[{q/.Last[s],Last[s]}];
]
```

Primer 2.6.3 Problem vlasnika hotela rešava se izrazom

```
MultiRelax[{15000x1+25000x2+30000x3,x1+2x2+3x3},
  {x1+x2+x3<=58, 20000x1+40000x2+60000x3<=2500000,x1>=16,x2>=0,x3>=0},
  {x1,x2,x3},{50000}]
```

Dobija se sledeće rešenje:

```
{{1365000,125},{x1->23,x2->0,x3->34}}
```

2.7 Metod e-ograničenja

U ovom metodu DO izdvaja kriterijum $Q_p(\mathbf{x})$, $1 \le p \le l$ koji ima najviši prioritet i njega maksimizira, dok ostale funkcije cilja ne smeju imati vrednosti manje od unapred zadatih ε_k, $k = 1, \dots, l$, $k \neq p$. Sve druge ciljne funkcije se koriste da se kreiraju dodatna ograničenja oblika $l_i \le Q_i(\mathbf{x}) \le \varepsilon_i$, $i = 1, \dots, l$, $i \neq p$ [12]. U ovoj relaciji, l_i i ε_i jesu donja i gornja granica ciljne funkcije $Q_i(\mathbf{x})$, respektivno.

Haimes u [9] je uveo *e-constraint* (ili trade-off) metod (metod e-ograničenja), u kome su sve vrednostoi l_i isključene. Sistematsko variranje vrednosti ε_i produkuje skup Pareto optimalnih rešenja [12]. Međutim, nepodesna selekcija vrednosti $\varepsilon \in \mathbf{R}^k$ može da proizvede rešenje koje ne ispunjava ograničenje. AOpšte pravilo za selektovanje ε_i je sledeće ([2]):

$$Q_p(\mathbf{x_i}^*) \le \varepsilon_i \le Q_i(\mathbf{x_i}^*),$$

gde je $\mathbf{x_i}^*$ tačka koja maksimizira ciljnu funkciju $Q_i(\mathbf{x})$.

Neophodno je da se reši sledeći problem

$$\begin{array}{rl} \text{Max:} & Q_p(\mathbf{x}) \\ \text{p.o.:} & x \in \mathbf{X}, \\ & Q_i(\mathbf{x}) \ge \varepsilon_i, \ i = 1, \dots, l, \ i \neq p, \end{array}$$

gde je $Q_p(\mathbf{x})$ odabrani kriterijum sa najvećim prioritetom i ϵ_i su odabrani realni brojevi. Ako optimalno rešenje ne postoji, neophodno je da se smanji vrednost za ε_i.

Teorema 2.7.1 *Rešenje problema ε ograničenja je slabo Pareto optimalno rešenje početnog problema VKO.*

Dokaz. Pretpostavimo da je $\mathbf{x}^*$ rešenje problema ε ograničenja, i da ono nije slabi Pareto optimum. Dakle, postoji $\mathbf{x} \in S$ tako da je $Q_j(\mathbf{x}) > Q_j(x^*) \wedge Q_i(\mathbf{x}) > Q_i(\mathbf{x}^*) \geq \varepsilon_i$ za svako $i = 1, \ldots, l$, $i \neq j$, odakle sledi da $\mathbf{x}^*$ nije rešenje problema ε ograničenja, što je kontradikcija. Zaključujemo da je rešenje problema ε ograničenja slabi Pareto optimum. $\square$

Teorema 2.7.2 *Tačka $\mathbf{x}^* \in S$ je Pareto optimalno rešenje polaznog problema VKO akko za svako $j = 1, \ldots, l$ tačka $\mathbf{x}^*$ jeste rešenje problema ε ograničenja pri čemu je $\varepsilon_i = Q_i(\mathbf{x}^*)$, $i = 1, \ldots, l$, $i \neq j$.*

Dokaz. Uslov je dovoljan jer iz njega sledi da ni za jedno $j = 1, \ldots, l$ ne postoji $\mathbf{x} \in S$ tako da je $Q_j(\mathbf{x}) > Q_j(\mathbf{x}^*) \wedge Q_i(\mathbf{x}) \geq \varepsilon_i = Q_i(\mathbf{x}^*)$, $i \neq j$, tj. sledi da je $\mathbf{x}^*$ Pareto optimum.

Pretpostavimo, sa druge strane, da je $\mathbf{x}^*$ Pareto optimum, a da za neko j ono nije rešenje problema ε ograničenja sa $\varepsilon_i = Q_i(\mathbf{x}^*)$, $i \neq j$. Dakle, postoji $\mathbf{x} \in S$ tako da važi $Q_j(\mathbf{x}) > Q_j(\mathbf{x}^*)$ i $Q_i(\mathbf{x}) \geq Q_i(\mathbf{x}^*) = \varepsilon_i$, $i \neq j$. Dolazimo u kontradikciju sa pretpostavkom da je $\mathbf{x}^*$ Pareto optimalno rešenje. Dakle, uslov je i potreban. $\square$

Primetimo da iz ove teoreme sledi da se metodom ε ograničenja može dobiti proizvoljno Pareto optimalno rešenje.

