Hijerarhijsko odlučivanje pomoću AHP metode

Piljić, Želimir

Sveučilište J.J. Strossmayera u Osijeku
Odjel za matematiku

Želimir Piljić

Hijerarhijsko odlučivanje pomoću AHP metode

Diplomski rad

Osijek, 2019.

Sveučilište J.J. Strossmayera u Osijeku
Odjel za matematiku

Želimir Piljić

Hijerarhijsko odlučivanje pomoću AHP metode

Diplomski rad

Voditelj: izv.prof.dr.sc. Zoran Tomljanović

Osijek, 2019.

Sadržaj

1 Uvod

Svakim danom suočeni smo s odlučivanjem. Uglavnom su to trivijalne odluke na koje ne obraćamo pažnju i koje su postale dio naše rutine. To mogu biti odluke kao na primjer što odjenuti, što doručkovati i slično. Međutim, postoje i odluke u svakodnevnom životu koje ipak zahtijevaju malo više razmišljanja kao što je kojim prijevozom ići na posao, koji automobil kupiti, koji fakultet upisati itd.

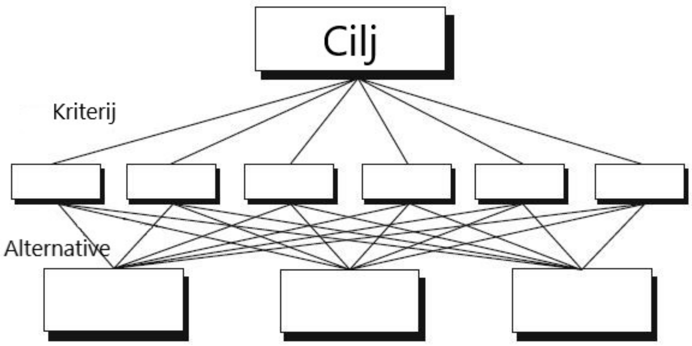
Primjera je bezbroj i potrebno je domisliti metodu koja će na sustavan, analitičan i nedvosmislen način odgovarati na ta pitanja. Jedna takva metoda je analitičko hijerarhijski proces (*kr.* AHP).

1.1 Hijerarhijsko odlučivanje

Hijerarhija dolazi od grčke riječi *hierarkhia*, a doslovan prijevod bio bi *zakon visokog svećenstva*. U svakodnevnom govoru obično mislimo na razvrstavanje na temelju podređenosti, odnosno nadređenosti[1]. Mogli bismo reći da je to obrazac koji osigurava efektivno funkcioniranje organizacije ili zajednice.

Zamislimo da je ljubitelj automobila dobitnik nagradne igre na kojoj može birati između dva ponuđena automobila. Jedan je skupocjena limuzina, s kožnim sjedalima, pune opreme i velike snage. Drugi je skupocjeni terenac s kožnim sjedalima, pune opreme, velike snage koji ima manje oštećenje na karoseriji. Koji će automobil osoba vjerojatno izabrati?

Najjednostavnija forma kojom strukturiramo problem odlučivanja je hijerarhija koja se sastoji od tri nivoa: cilj na vrhu, zatim kriteriji te alternative na posljednjem (Slika 1).



Slika 1: Hijerarhija s tri nivoa

Dekompozicija u hijerarhiju složenih sustava je osnovni način koji ljudi koriste kako bi se nosili s različitostima. Tako se faktori organiziraju od općih na vrhu ka pojedinačnim na nižim nivoima. Svrha strukture je napraviti mogućim procjene važnosti elemenata unutar istog nivoa ili u odnosu na gornji susjedni. Kada se konstruira hijerarhija potrebno je uključiti dovoljno detalja kako bi se problem prikazao temeljito, ali ne previše detaljno kako bi se izgubila osjetljivost na promjenu elemenata. Aranžiranje ciljeva, atributa, problema i dionika ima dvije svrhe. Ono daje širu sliku složenih veza među elementima u procesu procjenjivanja. Također, ono daje odlučitelju procjenu uspoređuje li elemente iste relativne veličine.

[1]Hrvatski leksikon

2 AHP metoda

Pretpostavimo da zainteresirani pojedinac razmatra n alternativa i da je njegov zadatak:

1. dati procjene relativne važnosti između alternativa i
2. osigurati da su procjene kvantitativne do te razine da se mogu kvanitativno interpretirati između alternativa.

Želimo imati metodu koja će iz procjena koje se mogu kvantitativno opisati, dati skup težina pridruženih svakoj individualnoj alternativi.

Neka su $A_1, A_2, \ldots, A_n$ alternative. Kvantitativne procjene parova alternativa (A_i, A_j) obzirom na odabrani kriterij možemo predstaviti matricom $A = (a_{ij})$, $i, j = 1, \ldots, n$. Cilj metode je pridružiti svakoj od n alternativa $A_1, A_2, \ldots, A_n$ težine $w_1, w_2, \ldots, w_n$ koje će reflektirati dane procjene.

Na primjer, *Aristarh sa Samosa*[2] je izračunao da je udaljenost između Zemlje i Sunca 19 puta veća nego što je udaljenost između Zemlje i Mjeseca [2]. To znači da je moguće na dužini između Zemlje i Sunca označiti 19 crtica. No, ako imamo aposlutnu skalu i mjerimo udaljenost između Zemlje i Mjeseca w_1 jedinica i w_2 jedinica između Zemlje i Sunca onda su relativne udaljenosti redom $\frac{w_2}{w_1}$ i $\frac{w_1}{w_2}$, tj. recipročne vrijednosti. Takva reprezentacija je valjana samo ako w_1 i w_2 pripadaju istoj skali tako da je $\frac{w_1}{w_2}$ neovisno o mjernoj jedinici [12].