Teorema 2.7.3 *Neka je vektor ε proizvoljan. Tada, ako problem ε ograničenja ima jedinstveno rešenje onda je ono Pareto optimalno rešenje polaznog problema VKO.*

Dokaz. Neka je $\mathbf{x}^*$ jedinstveno rešenje problema ε ograničenja. Pretpostavimo da $\mathbf{x}^*$ nije Pareto optimalno tj. da postoji $\mathbf{x}'$ iz S tako da važi $Q_i(\mathbf{x}') \geq Q_i(\mathbf{x}^*)$ $i = 1, \ldots, l$, pri čemu važi stroga jednakost za $i = l$. Ako je $l = j$, tada dobijamo $Q_j(\mathbf{x}') > Q_j(\mathbf{x}^*) \wedge Q_i(\mathbf{x}') \geq Q_i(\mathbf{x}^*) \geq \varepsilon_i$, $i = 1, \ldots, l$, $i \neq j$, tj. dobijamo kontradikciju sa pretpostavkom da je $\mathbf{x}^*$ rešenje problema ε ograničenja. Ako je $l \neq j$, tada je $Q_i(\mathbf{x}') \geq Q_i(\mathbf{x}^*) \geq \varepsilon_i$, $i = 1, \ldots, l$, $i \neq j$ i $Q_j(x') \leq Q_j(x^*)$. Međutim kako je x^* jedinstveno rešenje te iz $Q_i(\mathbf{x}') \geq \varepsilon_i$, $i = 1, \ldots, l$, $i \neq j$ sledi da mora biti $Q_j(\mathbf{x}') < Q_j(\mathbf{x}^*)$. Ponovo dolazimo do kontradikcije pa zaključujemo da je $\mathbf{x}^*$ zaista Pareto optimalno rešenje polaznog problema VKO. $\square$

U slučaju da problem ε ograničenja nema rešenja trebalo bi povećati vrednosti granica ε_i. Naravno, nijedno ε_i ne sme biti manje od idealne vrednosti funkcije cilja Q_i. U slucaju da DO sam izabere prioritetnu funkciju Q_j i konstante ε_i na ovaj metod se može gledati kao na a priori metod. Međutim, on u tom slučaju ne može biti siguran da će biti zadovoljan dobijenim rešenjem. Zbog toga je pristup obično drugačiji. Postupak rešavanja problema ε ograničenja se ponavlja za razne vrednosti granica ε_i i za razne funcije Q_j kao prioritetne, sve dok DO ne bude zadovoljan dobijenim rešenjem.

Da bismo dokazali da je rešenje dobijeno metodom ograničenja Pareto optimalno moramo dokazati da je ono jedinstveno ili da je rešenje l različitih problema za svaku od funkcija Q_j, što nije nimalo jednostavno. Ovo je glavna mana ove metode.

Sledi opis implementacije prema [22].

Ulazni parametri:

$q\_$, $constr\_$, $var\_$: unutrašnja forma problema;

$p\_$: redni broj ciljne funkcije sa najvećim prioritetom.

```
MultiEps[q_List, constr_List, var_List, p_Integer] :=
 Module[{s, f = constr, i, j, k, l = Length[q], eps, lb = {}, ub = {}},
   For[i = 1, i <= l, i++, (* Find bounds in (3.4)  *)
     xi = Maximize[q[[i]], constr, var];
     AppendTo[lb,q[[p]]/.First[Rest[xi]]];
     AppendTo[ub,q[[i]]/.First[Rest[xi]]];
   ];
   For[i = 1, i <= l, i++,
      For[eps = lb[[i]], eps <= ub[[i]], eps = eps + 0.1,
         f = constr;
         For[k = 1, k <= l, k++,
            If[k != p, AppendTo[f, q[[k]]>=eps ]  ]  (* 3V *)
         ];
         s = Maximize[q[[p]], f, var];   (* 2V, 1V *)
      ]
   ]
 ]
```

Primer 2.7.1 Rešiti problem:

$$\begin{array}{ll} \text{Max:} & \{Q_1(\mathbf{x}) = x_1,\ Q_2(\mathbf{x}) = x_2\} \\ \text{p.o.:} & 0 \leq x_1 \leq 1,\ 0 \leq x_2 \leq 1. \end{array}$$

Pretpostavimo da prvi kriterijum ima najveći prioritet. Ovaj problem se rešava sledećim pozivom funkcije *MultiEps*:

In[5]:= `MultiEps[{x1,x2},{x1>=0,x2>=0,x1<=1,x2<=1},{x1,x2},{0.5},1]`

U slučaju kada drugi kriterijum uzima vrednosti veće ili jednake od 0.5 ($eps = 0.5$) neophodno je da se reši sledeći jednokriterijumski problem:

$$\begin{array}{ll} \text{Max:} & Q_1(\mathbf{x}) = x_1 \\ \text{p.o.:} & 0 \leq x_1 \leq 1,\ 0 \leq x_2 \leq 1,\ x_2 \geq 0.5. \end{array}$$

Njegova unutrašnja forma, korišćena u MATHEMATICA funkciji `Maximize` je

$x1,\ \{x1>=0, x2>=0, x1<=1, x2<=1, x2>=0.5\},\ \{x1, x2\}$

Rešenje problema je:

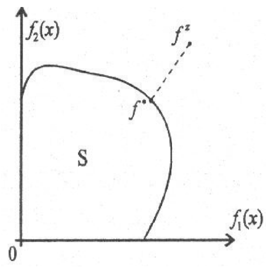
`{{1.,0.5},{x1->1., x2->0.5}}`

i nije Pareto optimalno. Pareto optimalno rešenje se generiše u slučaju $eps = 1$, i jednako je

`{{1.,1},{x1->1, x2->1}}`

2.8 Metodi rastojanja

Metodi rastojanja čine grupu za rešavanje zadataka VKO čija je osnovna ideja da se u kriterijumskom prostoru traži tačka koja je najbliža nekoj unapred određenoj tački koja se želi dostići ili ka kojoj treba težiti ako ona nije dopustiva. Drugim rečima, minimizira se rastojanje između željene tačke i dopustive oblasti. Razlike između metoda ove grupe potiču od toga kako se željena tačka određuje, na koji način se rastojanje od nje "meri", da li se uvode težinski koeficijenti, itd.



Slika 3.8.1. Tačka najbliža željenoj vrednosti f^z.

U opsťem slučaju, DO za svaki od l kriterijuma zadaje željene vrednosti ili određuje način kako će se one izračunati. Na taj način se u l -dimenzionalnom kriterijumskom prostoru definiše tačka

$$f^z = (f_1^z, ..., f_l^z)$$

koja po pravilu ne pripada dopustivoj oblasti S. U slučaju

$$f^z \in S$$

zadatak se rešava rešavanjem sistema jednačina koje su definisane skupom ograničenja.

U metodu rastojanja rešava se sledeći opšti zadatak:

$$\begin{array}{rl} \text{Min:} & d(f^z - f(x)) \\ \text{p.o:} & \mathbf{x} \in \mathbf{x} \end{array}$$

gde $d(*.*)$ označava rastojanje definisano pogodnom metrikom.

U kontekstu merenja rastojanja u kriterijumskom prostoru ovde ćemo ponoviti definicije metrika koje se najčešće koriste u metodima rastojanja:

$$l_1 \ (pravougaona) \ metrika : d_{l_1}(f^z, f(\mathbf{x})) = \sum_{k=1}^{l} |f_k^z - f_k(\mathbf{x})| \,,$$

$$l_2 \ (Euklidova) \ metrika : d_{l_2}(f^z, f(\mathbf{x})) = \sqrt{\sum_{k=1}^{l} (f_k^z - f_k(\mathbf{x}))^2},$$

$$l_\infty \ (Čebiševljeva) \ metrika : d_{l_\infty}(f^z, f(\mathbf{x})) = \max_{1 \le k \le l} \{|f_k^z - f_k(\mathbf{x})|\} \,.$$

Kriterijumima je moguće dodeliti težinske koeficijente, tako da prethodne formule za rastojanje između željenog i traženog rešenja dobijaju oblik:

$$d_{l_1}(f^z, f(x)) = \sum_{k=1}^{l} w_k \, |f_k^z - f_k(\mathbf{x})|$$

za pravougaonu metriku,

$$d_{l_2}(f^z, f(x)) = \sqrt{\sum_{k=1}^{l} w_k(f_k^z - f_k(x))^2}$$

za Euklidovu metriku,

$$d_{l_\infty}(f^z, f(\mathbf{x})) = \max_{1\leq k\leq l} \{w_k\, |f_k^z - f_k(\mathbf{x})|\}\,.$$

za Čebiševljevu metriku.

Bez obzira na oblik polaznih funcija cilja, zadaci minimizacije rastojanja su problemi nelinearnog programiranja, za koje, u opštem slučaju, nije jednostavno pronaći optimalno rešenje. Izuzetno se u slučaju l_1 metrike i linearnih funkcija cilja i ograničenja zadatak minimizacije rastojanja može formulisati kao problem LP što ćemo objasniti na sledećem primeru.

Napomena 2.8.1 *Objašnjene transformacije su osnova za formulaciju zadatka više-kriterijumskog linearnog programiranja (VLP) i oblik koji se naziva ciljno programiranje . Prema tome, zadatak ciljnog programiranja je poseban oblik zadatka metode rastojanja u VKO. Dodatno je moguće kriterijumima dodeliti prioritete dostizanja željenih vrednosti i težinske koeficijente.*

Rešenje zadatka VKO dobijeno metodom rastojanja, kada skupu vrednosti kriterijuma ne pripada željna vrednost, predstavlja slabi Pareto optimum, a ako je jedinstveno, onda je i Pareto optimalno.

Metod rastojanja (ciljno programiranje ili goal programming method) je razvijen u [4], [5] [13]. Za svaku od l kriterijumskih funkcija odabiraju se ciljne vrednosti, ili se izračunavaju pomoću funkcije *Maximize*. Na taj način, generiše se l-dimenzionalna tačka $b(\mathbf{x})=(b_1, \ldots, b_l)$, koja sadrži ciljne vrednosti za ciljne funkcije. Potom se minimizira rastojanje između tačke $b(\mathbf{x})$ i $Q(\mathbf{x})=(q_1(\mathbf{x}), \ldots, q_l(\mathbf{x}))$. Drugim rečima, rešava se sledeći opšti zadatak:

$$\begin{array}{ll} \text{Min:} & d(b - Q(\mathbf{x})) \\ \text{p.o.:} & \mathbf{x} \in \mathbf{X}, \end{array}$$

gde je $d(*.*)$ metrika koja definiše rastojanje između tačaka b i $Q(\mathbf{x})$. Uzmimo, na primer metriku definisanu L_1-normom:

$$d(b - Q(\mathbf{x})) = \sum_{i=1}^{l} |b_i - q_i(\mathbf{x})|.$$

U ovom slučaju, prirodno je da se koriste zamene $y_i = b_i - q_i(\mathbf{x})$, $i = 1, \ldots, l$. Koristeći dodatne smene $y_i = \$[i]^- - \$[i]^+$, $i = 1, \ldots, l$, u kojima je $\$[i]^-, \$[i]^+ \geq 0$, (i tada je $|y_i| = \$[i]^+ + \$[i]^-$, $i = 1, \ldots, l$), dobija se sledeći jednokriterijumski optimizationi problem:

$$\begin{array}{ll} \text{Min:} & \sum_{i=1}^{l} (\$[i]^+ + \$[i]^-) \\ \text{p.o.:} & \mathbf{x} \in \mathbf{X} \\ & Q_i(\mathbf{x}) - \$[i]^+ + \$[i]^- = b_i, \; i = 1, \ldots, l \\ & \$[1]^+ \geq 0, \ldots, \$[l]^+ \geq 0, \$[1]^- \geq 0, \ldots, \$[l]^- \geq 0. \end{array}$$

Lista ograničenja se transformiše ciklusom:

```
For[i=1,i<=l,i++,
   AppendTo[constr,q[[i]]-$[2i-1]+$[2i]==b[[i]]];
];
For[i=1,i<=2l,i++, AppendTo[constr,$[i]>=0]  ];
```

Odgovarajuća jednokriterijumska ciljna funkcija se generiše i minimizira izrazom

```
s=Minimize[Array[$,2l,1,Plus],constr,Union[Variables[Array[$,2l,1]],var]];
```

Standardna funkcija `Variables[expr]` daje listu nezavsinih promenljivih u izrazu `expr`. Izraz `Union[l`$_1$`,l`$_2$`,...]` daje sortiranu listu različitih elemenata koji se pojavljuju u bilo kom od izraza l_i [29].