Formirajmo matricu W čiji retci sadrže omjere mjerenja w_i za svaki od n alternativa u odnosu na preostale alternative.

$$W = \begin{bmatrix} \frac{w_1}{w_1} & \frac{w_1}{w_2} & \cdots & \frac{w_1}{w_n} \\ \frac{w_2}{w_1} & \frac{w_2}{w_2} & \cdots & \frac{w_2}{w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{w_n}{w_1} & \frac{w_n}{w_2} & \cdots & \frac{w_n}{w_n} \end{bmatrix} = \begin{bmatrix} 1 & \frac{w_1}{w_2} & \cdots & \frac{w_1}{w_n} \\ \frac{w_2}{w_1} & 1 & \cdots & \frac{w_2}{w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{w_n}{w_1} & \frac{w_n}{w_2} & \cdots & 1 \end{bmatrix} \tag{1}$$

Uočimo da je (1) jednako umnošku

$$W = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \begin{bmatrix} \frac{1}{w_1} & \frac{1}{w_2} & \cdots & \frac{1}{w_n} \end{bmatrix}.$$

Definirajmo nekoliko pojmova koji će se protezati tijekom rada.

Definicija 2.1 (Pozitivna matrica)**.** Neka je dana matrica $A \in \mathbb{R}^{m \times n}$, $A = (a_{ij})$. Kažemo da je A pozitivna ako $a_{ij} > 0$ za sve $i = 1, \ldots, m$, $j = 1, \ldots, n$.

Definicija 2.2 (Recipročna matrica)**.** Neka je dana matrica $A \in \mathbb{R}^{n \times n}$, $A = (a_{ij})$, $a_{ij} \neq 0$, $i, j \in \{1, \ldots, n\}$. Kažemo da je A recipročna ako $a_{ij} = \frac{1}{a_{ji}}$ za sve $i, j = 1, \ldots, n$.

Svaka pozitivna recipročna matrica je sljedećeg oblika:

$$\begin{bmatrix} 1 & a_{12} & \cdots & a_{1n} \\ \frac{1}{a_{12}} & 1 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_{1n}} & \frac{1}{a_{2n}} & \cdots & 1 \end{bmatrix}. \tag{2}$$

[2] 310.–230.pr.Kr. Antički grčki matematičar i astronom.

Definicija 2.3 (Konzistentnost)**.** Neka je dana pozitivna recipročna matrica $A \in \mathbb{R}^{n\times n}$, $A = (a_{ij})$. Kažemo da je A (Saaty) konzistentna ako $a_{ij}a_{jk} = a_{ik}$, $i, j, k = 1, \dots, n$.

Teorem 2.1. *Pozitivna matrica $A \in \mathbb{R}^{n\times n}$, $A = (a_{ij})$ je konzistentna ako i samo ako postoje pozitivni brojevi $w_i > 0$, $i = 1, \dots, n$ takvi da je $a_{ij} = \frac{w_i}{w_j}$.*

Dokaz. Neka je $A \in \mathbb{R}^{n\times n}$ konzistentna. Definirajmo $w_i = \frac{a_{ij}}{a_{sj}}$, $s \in \{1, \dots, n\}$. Tada vrijedi

$$a_{ij} = \frac{a_{ik}}{a_{jk}} = \frac{a_{ik}}{a_{sk}} \cdot \frac{a_{sk}}{a_{jk}} = w_i \cdot \frac{1}{w_j} = \frac{w_i}{w_j}.$$

Dokažimo obrat. Neka je $a_{ij} = \frac{w_i}{w_j}$, $w_i > 0$, $i = 1, \dots, n$. Tada

$$a_{ij} \cdot a_{jk} = \frac{w_i}{\cancel{w_j}} \cdot \frac{\cancel{w_j}}{w_k} = \frac{w_i}{w_k} = a_{ik},$$

tj. A je konzistentna. □

Uočimo, ako je W matrica zadana sa (1) i $w = \left[w_1, \dots, w_n\right]^\top$, onda direktnim množenjem vidimo da vrijedi

$$Ww = nw. \tag{3}$$

Podsjetimo se teorema o rangu i defektu, rezultatu iz linearne algebre, čiji ćemo iskaz iskoristiti za dokaz sljedećeg teorema.

Teorem 2.2 (Teorem o rangu i defektu)**.** *Neka je $A\colon V \to W$ linearan operator takav da je* $\dim V < \infty$ *te neka je* $\mathrm{r}(A) = \dim(\mathrm{Im}\, A)$ *i* $\mathrm{d}(A) = \dim(\mathrm{Ker}\, A)$. *Tada je* $\mathrm{r}(A) + \mathrm{d}(A) = \dim V$.

Za dokaz Teorema 2.2 vidi [1, str. 105–106].

Teorem 2.3. *Neka je $A \in \mathbb{R}^{n\times n}$ pozitivna i recipročna matrica. Matrica A je konzistentna ako i samo ako je $\lambda_{max} = n$ najveća svojstvena vrijednost matrice A.*

Dokaz. Neka je $A \in \mathbb{R}^{n\times n}$ konzistentna. A je konzistetna pa je prema Teoremu 2.1 oblika (1). Tada je, prema (3), n jedna svojstvena vrijednost. Kako su svi retci jednaki do na množenje konstantom, tj. broj linearno nezavisnih redaka je 1, slijedi da je rang matrice A jednak 1. Kako je $M_{m\times n}(\mathbb{F}) \cong Hom(\mathbb{F}^n, \mathbb{F}^m)$, prema Teoremu 2.2 slijedi da je defekt matrice A, $\mathrm{d}(A) = n - 1$, tj. $(n-1)$ svojstvenih vrijednosti jednako je 0. Kako je $n > 0$, slijedi da je n najveća svojstvena vrijednost od A. Dokažimo obrat.