Primer 2.8.1 Dat je zadatak VLP

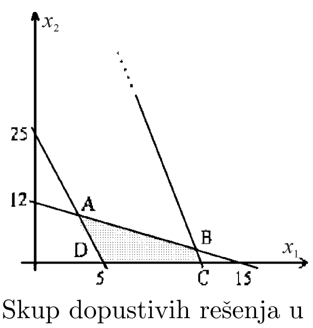
$$\begin{array}{ll} \text{Max:} & [Q_1(\mathbf{x}), Q_2(\mathbf{x}), Q_3(\mathbf{x})] \\ \text{p.o.:} & 35x_1 + 7x_2 \geq 175 \\ & 136x_1 + 170x_2 \leq 2400 \\ & 20x_1 + 3x_2 \leq 240 \\ & x_1, x_2 \geq 0 \end{array}$$

gde su :

$$Q_1(\mathbf{x}) = 4x_1 + 5x_2, \; Q_2(\mathbf{x}) = x_1 + x_2, \; Q_3(\mathbf{x}) = 40x_1 + 6x_2.$$

a) Rešiti zadatak metodom rastojanja sa pravougaonom metrikom, ako se želi dostići idealna tačka.

b) Koja rešenja predstavljaju Pareto, a koja slabi Pareto optimum polaznog zadatka.



Slika 3.8.2. Skup dopustivih rešenja u primeru f^z.

Rešenje. a) U ovom zadatku DO je odredio da su željene vrednosti kriterijumskih funkcija idealne vrednosti tih funkcija. Zato je najpre potrebno odrediti marginalna rešenja i odgovarajuće idealne vrednosti funkcije:

$f_1^* = 60$ za višestruko rešenje koje pripada duži **AB** sa krajnjim tačkama:

$$x_1^{(1)^*} = 3.09,\ x_2^{(1)^*} = 9.52$$

i

$$x_1^{(1)^{**}} = 11.59,\ x_2^{(1)^{**}} = 2.73.$$

$f_2^* = 14.32$ za : $x_1^{(2)^*} = 11.59$, $x_2^{(2)^*} = 2.73$ (tačka **B**),
$f_3^* = 480$ za višestruko rešenje koje pripada duži **BC** sa krajnjim tačkama :

$$x_1^{(3)^{**}} = 11.59,\ x_2^{(3)^{**}} = 2.73$$

i

$$x_1^{(3)^{**}} = 12,\ x_2^{(3)^{**}} = 0.$$

Zadatak VKO prilagođen za rešavanje metodom rastojanja sada ima oblik :

$$(min)\Phi(\mathbf{x}) = |f_1^* - f_1(\mathbf{x})| + |f_2^* - f_2(\mathbf{x})| + |f_3^* - f_3(\mathbf{x})|$$

pri istim ograničenjima.

Posto sve kriterijume trebamo maksimizirati, a po definiciji marginalnih rešenja je $f_k^* \geq f_k(x)$, može se staviti da je $|f_k^* - f_k(\mathbf{x})| = f_k^* - f_k(\mathbf{x})$, $k = 1, 2, 3$. Tada se umesto polaznog zadatka može rešavati zadatak minimizacije funkcije $\Phi(\mathbf{x}) = 554,32 - 45x_1 - 12x_2$, pri istim ograničenjima. Pokazaćemo kako se ovaj zadatak rešava opštim postupkom koji koristi definiciju prebačaja i podbačaja.U tom slučaju matematički model ima oblik :

$$\begin{array}{ll}
\text{Min:} & F(x,y) = y_1^+ + y_1^- + y_2^+ y_2^- + y_3^+ + y_3^- \\
\text{p.o.:} & 4x_1 + 5x_2 - y_1^+ + y_1^- = 60 \\
& x_1 + x_2 - y_2^+ + y_2^- = 14,32 \\
& 40x_1 + 6x_2 - y_3^+ + y_3^- = 480 \\
& 35x_1 + 7x_2 \geq 175 \\
& 136x_1 + 170x_2 \leq 2400 \\
& 20x_1 + 3x_2 \leq 240 \\
& x_1, x_2, y_1^+, y_1^-, y_2^+, y_2^-, y_3^+, y_3^- \geq 0.
\end{array}$$

Rešavanjem ovog zadatka dobija se jedinstveno rešenje : $F^* = 0$, $x_1^* = 11.59$ i $x_2^* = 2.73$.

To što je vrednost funkcije cilj jednako nuli ukazuje da je dobijeno rešenje savršeno jer nema ni prebačaja ni podbačaja vrednosti funkcija cilja od idealne tačke. Ovo je očigledno i na grafiku ; vidimo da u tački **B** sve funkcije cilja dostižu svoju najveću vrednost.

b) Jedino je tačka **B**(11.59, 2.73) Pareto optimalna. Sve tačke na duži **AB** na jednu stranu i **BC** na drugu stranu su slabi Pareto optimumi.

Sledi funkcija **MultiDist** kojom se daje implementacija metode rastojanja. Kao pomoćna funkcija koristi se funkcija **ideal** [22].

```
ideal[q_,constr_,var_]:=
  Module[{res={},i,l=Length[q]},
     For[i=1,i<=l,i++,
```

```
      AppendTo[res,First[Maximize[q[[i]],constr,var]]];
    ];
    Return[res];
  ]

MultiDist[q_,constr_,w_List,var_]:=
 Module[{con=constr,point={},l=Length[q],s},
   If[w=={}, point=ideal[q,con,var], point=w];
   For[i=1,i<=l,i++,
      AppendTo[con,q[[i]]-$[2i-1]+$[2i]==point[[i]]]; (* 3V,2V *)
   ];
   For[i=1,i<=2l,i++, AppendTo[con,$[i]>=0]  ];
   s=Minimize[Array[$,2l,1,Plus],con,Union[Variables[Array[$,2l,1]],var]];
   (* 3V,2V *)
   Return[{q/.Last[s],Last[s]}]
  ];
```

Primer 2.8.2 Rešiti problem

$$\begin{array}{ll} \text{Max:} & \{Q_1(\mathbf{x}) = x_1,\ Q_2(\mathbf{x}) = x_2\} \\ \text{p.o.:} & 0 \leq x_1 \leq 1,\ 0 \leq x_2 \leq 1. \end{array}$$

Ovaj problem se rešava izrazom

In[5]:= `MultiDist[{x1,x2},{x1>=0,x2>=0,x1<=1,x2<=1},{},{x1,x2}]`

Pozivom funkcije *ideal*, dobija se idealna tačka $\{2, 3\}$. Posle toga, dobija se sledeći:

$$\begin{array}{ll} \text{Min:} & \Phi(\mathbf{x}) = |2 - x_1| + |3 - x_2| \\ \text{p.o.:} & x_1 \leq 1,\ x_2 \leq 1\ x_1 \geq 0,\ x_2 \geq 0. \end{array}$$

Posle zamena $y_1 = 2-x1 = \$[1]^- - \$[1]^+$, $y_2 = 3-x2 = \$[2]^- - \$[2]^+$, koristeći $|y_1| = \$[1]^+ + \$[1]^-$, $|y_2| = \$[2]^+ + \$[2]^-$, s obzirom na (2.7), polazni problem se transformiše u sledeći:

$$\begin{array}{ll} \text{Min:} & \$[1]^+ + \$[1]^- + \$[2]^+ + \$[2]^- \\ \text{p.o:} & x_1 - \$[1]^+ + \$[1]^- = 2,\ x_2 - \$[2]^+ + \$[2]^- = 3, \\ & x_1 \leq 1,\ x_2 \leq 1, \\ & x_1 \geq 0,\ x_2 \geq 0,\ \$[1]^+ \geq 0,\ \$[1]^- \geq 0,\ \$[2]^+ \geq 0,\ \$[2]^- \geq 0. \end{array}$$

Unutrašnja reprezentacija ovog problema je

$\$[1] + \$[2] + \$[3] + \$[4],$
$\{x1 >= 0, x2 >= 0, x1 <= 1, x2 <= 1,$
$x1 - \$[1] + \$[2] == 2, x2 - \$[3] + \$[4] == 3,$
$\$[1] >= 0, \$[2] >= 0, \$[3] >= 0, \$[4] >= 0\},$
$\{x1, x2, \$[1], \$[2], \$[3], \$[4]\}$

dok je njegovo optimalno rešenje

$\{3,\ \{x1 -> 1, x2 -> 1, \$[1] -> 0, \$[2] -> 1, \$[3] -> 0, \$[4] -> 2\}\}$

Prema tome, optimalno rešenje polaznog problema je

Out[5]= `{{1,1},{x1->1, x2->1, $[1]->0,$[2]->1,$[3]->0,$[4]->2}}`

2.9 Metod PROMETHEE

Metod **PROMETHEE** (Preference Ranking Organization METHods for Enrichment Evaluation) je metod višekriterijumske analize i služi za rangiranje konačnog broja alternativa. Postoje četiri varijante ovog metoda: PROMETHEE I, II, III i IV (poslednja predstavlja proširenje za neprekidne skupove). Opisaćemo metod II, koja daje potpuni poredak alternativa [19].

Cilj nam je da, od N tačaka $x^{(1)}$,..., $x^{(N)}$ u skupu dopustivih rešenja S izaberemo onu koja daje najbolju vrednost rešenja.

Uvedimo funkciju preferencije $P_i(x^{(1)}, x^{(2)})$ za alternative $x^{(1)}$ i $x^{(2)}$ u odnosu na kriterijum Q_i:

$$P_i(x^{(1)}, x^{(2)}) = \begin{cases} 0, & \text{ako je } Q_i(x^{(1)}) \geq Q_i(x^{(2)}) \\ P_i(f_i(x^{(2)}) - f_i(x^{(1)})), & \text{ako je } Q_i(x^{(1)}) < Q_i(x^{(2)}) \end{cases}$$

Uvodimo oznaku $d = Q_i(x^{(2)}) - Q_i(x^{(1)})$.

Predloženo je šest tipova funkcije preferencije:

1. Jednostavan kriterijum

$$P_i(x^{(1)}, x^{(2)}) = \begin{cases} 0, & \text{ako je } d \leq 0 \\ 1, & \text{ako je } d > 0 \end{cases}$$

2. Kvazi-kriterijum

$$P_i(x^{(1)}, x^{(2)}) = \begin{cases} 0, & \text{ako je } d \leq q \\ 1, & \text{ako je } d > q \end{cases}$$

3. Kriterijum sa linearnom preferencijom

$$P_i(x^{(1)}, x^{(2)}) = \begin{cases} 0, & \text{ako je } d \leq 0 \\ \frac{d}{p,} & \text{ako je } 0 < d \leq p \\ 1, & \text{ako je } d > p \end{cases}$$

4. Nivoski kriterijum

$$P_i(x^{(1)}, x^{(2)}) = \begin{cases} 0, & \text{ako je } d \leq q \\ \frac{1}{2,} & \text{ako je } q < d \leq p \\ 1, & \text{ako je } d > p \end{cases}$$

5. Kriterijum sa linearnom preferencijom i oblašću indiferentnosti

$$P_i(x^{(1)}, x^{(2)}) = \begin{cases} 0, & \text{ako je } d \leq q \\ \frac{d-q}{p-q,} & \text{ako je } q < d \leq p \\ 1, & \text{ako je } d > p \end{cases}$$

6. Gausov kriterijum

$$P_i(x^{(1)}, x^{(2)}) = \begin{cases} 0, & \text{ako je } d \leq 0 \\ 1 - e^{\frac{-d^2}{2\sigma^2}}, & \text{ako je } d > 0 \end{cases}$$

Pri tome paremetre p, q, σ treba zadati za svaku kriterijumsku funkciju.