Neka je zadana pozitivna i recipročna matrica $A \in \mathbb{R}^{n\times n}$ te neka je n najveća svojstvena vrijednost od A, tj. $Aw = \lambda_{max}w$, $\lambda_{max} = n$, $w \neq 0$. Neka je $W \in \mathbb{R}^{n\times n}$, $W = (w_{ij})$, $w_{ij} = \frac{w_i}{w_j}$ za sve $i, j = 1, \dots, n$ te neka je matrica $E \in \mathbb{R}^{n\times n}$, $E = (e_{ij})$, $e_{ij} = a_{ij}\frac{w_j}{w_i}$ za sve $i, j = 1, \dots, n$. Kako je w svojstveni vektor s pripadnom svojstvenom vrijednosti λ_{max} imamo

$$\begin{aligned}
\sum_{j=1}^{n} e_{ij} &= \sum_{j=1}^{n} a_{ij}\frac{w_j}{w_i} \\
&= \frac{1}{w_i}\sum_{j=1}^{n} a_{ij}w_j \\
&= \frac{1}{\cancel{w_i}}\lambda_{max}\cancel{w_i} \\
&= \lambda_{max}.
\end{aligned}$$

Dakle, suma svih elemenata matrice E jednaka je $n\lambda_{max}$. Kako je E pozitivna recipročna matrica, prema (2) elementi na dijagonali jednaki su 1. Slijedi da je suma svih elemenata matrice E jednaka

$$\begin{aligned}\sum_{i,j=1}^{n} e_{ij} &= \sum_{i=1}^{n} e_{ii} + \sum_{1\leq i<j\leq n}\left(e_{ij} + \frac{1}{e_{ij}}\right)\\ &\geq n + 2\binom{n}{2}.\end{aligned}$$

Nejednakost slijedi iz činjenice da $x + x^{-1} \geq 2$ za sve $x > 0$. Jednakost vrijedi ako i samo ako $x = 1$. Stoga imamo, $n + 2\binom{n}{2} \leq n\lambda_{max}$ pri čemu jednakost vrijedi ako i samo ako $e_{ij} = 1$, za svaki $i, j = 1, \ldots, n$. Pretpostavimo li da je $\lambda_{max} = n$, tada $n + 2\binom{n}{2} = n^2$ te imamo jednakost. Dakle, $e_{ij} = 1$ za sve $i, j = 1, \ldots, n$ iz čega zaključujemo $A = W$. Kako je W prema Teoremu 2.1 konzistentna, slijedi da je A je konzistentna čime je teorem dokazan. □

2.1 Fundamentalna ljestvica

U teoriji podražaja, reakcije i posljedično ljestvice omjera, u 18. i 19. stoljeću, ističu se *Ernst Heinrich Weber*[3] i *Gustav Theodor Fechner*[4]. Weber je 1846. godine formulirao zakon izmjerljivih podražaja. Tijekom istraživanja otkrio je da npr. ljudi mogu razlikovati uteg od 20 g koji drže u jednoj ruci od utega od 21 g kojeg drže u drugoj ruci. No, ako bi uteg u drugoj ruci bio 20.5 g, više ne bi mogli razlikovati. Dodatno, ne bi mogli razlikovati utege od 40 g i 41 g, ali bi mogli 40 g i 42 g [11].

Dakle, potrebno je povećati podražaj s za najmanje Δs kako bi ljudski osjeti najranije razlikovali s od $s+\Delta s$. Δs se naziva jedva zamjetna razlika (*eng.* just noticeable difference). Omjer $r = \frac{\Delta s}{s}$ ne ovisi o s. Weberov zakon kaže da se razlika u osjetu zamijeti onda kada je podražaj povećan za konstantni postotak samog podražaja. Ovaj zakon vrijedi kada je Δs puno manji od s, a u praksi ne vrijedi kada je s ili premal ili prevelik. Fechner je 1860. promatrao niz jedva zamjetnih povećanja podražaja. Prvi je označio sa s_0. Idući jedva zamjetni podražaj, koristeći Weberov zakon, bio bi

$$s_1 = s_0 + \Delta s_0 = s_0 + \frac{\Delta s_0}{s_0} s_0 = s_0(1 + r) = s_0\alpha.$$

Slično,

$$s_2 = s_1 + \Delta s_1 = s_1(1 + r) = s_0(1 + r)^2 = s_0\alpha^2.$$

Općenito,

$$s_n = s_{n-1}\alpha = s_0\alpha^n \, , n = 1, 2, \ldots \tag{4}$$

Dakle, niz jedva zamjetnih povećanja podražaja je geometrijski niz. Fechner je dodao da bi pripadni niz osjeta trebao biti aritmetički niz diskretnih točaka u kojima se jedva zamjetna razlika događa. Rješavanjem (4) po n dobivamo

$$n = \frac{\log s_n - \log s_0}{\log \alpha},$$

[3]1795.–1878. Njemački fiziolog. Smatra se jednim od osnivača eksperimentalne psihologije.

[4]1801.–1887. Njemački fizičar, filozof, psiholog i književnik. Utemeljitelj tzv. eksperimentalne estetike i psihofizike, znanosti koja se bavi kvantitativnim odnosima između osjeta i podražaja koji ih izazivaju.

tj. osjet je linearna funkcija logaritama podražaja. Ako M označava osjet i s podražaj, Weber–Fechnerov zakon psihologije dan je s

$$M = a \log s + b, a \neq 0.$$

Pretpostavljamo da osjet igra ulogu u uspoređivanju po parovima aktivnosti koje su smisleno usporedive. Kako nas zanimaju samo omjeri uzastopnih zamjetnih osjeta, stavljamo $b = 0$ iz čega slijedi da moramo imati $\log s_0 = 0$, tj. $s_0 = 1$ koji ima svrhu kalibriranja podražaja. To se postiže uspoređivanjem jedne aktivnosti sa samom sobom. Sljedeći zamjetni osjet za dani podražaj dan je s

$$s_1 = s_0 \alpha = \alpha,$$

tj. $n = 1$. Sljedeći podražaj je

$$s_2 = s_0 \alpha^2$$

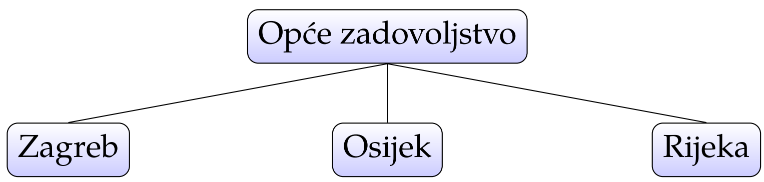
što daje $n = 2$. Na ovakav način dobiven je niz $1, 2, 3, \ldots$

Moguće je kvalitativno razlikovati pet intenziteta: jednako, umjereno, jako, vrlo jako i ekstremno. Ukoliko je potrebna veća preciznost, između dvaju susjednih intenziteta moguće je dodati po još jedan. Dakle, zahtijevamo devet vrijednosti koje, obzirom na prethodnu raspravu, trebaju biti uzastopne.

Intenzitet važnosti	Definicija	Objašnjenje
1	Jednaka važnost	Dvije aktivnosti doprinose jednako obzirom na kriterij
2	Slaba	
3	Umjerena važnost	Iskustvo i prosudba blago favoriziraju jednu aktivnost u odnosu na drugu
4	Umjerenija	
5	Jaka važnost	Iskustvo i prosudba jako favoriziraju jednu aktivnost u odnosu na drugu
6	Jača	
7	Vrlo jaka važnost	Aktivnost je vrlo jako favorizirana u odnosu na drugu; dominantnost je vidljiva u praksi
8	Vrlo vrlo jako	
9	Ekstremna važnost	Dokaz za favoriziranje jedne aktivnosti u odnosu na drugu je na najvećoj mogućoj razini
Recipročne vrijednosti gore navedenih	Ako aktivnost i ima jednu od gore navedenih vrijednosti u usporedbi s aktivnosti j, onda j ima recipročnu vrijednost kada se uspoređuje s i	Smislena pretpostavka
Racionalne vrijednosti	Omjeri dobiveni iz ljestvice	Ako je potrebno postići konzistentnost dobivanjem n numeričkih vrijednosti koje razapinju matricu

Tablica 1: Fundamentalna ljestvica [10]

Primjer 2.1. *Pretpostavimo da želimo odabrati mjesto studiranja te da biramo između Zagreba, Osijeka i Rijeke u odnosu na kriterij opće zadovoljstvo.*



Slika 2: Hijerarhija alternativa u odnosu na kriterij opće zadovoljstvo

Zagreb preferiramo vrlo jako u odnosu na Osijek, Zagreb preferiramo vrlo vrlo jako u odnosu *na Rijeku, a Osijek preferiramo umjereno u odnosu na Rijeku. Tada je matrica usporedbi jednaka*

$$\begin{array}{c} \\ ZG \\ OS \\ RI \end{array} \begin{array}{c} \begin{array}{ccc} ZG & OS & RI \end{array} \\ \begin{bmatrix} 1 & 7 & 8 \\ 1/7 & 1 & 3 \\ 1/8 & 1/3 & 1 \end{bmatrix} \end{array}.$$

Uočimo kako nam je bilo dovoljno $\binom{n}{2} = \frac{n(n-1)}{2}$ *usporedbi da bismo popunili cijelu matricu, pri čemu je* n *broj alternativa.*

Primjer 2.2. *Pogledajmo matricu* $A \in \mathbb{R}^{n\times n}$ *zadanu na sljedeći način*

$$A = \begin{bmatrix} 1 & 3 & x \\ \frac{1}{3} & 1 & \frac{1}{5} \\ \frac{1}{x} & 5 & 1 \end{bmatrix}, \quad x > 0.$$

Uočimo, A *je pozitivna recipročna matrica. Pretpostavimo da je* A *konzistentna. Tada koristeći Definiciju 2.3 mora vrijediti* $a_{12}a_{23} = a_{13}$*, tj.* $x = a_{13} = 3 \cdot \frac{1}{5} = \frac{3}{5}$*. Tada matrica* A *poprima sljedeći oblik*

$$A = \begin{bmatrix} 1 & 3 & \frac{3}{5} \\ \frac{1}{3} & 1 & \frac{1}{5} \\ \frac{5}{3} & 5 & 1 \end{bmatrix}.$$

Uočimo da $x = \frac{3}{5}$*, kao niti* $\frac{1}{x} = \frac{5}{3}$*, nisu elementi fundamentalne ljestvice. Drugim riječima, pokazali smo da koristeći fundamentalnu ljestvicu često nije moguće postići konzistentnost.*

2.2 Metoda svojstvenog vektora

Matrica usporedbi po parovima kod AHP metode vodi ka jednadžbi $Aw = \lambda_{max}w$, gdje je $w \neq 0$. Vektor w je svojstveni vektor pridružen svojstvenoj vrijednosti λ_{max}. Manipulacijom izraza dobivamo ekvivalentan izraz $(A - \lambda_{max}I)w = 0$. Kako je w prema definiciji različit od 0 slijedi da je $A - \lambda_{max}I$ singularna, tj. da je $\det(A - \lambda_{max}I) = 0$. Dakle, traženje svojstvene vrijednosti matrice A svodi se na traženje nultočaka polinoma $k_A(\lambda) = \det(A - \lambda I)$. Polinom k_A naziva se karakteristični polinom matrice A. Vrijedi i obrat, tj. za svaki polinom može se naći matrica kojoj je taj polinom karakteristični polinom. Neka je dan polinom $p(z) = a_0 + a_1z + \ldots + a_{n-1}z^{n-1} + z^n$. Definirajmo $A \in \mathbb{C}^{n\times n}$ sa

$$A = \begin{bmatrix} 0 & & & & -a_0 \\ 1 & & & & -a_1 \\ & \ddots & & & \vdots \\ & & \ddots & & -a_{n-2} \\ & & & 1 & -a_{n-1} \end{bmatrix}.$$

Indukcijom se može pokazati da je $k_A(z) = \det(A - zI) = (-1)^n p(z)$. Zaključujemo da je problem određivanja svojstvenih vrijednosti ekvivalentan problemu određivanja nultočaka polinoma.