Definišemo višekriterijumski indeks preferencije alternative $x^{(1)}$ nad $x^{(2)}$

$$\Pi(x^{(1)}, x^{(2)}) = \sum_{i=1}^{k} w_i P_i(x^{(1)}, x^{(2)})$$

gde je w_i težina i-tog kriterijuma. Podrazumeva se $\sum_{i=1}^{k} w_i = 1$.

Uvodimo zatim tokove preferencije:

$$\begin{aligned} \phi_i^+(x^{(i)}) &= \sum_{m=1}^{N} \Pi(x^{(i)}, x^{(m)}) \qquad \text{(pozitivni tok)} \\ \phi_i^-(x^{(i)}) &= \sum_{m=1}^{N} \Pi(x^{(m)}, x^{(i)}) \qquad \text{(negativni tok)} \\ \phi_i(x^{(i)}) &= \phi_i^+(x^{(i)}) - \phi_i^-(x^{(i)}) \qquad \text{(neto tok).} \end{aligned}$$

Neto tok je funkcija pomoću koje rangiramo alternative. Alternativa $x^{(i)}$ je višekriterijumski bolja od $x^{(j)}$ ako je $\phi_i(x^{(i)}) > \phi_j(x^{(j)})$.

Time smo dobili potpuno uređenje skupa alternativa na osnovu koga možemo izabrati najbolju.

2.10 Metod ELECTRE

Metod ELECTRE I (ELimination and ET Choice Translating REality) prvi put je objavio Roy sa svojim saradnicima (1971.). Za određivanje delimičnih poredaka alternativa najčešće se koristi metod ELECTRE I, a za potpuno uređenje skupa alternativa metod ELECTRE II. Ovi metodi omogućavaju parcijalno uređenje skupa rešenja na osnovu preferencija donosioca odluke, a pogodne su za diskretne probleme i raznorodne kriterijumske funkcije. Modeli dozvoljavaju uključivanje subjektivnih procena, bilo kroz vrednosti kriterijumskih funkcija, bilo kroz relativne važnosti pojedinih kriterijuma. Metodi ELECTRE III i IV su metodi "višeg" ranga.

Metod **ELECTRE** je metod višekriterijumske analize pomoću koga se može dobiti delimično uređenje skupa alternativa. Na osnovu dobijene relacije delimičnog uređenja ρ konstruiše se graf G u kome čvorovi predstavljaju alternative. Grana grafa se usmerava od čvora $x^{(i)}$ prema čvoru $x^{(j)}$ ako je $x^{(i)} \rho x^{(j)}$. Na osnovu tako konstruisanog grafa dobija se parcijalna rang lista (nepotpuno uređen skup) alternativa, a može se desiti da neke alternative ostanu izolovane.

Dakle, metod ELECTRE je razvijen za analizu odnosa među alternativama, a ne za potpuno uređenje skupa alternativa.

Opisaćemo ukratko noviju varijantu ELECTRE II [19].

Označimo sa $I = \{1, \ldots, l\}$ skup indeksa kriterijumskih funkcija. Za svaki par rešenja $(x^{(1)}, x^{(2)})$ definišemo:

$$I^{+}(x^{(1)}, x^{(2)}) := \{i \in I \mid Q_i(x^{(1)}) < Q_i(x^{(2)})\}$$

$$I^{=}(x^{(1)}, x^{(2)}) := \{i \in I \mid Q_i(x^{(1)}) = Q_i(x^{(2)})\}$$
$$I^{-}(x^{(1)}, x^{(2)}) := \{i \in I \mid Q_i(x^{(1)}) > Q_i(x^{(2)})\}$$

Donosilac odluke zadaje težine kriterijuma w_i, $i = 1 \ldots, k$. Zatim se određuje

$$\begin{aligned} W^{+}(x^{(1)}, x^{(2)}) &= \sum_{i \in I^{+}} w_i \\ W^{=}(x^{(1)}, x^{(2)}) &= \sum_{i \in I^{=}} w_i \\ W^{-}(x^{(1)}, x^{(2)}) &= \sum_{i \in I^{-}} w_i \\ W &= W^{+} + W^{=} + W^{-}. \end{aligned}$$

Definišemo indeks saglasnosti

$$C(x^{(1)}, x^{(2)}) = \frac{W^{+} + W^{=}}{W}$$

Uslov saglasnosti za par $(x^{(1)}, x^{(2)})$ je ispunjen ako važi

$$C(x^{(1)}, x^{(2)}) \geq q \quad \text{i} \quad \frac{W^{+}}{W^{-}} > 1$$

gde je q parametar čijim variranjem ćemo dobijati različite rezultate.

Da bismo definisali indeks nesaglasnosti moramo uvesti intervalnu skalu S koja će omogućiti poređenje funkcija raznorodnih vrednosti. Tako se dobijaju funkcije s_i koje predstavljaju surogat kriterijumskih funkcija Q_i.

Indeks nesaglasnosti je:

$$d(x^{(1)}, x^{(2)}) = \frac{1}{S} \max_{i \in I^{-}} |s_i(x^{(1)}) - s_i(x^{(2)})|$$

Uslov nesaglasnosti za par $(x^{(1)}, x^{(2)})$ je ispunjen ako važi

$$d(x^{(1)}, x^{(2)}) \leq r$$

gde variranjem parametra r dobijamo različite rezultate.

Alternativa $x^{(j)}$ je superiorna nad $x^{(k)}$ ako su ispunjeni odgovaraju 'ci uslovi saglasnosti i nesaglasnosti. U tom slučaju grana $(x^{(j)}, x^{(k)})$ ulazi u rezultantni graf.