Teorem 2.4 (Abel-Ruffinijev teorem [14])**.** *Za svaki* $n \geq 5$ *postoji polinom* p *stupnja* n *s racionalnim koeficijentima koji ima realnu nultočku koja se ne može izraziti koristeći samo racionalne brojeve, zbrajanje, oduzimanje, množenje, dijeljenje i vađenje* n*-tog korijena.*

Kako računalo može provoditi samo operacije koje se navode u Teoremu 2.4, zaključujemo da za računanje svojstvenih vrijednosti matrice stupnja većeg ili jednakog od 5, možemo koristiti isključivo neku od iterativnih metoda. Jedna takva metoda je metoda potencija.

Teorem 2.5 (Perron-Frobeniusov teorem)**.** *Neka je $A = (a_{ij})$, $a_{ij} > 0$, $i, j = 1, \dots, n$. Tada A ima svojstvenu vrijednost $\lambda_{max} > 0$ i $\lambda_{max} > |\lambda_k|$ za sve $k \neq max$. Nadalje, postoji jedinstveni svojstveni vektor $w = [w_1, \dots, w_n]^\top$ pridružen svojstvenoj vrijednosti λ_{max} takav da je $\sum_{i=1}^{n} w_i = 1$, $w_i > 0$, $i = 1, \dots, n$.*

Dokaz Teorema 2.5 može se naći u [3, str. 524–526].

Uočimo kako za matricu alternativa vrijede pretpostavke Teorema 2.5. Dakle, svojstveni vektor w pridružen svojstvenoj vrijednosti λ_{max} matrice alternativa je pozitivan (do na množenje negativnim brojem). Vektor težina matrice alternativa je rješenje sustava

$$\begin{cases} Aw = \lambda_{max} w \\ w^\top I_{1\times n} = 1 \end{cases}$$

pri čemu je $I_{1\times n} = [\underbrace{1, \dots, 1}_{\text{n puta}}]^\top$, a n dimenzija matrice A.

2.2.1 Metoda potencija

Neka je $A \in \mathbb{R}^{n\times n}$. Općenito, ako je w svojstveni vektor matrice A, onda $Aw = \lambda w$ za $w \neq 0$, ali i $A^k w = \lambda^k w$, za sve $k \in \mathbb{N}$. Ova činjenica je temelj za metodu potencija. Pretpostavimo da je $\{w_i\}$ skup ortonormiranih svojstvenih vektora od A koji čini bazu od $\mathbb{R}^n$ sa pripadnim svojstvenim vrijednostima λ_i takvima da $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$. Neka je $v^{(0)}$ aproksimacija svojstvenog vektora od A takav da je $\|v^{(0)}\| = 1$ i $\langle v^{(0)}, w_1\rangle \neq 0$. $v^{(0)}$ možemo prikazati kao linearnu kombinaciju svojstvenih vektora od A. Za neke $c_1, \dots, c_n \in \mathbb{R}$ imamo da

$$v^{(0)} = c_1 w_1 + \dots + c_n w_n.$$

Kako $v^{(0)}$ i w_1 ne smiju biti ortogonalni vrijedi da $c_1 \neq 0$. Sada,

$$Av^{(0)} = c_1\lambda_1 w_1 + c_2\lambda_2 w_2 + \dots + c_n\lambda_n w_n$$

te

$$\begin{aligned} A^k v^{(0)} &= c_1\lambda_1^k w_1 + c_2\lambda_2^k w_2 + \dots + c_n\lambda_n^k w_n \\ &= \lambda_1^k \left(c_1 w_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k w_2 + \dots + c_n \left(\frac{\lambda_n}{\lambda_1}\right)^k w_n \right). \end{aligned}$$

Kako je pretpostavka da su svojstvene vrijednosti realne, različite i poredane u padajućem poretku prema apsolutnoj vrijednosti, slijedi da za sve $i = 2, \dots, n$

$$\lim_{k\to\infty} \left(\frac{\lambda_i}{\lambda_1}\right)^k = 0.$$

Dakle, kako se k povećava, $A^k v^{(0)}$ se približava $c_1\lambda_1^k w_1$, te za sve veće vrijednosti k

$$w_1 \approx \frac{A^k v^{(0)}}{\|A^k v^{(0)}\|}.$$

Iz prethodne rasprave konstruirajmo sljedeći algoritam za računanje svojstvenog vektora.

ulaz: Početni vektor $v^{(0)}$, $\left\|v^{(0)}\right\| = 1$
for $k \leftarrow 1$ **to** *maxiter* **do**
$w = Av^{(k-1)}$;
$v^{(k)} = \frac{w}{\|w\|}$;
$\lambda^{(k)} = {v^{(k)}}^\top A v^{(k)}$
end

Algoritam 1: Metoda potencija

U svakoj iteraciji, $v^{(k)}$ se približava svojstvenom vektoru w_1.

Prije nego iskažemo i dokažemo teorem o konvergenciji Algoritma 1., iskažimo i dokažimo pomoćnu tvrdnju koju ćemo iskoristiti u dokazu Teorema 2.7.

Teorem 2.6. *Neka je $A \in \mathbb{R}^{n\times n}$ simetrična i $x \in \mathbb{R}^n$, $x \neq 0$. Tada je x svojstveni vektor od A s pripadnom svojstvenom vrijednosti λ ako i samo ako je $\nabla r_A(x) = 0$ i $r_A(x) = \lambda$ pri čemu je $r_A(x) = \frac{\langle x, Ax\rangle}{\langle x, x\rangle}$ za $A \in \mathbb{R}^{n\times n}$ i sve $x \in \mathbb{R}^n$.*

Dokaz. Gradijent od r_A možemo izračunati na sljedeći način

$$\begin{aligned}
\nabla r_A(x) &= \left(\frac{\partial}{\partial x_i}\frac{\sum_{j,k=1}^n x_j a_{jk} x_k}{\sum_{j=1}^n x_j^2}\right)_{i=1,\ldots,n} \\
&= \left(\frac{\left(\sum_{k\neq i} a_{ik}x_k + \sum_{j\neq i} x_j a_{ji} + 2a_{ii}x_i\right)\sum_{j=1}^n x_j^2 - \sum_{j,k} x_j a_{jk} x_k \cdot 2x_i}{\left(\sum_{j=1}^n x_j^2\right)^2}\right)_{i=1,\ldots,n} \\
&= \frac{2Ax \cdot \langle x, x\rangle - 2\langle x, Ax\rangle \cdot x}{\langle x, x\rangle^2} \\
&= \frac{2}{\|x\|_2^2}(Ax - r_A(x)\cdot x).
\end{aligned}$$

Pretpostavimo da je $Ax = \lambda x$. Tada je $r_A(x) = \frac{\langle x, \lambda x\rangle}{\langle x, x\rangle} = \lambda$ i

$$\nabla r_A(x) = \frac{2}{\|x\|_2^2}(\lambda x - \lambda x) = 0.$$

S druge strane, ako je $r_A(x) = 0$, onda je $Ax - r_A(x)x = 0$, pa stoga imamo $Ax = r_A(x)\cdot x$, odnosno $\lambda = r_A(x)$. □

U praksi, korisnost Algoritma 1 ovisi o omjeru $\frac{|\lambda_2|}{|\lambda_1|}$ obzirom da on diktira brzinu konvergencije, a o tome govori Teorem 2.7.

Teorem 2.7. *Neka je $|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|$ i $\langle z^{(0)}, x_1\rangle \neq 0$. Tada postoji niz $\left(\sigma^k\right)_{k\in\mathbb{N}}$ gdje je $\sigma^{(k)} \in \{-1, +1\}$ za sve $k \in \mathbb{N}$ tako da nizovi $\left(z^{(k)}\right)$ i $\left(\lambda^{(k)}\right)$ dobiveni algoritmom metode potencija zadovoljavaju*

$$\left\|z^{(k)} - \sigma^{(k)}x_1\right\|_2 = \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^k\right) \tag{5}$$

i

$$|\lambda^{(k)} - \lambda_1| = \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{2k}\right). \tag{6}$$

Dokaz. a) Neka je $x_1, \ldots, x_n$ ortonormalan sustav svojstvenih vektora sa svojstvenim vrijednostima $\lambda_1, \ldots, \lambda_n$ i

$$z^{(0)} = \sum_{i=1}^{n} \alpha_i x_i.$$

Budući da smo pretpostavili da je $\alpha_1 = \langle z^{(0)}, x_1 \rangle \neq 0$, imamo

$$A^k z^{(0)} = \sum_{i=1}^{n} \alpha_i \lambda^k x_i = \alpha_i \lambda_1^k \left(x_1 + \sum_{i=2}^{n} \frac{\alpha_i}{\alpha_1} \left| \frac{\lambda_i}{\lambda_1} \right|^k x_i \right),$$

pa iz Pitagorinog teorema slijedi

$$\left\|A^k z^{(0)}\right\|_2^2 = \left|\alpha_1 \lambda_1^k\right|^2 \left(1 + \sum_{i=2}^{n} \left(\frac{\alpha_i}{\alpha_1}\right)^2 \left|\frac{\lambda_i}{\lambda_1}\right|^{2k} \right) \leq |\alpha_1 \lambda_1^k|^2 \left(1 + \left|\frac{\lambda_2}{\lambda_1}\right|^{2k} \sum_{i=2}^{n} \left(\frac{\alpha_i}{\alpha_1}\right)^2 \right).$$

Ako iskoristimo Taylorovu aproksimaciju $\sqrt{1+x^2} \approx 1 + \frac{x^2}{2}$, možemo zaključiti

$$\left\|A^k z^{(0)}\right\|_2 = \left|\alpha_1 \lambda_1^k\right| \left(1 + \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{2k} \right) \right).$$

Definirajmo $\sigma^{(k)} = \operatorname{sign}(\alpha_1 \lambda_1^k)$. Tada je

$$\left\| \frac{A^k z^{(0)}}{|\alpha_1 \lambda_1^k|} - \sigma^{(k)} x_1 \right\|_2^2 = \left\| \frac{A^k z^{(0)}}{\alpha_1 \lambda_1^k} - x_1 \right\|_2^2 = \sum_{i=2}^{n} \left(\frac{\alpha_i}{\alpha_1}\right)^2 \left|\frac{\lambda_i}{\lambda_1}\right|^{2k} = \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{2k} \right)$$

i stoga je

$$\begin{aligned}
\left\|z^{(k)} - \sigma^{(k)} x_1\right\|_2 &\leq \left\| \frac{A^k z^{(0)}}{\left\|A^k z^{(0)}\right\|_2} - \frac{A^k z^{(0)}}{|\alpha_1 \lambda_1^k|} \right\|_2 + \left\| \frac{A^k z^{(0)}}{|\alpha_1 \lambda_1^k|} - \sigma^{(k)} x_1 \right\|_2 \\
&\leq \left\| \frac{A^k z^{(0)}}{|\alpha_1 \lambda_1^k|} \right\| \cdot \left\| \frac{1}{1 + \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{2k} \right)} - 1 \right\| + \left\| \frac{A^k z^{(0)}}{|\alpha_1 \lambda_1^k|} - \sigma^{(k)} x_1 \right\|_2 \\
&= \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{2k} \right) + \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{k} \right) = \mathcal{O}\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{k} \right)
\end{aligned} \tag{7}$$

što dokazuje tvrdnju (5).