Povećanjem vrednosti parametra q i smanjivanjem vrednosti parametra r smanjuje se broj grana rezultantnog grafa. U praksi je dobro ispitati vrednosti $0.5 \leq q \leq 1$ i $0 \leq r \leq 0.5$.

Iz rezultantnog grafa možemo identifikovati alternative nad kojima nema superiornih. Naravno, neke alternative mogu ostati izolovane. Treba imati na umu da se metodom ELECTRE pre mogu dobiti korisne informacije o alternativama nego što se može potpuno rešiti problem multikriterijumske analize.

2.11 Interaktivni metodi

U interaktivnim metodima DO aktivno učestvuje tokom rešavanja problema. Najpre DO daje preliminarne informacije o svojim preferencijama na osnovu kojih analitičar generiše neki skup rešenja ili neki skup novih korisnih informacija. Kada dobije ove podatke DO obezbeđuje nove informacije o svojim zahtevima i postupak se iterativno ponavlja sve dok DO ne bude konačno zadovoljan dobijenim rešenjem. Prednost ovakvog pristupa reävanju VKO je taj što se generiše samo deo Pareto optimalnog skupa i što DO iskazuje i menja svoje odluke i preferencije tokom samog procesa rešavanja problema.

Postoji mnogo interaktivnih metoda a mi ćemo navesti ukratko jednu od njih.

2.12 Metod referentne tačke

U raznim metodima VKO se koristi takozvana *funkcija ostvarenja* (achievement function). Uočimo referentnu tačku $\overline{z} \in \mathbf{R}^k$ čije su koordinate željene vrednosti kriterijumskih finkcja. Tačka $\overline{z}$ može biti ili ne biti dostižna. Funkcija ostvarenja je funkcija $s_{\overline{z}}$ koja zavisi od $\overline{z}$,

$$s_{\overline{z}} : Z \to \mathbf{R}$$

Posmatrajmo problem:

$$\text{Minimizirati } s_{\overline{z}}(Q(\mathbf{x})) \text{ pod uslovom } \mathbf{x} \in S \tag{2.12.0.1}$$

Teorema 2.12.1 *Ako je funkcija ostvarenja $s_{\overline{z}} : Z \to \mathbf{R}$ strogo rastuća, tada je rešenje problema* (2.12.0.1) *slabo Pareto optimalno rešenje početnog problema VKO. Ako je izvršna funkcija $s_{\overline{z}} : Z \to \mathbf{R}$ jako rastuća, tada je rešenje problema* (2.12.0.1) *Pareto optimalno rešenje početnog problema VKO.*

Dokaz. Pretpostavimo suprotno, da je $\mathbf{x}^* \in S$ rešenje Problema (2.12.0.1), a da ono nije slabo Pareto optimalno. Sledi da postoji neko $\mathbf{x} \in S$ za koje važi da je $Q_i(\mathbf{x}) <$

$Q_i(\mathbf{x}^*)$ za svako $i = 1, \dots, k$. Kako je $s_{\overline{z}}$ strogo rastuća funkcija, dobijamo $s_{\overline{z}}(Q(\mathbf{x})) < s_{\overline{z}}(Q(\mathbf{x}^*))$, odakle dobijamo kontradikciju sa pretpostavkom da je $\mathbf{x}^*$ rešenje problema (2.12.0.1).

Drugi deo teoreme se dokazuje slično. Pretpostavimo da je $\mathbf{x}^*$ rešenje problema (2.12.0.1), a da ono nije Pareto optimalno. Dakle, postoji $\mathbf{x} \in S$ za koje važi $Q_i(\mathbf{x}) \leq Q_i(\mathbf{x}^*)$ za svako $i = 1, \dots, k$, pri čemu za neko $j \leq k$ važi $Q_j(\mathbf{x}) < Q_j(\mathbf{x}^*)$. Kako je $s_{\overline{z}}$ jako rastuća funkcija dobijamo $s_{\overline{z}}(f(\mathbf{x})) < s_{\overline{z}}(Q(\mathbf{x}^*))$, odakle sledi da $\mathbf{x}^*$ nije rešenje problema (2.12.0.1), što je kontradikcija. $\square$

U metodu referentne tačke generisanje Pareto optimalnog skupa je bazirano na referentnoj tački, a ne na vrednosnoj funkciji ili težinskim koeficijentima. Najpre se donosiocu odluke, ako je moguće, dostave neke informacije o problemu (idealna tačka, maksimalne vrednosti kriterijumskih funkcija u odnosu na dopustivi skup i slično). Takođe, potrebno je odrediti odgovarajuću funkciju ostvarenja. Na osnovu prethodne teoreme se može zaključiti da je poželjno da izvršna funkcija bude jako rastuća. Metod se sastoji iz sledećih koraka [**?**]:

1. Donosiocu odluka se predstave informacije o problemu. Stavi se $h = 1$.

2. DO navodi referentnu tačku $\overline{z}^h$ (koordinate ove tačke su željeni nivoi kriterijumskih funkcija).

3. Minimizira se funkcija ostvarenja pod uslovom $\mathbf{x} \in S$ i dobije se Pareto optimalno rešenje $\mathbf{x}^h$ i odgovarajuće z^h. Rešenje z^h se predstavi DO.

4. Minimizira se k funkcija ostvarenja $s_{\overline{z}(i)}$ sa referentnim tačkama $\overline{z}(i) = \overline{z}^h + ||\overline{z}^h - z^h||e^i$, gde je e^i i-ti jedinični vektor, $i = 1, \dots, k$. Na ovaj način se dobije k novih Pareto optimalnih rešenja. Dakle, ukupno imamo $k + 1$ dominantnih rešenja, tj. $k + 1$ odgovarajućih tačaka u kriterijumskom skupu Z.