b) Iz Teorema 2.6 znamo da je $r_A(\sigma^{(k)} x_1) = \lambda_1$ i $\nabla r_A(\sigma^{(k)} x_1) = 0$. Taylorov razvoj r_A oko $\sigma^{(k)} x_1$ daje nam

$$\begin{aligned}
r_A(z) &= r_A(\sigma^{(k)} x_1) + \langle \nabla r_A(\sigma^{(k)} x_1, z - \sigma^{(k)} x_1 \rangle + \mathcal{O}\left(\left\|z - \sigma^{(k)} x_1\right\|_2^2 \right) \\
&= \lambda_1 + 0 + \mathcal{O}\left(\left\|z - \sigma^{(k)} x_1\right\|_2^2 \right).
\end{aligned}$$

Koristeći (7) slijedi

$$|\lambda^{(k)} - \lambda_1| = |r_A(z^{(k)}) - \lambda_1| = O\left(\left\|z^{(k)} - \sigma^{(k)}x_1\right\|_2^2\right) = O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{2k}\right)$$

čime smo dokazali relaciju (6). □

Algoritam 1 se zaustavlja kada dostigne određeni broj koraka, u ovom slučaju $maxit$. Kako algoritam radi iterativno, pogrešku možemo samo procijeniti te ukoliko je točnost koju zahtijevamo veća ili jednaka onoj postignutoj, možemo zaustaviti algoritam i prije nego što je broj koraka jednak unaprijed definiranom ograničenju $maxit$. U tu svrhu iskažimo i dokažimo sljedeći teorem.

Teorem 2.8. *Neka je $A \in \mathbb{R}^{n\times n}$ simetrična matrica, $\sigma(A)$ spektar od A te neka je $\widehat{\lambda}$ aproksimacija svojstvene vrijednosti λ i $\widehat{x}$ aproksimacija svojstvenog vektora x matrice A. Tada je*

$$\min_{\lambda_i\in\sigma(A)} |\widehat{\lambda} - \lambda_i| \leq \frac{\left\|A\widehat{x} - \widehat{\lambda}\widehat{x}\right\|_2}{\|\widehat{x}\|_2}, \quad x \neq 0, \tag{8}$$

Dokaz. Kako je A simetrična, postoji ortonormalan sustav svojstvenih vektora $\{u_k\}$ koji je baza za $\mathbb{R}^n$. Označimo $\widehat{r} = A\widehat{x} - \widehat{\lambda}\widehat{x}$, $\widehat{x} \neq 0$. Tada $\widehat{x} = \sum_{i=1}^n \alpha_i u_i$ gdje je $\alpha_i = u_i^\top \widehat{x}$ te stoga $\widehat{r} = \sum_{i=1}^n \alpha_i \left(\lambda_i - \widehat{\lambda}\right) u_i$. Iz toga slijedi

$$\left(\frac{\|\widehat{r}\|_2}{\|\widehat{x}\|_2}\right)^2 = \sum_{i=1}^n \beta_i \left(\lambda_i - \widehat{\lambda}\right)^2, \quad \text{gdje je } \beta_i = \frac{|\alpha_k|^2}{\sum_{j=1}^n \left|\alpha_j\right|^2}. \tag{9}$$

Kako je $\sum_{i=1}^n \beta_i = 1$, nejednakost (8) slijedi iz (9). □

Za zaustavni kriterij koristit ćemo relativnu pogrešku danu izrazom

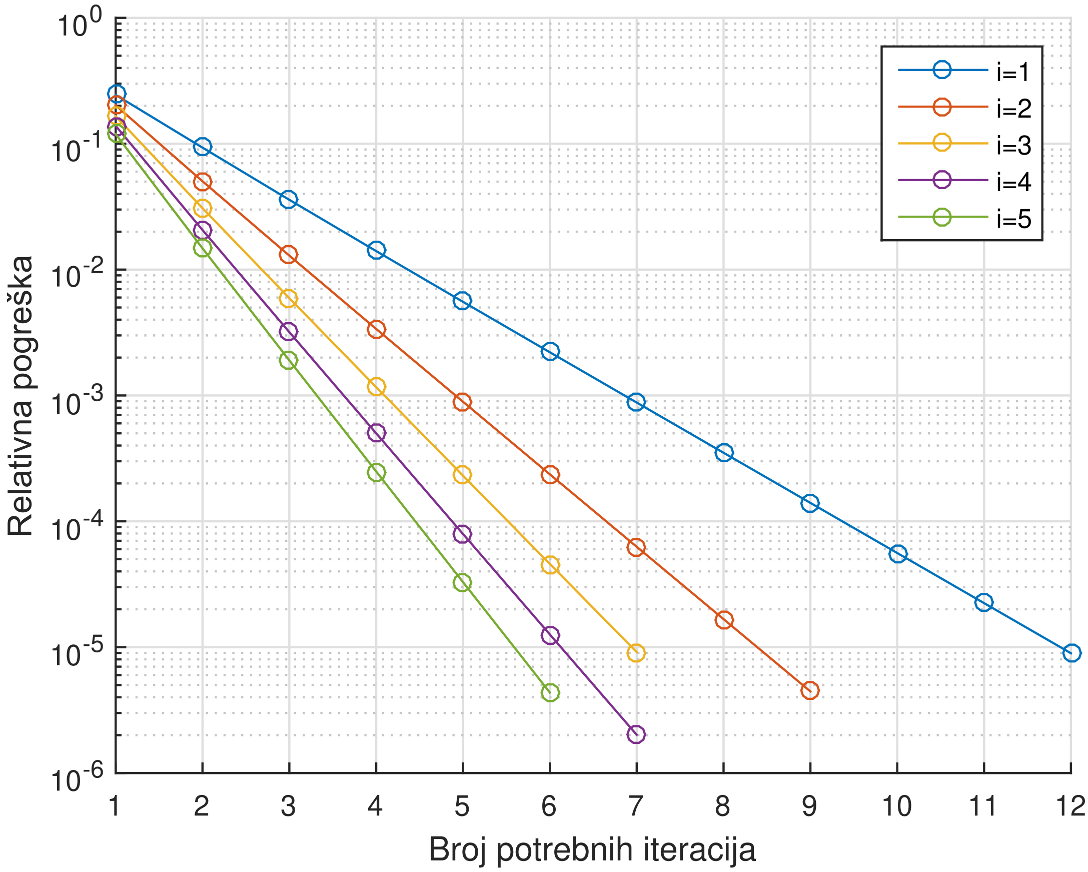
$$\frac{\|Av_k - \lambda v_k\|_2}{|\lambda_k|}, \tag{10}$$