5. Ako među ovih $k + 1$ tačaka DO izabere neku kao zadovoljavajuću, onda je odgovarajuće x^h konačno rešenje. U suprotnom, DO bira (među dobijenih $k + 1$ tačaka skupa Z) novu referentnu tačku $\overline{z}^{h+1}$. Postavljamo $h = h + 1$ i prelazimo na korak 3.

Prednost ovog metoda je što DO neposredno upravlja procesom rešavanja problema i što je u mogućnosti da menja svoje mišljenje tokom procesa rešavanja. Nedostatak je što to može dugo da traje i što DO ne može biti siguran da je u koraku 5 koraku dobro izabrao novu referentnu tačku.

Literatura

[1] H.P. Benson, *Existence of efficient solution for vector maximization problems*, Journal of Optimization Theory and Applications, **28** (1978) 569-580.

[2] D.G. Carmichael, *Computation of Pareto optima in structural design*, Int. J. Numer. Methods Eng. 15, (1980) 925-952.

[3] V. Chankong, Y. Haimes, *Multiobjective decision making: Theory and methodology Series, Volume 8*, North-Holland, New York, Amsterdam, Oxford, 1983.

[4] A. Charnes, W.W. Cooper, *Management Models and Industrial Applications of Linear Programming*, New York: John Wiley and Sons, 1961.

[5] A. Charnes, W.W. Cooper, *Goal programming and multiple objective optimization; part 1*, Eur. J. Oper. Res. 1, (1977) 39-54.

[6] C.A. Coello, *A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques*,

[7] M. Ćupić, R.V.M. Tummala, M. Suknović M., *Odlučivanje: Formalni pristup*, FON, Beograd, 2001.

[8] I. Das, J.E Dennis, *A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems*, Struct. Optim. **14** (1997) 63-69.

[9] Y.Y. Haimes, L.S. Lasdon, D.A. Wismer, *On a bicriterion formulation of the problems of integrated system identification and system optimization*, IEEE Trans. Syst. Man Cybern. SMC-1, (1971) 296-297.

[10] B.F. Hobbs, *A comparison of weighting methods in power plant siting.*, Decis. Sci., **11** (1980) 725-737.

[11] C.L. Hwang, K. Yoon, *Multiple attribute decision making methods and applications: a state-of-the-art survey.*, In: Beckmann, M.; Kunzi, H.P. (eds.) Lecture Notes in Economics and Mathematical Systems, No. 186. Berlin: Springer-Verlag, 1981.

[12] C.L. Hwang, Md. Masud, A.S., in collaboration with Paidy, S.R. and Yoon, K. *Multiple objective decision making, methods and applications: a state-of-the-art survey* In: Beckmann, M.; Kunzi, H.P. (eds.) Lecture Notes in Economics and Mathematical Systems, No. 164. Berlin: Springer-Verlag, 1979.

[13] Y. Ijiri, *Management Goals and Accounting for Control*, Amsterdam: North-Holland, 1965.

[14] J.P. Ignizio, *Linear programming in single-multiple-objective systems*, Englewood Cliffs: Prentice Hall, 1982.

[15] K.R. Mac-Crimmon, *An overview of multiple objective decision making, in J.L. Cochrane and M. Zeleny (Eds), Multiple criteria decision making*, University of South Carolina Press, Columbia, 1973.

[16] R. Maeder, *Programming in Mathematica, Third Edition* Redwood City, California: Adisson-Wesley, 1996.

[17] R.T. Marler, *Survey of multi-objective optimization methods for engineering* Struct. Multidisc. Optim. **26** (2004) 369–395.

[18] I. Nikolić, S. Borović., *Višekriterijumska optimizacija: metode, primena u logistici*, Centar vojnih škola Vojske Jugoslavije, Beograd, 1996.

[19] S. Opricović, *Optimizacija sistema*, Građevinski fakultet Univerziteta u Beogradu, 1992.

[20] A. Osyczka, *Multicriterion Optimization in Engineering with Fortran Programs*, New York: John Wiley and Sons, 1984.

[21] W. Stadler, *Fundamentals of multicriteria optimization*, In: Stadler,W. (ed.) Multicriteria Optimization in Engineering and in the Sciences, pp. 125. New York: Plenum Press, 1988.

[22] P.S. Stanimirović and I.P. Stanimirović, *Implementation of polynomial multi-objective optimization in MATHEMATICA*, Structural Multidisciplinary Optimization, U štampi.

[23] P.S. Stanimirović, G.V. Milovanović, *Programski paket MATHEMATICA i primene*, Elektronski fakultet u Nišu, Edicija monografije, Niš, 2002.

[24] P.S. Stanimirović, N.V. Stojković, M.D. Petković, *Run-time transformations in implementation of linear multi-objective optimization*, PRIM, Budva 2004.

[25] H. Voogd, *Multicriteria Evaluation for Urban and Regional Planning*, London: Pion, 1983.

[26] M. Vujošević , M. Stanojević , N. Mladenović , *Metode optimizacije*, Društvo operacionih istraživača Jugoslavije, Beograd, 1996.

[27] F.M. Waltz *An engineering approach: hierarchical optimization criteria*, IEEE Trans. Autom. Control AC-12, (1967) 179-180.

[28] R.E. Wendell, D.N. Lee, *Efficiency in multiple objective optimization problems*, Mathematical Programming, **12** (1977) 406-414.

[29] S. Wolfram, *The Mathematica Book, 4th ed.*, Wolfram Media/Cambridge University Press, 1999.

[30] L.A. Zadeh, *Optimality and non-scalar-valued performance criteria*, IEEE Trans. Autom. Control AC-8, (1963) 5960.

[31] S. Zionts, *Multiple criteria mathematical programming: an updated overview and several approaches.*, In: Mitra, G. (ed.) Mathematical Models for Decision Support, 1988, pp. 135167, Berlin: Springer-Verlag

[32] K. Zotos, *Performance comparison of Maple and Mathematica*, Appl. Math. Comput. (2006).