pri čemu je v_k aproksimacija svojstvenog vektora dobivena metodom potencija u k-tom koraku, λ_k aproksimacija svojstvene vrijednosti u k-tom koraku te A matrica s ulaza. Uočimo da je za implementaciju Algoritma 1 jedino potrebna podrutina koja može računati umnožak matrice i vektora oblika Av. Nije potrebno spremati A u polje $n \times n$. Upravo zbog toga, algoritam može biti od značaja kada je A velika i rijetko popunjena te kada je značajna udaljenost između $|\lambda_1|$ i $|\lambda_2|$.

Primjer 2.3. *Neka je zadana matrica $C \in \mathbb{R}^{5\times 5}$*

$$C = \begin{bmatrix} 9 & 8 & -1 & 3 & 5 \\ -8 & -9 & -8 & -9 & -10 \\ 5 & -5 & 10 & -9 & -9 \\ 5 & -9 & -4 & 6 & 4 \\ 1 & -1 & -4 & 9 & 2 \end{bmatrix}$$

te neka je matrica $Q \in \mathbb{R}^{5\times 5}$ dobivena QR faktorizacijom matrice C. Nadalje, neka su dijagonalne matrice $\Lambda_1, \Lambda_2, \ldots, \Lambda_5 \in \mathbb{R}^{5\times 5}$ takve da $\Lambda_i = \operatorname{diag}(x_i, 8, 6, 4, 2)$, $x_i = 10(1+i)$, $i = 1, \ldots, 5$ i *matrice A_i zadane s $A_i = Q^{-1}\Lambda_i Q$, $i = 1, \dots, 5$. Matrice A_i i Λ_i su slične pa imaju jednake karakteristične polinome, a samim time i jednake svojstvene vrijednosti [1]. Dakle, najveća svojstvena vrijednost matrice A_i, $\lambda_{max}^{(i)}$ jednaka je $10(1+i)$, $i = 1, \dots, 5$. Uočimo, kako je Q ortogonalna, vrijedi $Q^{-1} = Q^\top$ pa $A_i = Q^\top \Lambda_i Q$, $i = 1, \dots, 5$. Omjer $\frac{\lambda_2}{\lambda_1}$ za A_i jednak je $\frac{8}{10(1+i)}$ za $i = 1, \dots, 5$. Prema Teoremu 2.7, metoda potencija najbrže će konvergirati za matricu A_5, a najsporije za A_1. Na Slici 3 prikazan je graf funkcije relativne pogreške ovisne o koraku Algoritma 1 pokrenutog za svaku od matrica A_i pri čemu je zaustavni kriterij, relativna pogreška dana izrazom (10), postavljen na 10^{-5}.*



Slika 3: Grafički prikaz pogreške

2.3 Indikatori konzistentnosti

Ideja konzistentnosti najbolje se može ilustrirati jednostavnim primjerom: Ako automobil preferiramo dvostruko više od vlaka, a vlak trostruko više od zrakoplova, koliko preferiramo automobil u odnosu na zrakoplov? Ako bismo željeli biti potpuno matematički konzistentni, odgovor bi bio šest. Dakle, automobil preferiramo šesterostruko u odnosu na zrakoplov. Međutim, to nije uvijek moguće postići što zbog same ljestvice (Primjer 2.2), što zbog prirode misaonog procesa pri prosudbi alternativa. No, doza nekonzistentnosti je dozvoljena i očekivana u AHP metodi [5].

Pokažimo da je tranzitivnost nužan uvjet konzistentnosti. Uzmimo recipročnu matricu A. Kako je A recipročna, bez smanjenja općenitosti možemo pretpostaviti $a_{ij} \geq 1$ za sve

$i \le j$. Za $i \le j \implies a_{ij} \ge 1$ i $j \le k \implies a_{jk} \ge 1$, koristeći Definiciju 2.3 slijedi

$$a_{ij}a_{ik} = a_{ik} \ge 1,$$

te $i \le k$. Zaključujemo da je tranzitivnost nužan uvjet konzistentnosti.

Očito nije i dovoljan jer npr. za $a_{ij} = 2$, $a_{jk} = 3$ i $a_{ik} = 4$ iako tranzitivno, nije i konzistentno. AHP metoda ne zahtijeva da prosudbe budu konzistentne pa čak ni tranzitivne [11]. Postavlja se logično pitanje: Ako su prosudbe potpuno nasumične, koliku će konzistentnost AHP metoda imati? Konzistentnost matrice takvih nasumičnih prosudbi trebala bi biti puno manja nego konzistentnost matrice informiranih prosudbi. Mjera konzistentnosti se može koristiti za uspoređivanje i evaluaciju razine konzistentnosti informiranih prosudbi. Korisnik AHP metode može biti savršeno konzistentan ako želi sačuvati relaciju $a_{ij}a_{jk} = a_{ik}$ koristeći redundantne prosudbe iz relacija između prethodno danih prosudbi.

Mala perturbacija konzistentne matrice ostavlja najveću svojstvenu vrijednost λ_{max} blizu n. Ostale svojstvene vrijednosti pomaknute su blizu nule. Odabir perturbacije koja će prikladno opisati učinak nekonzistentnosti na svojstveni vektor ovisi o tome koji je psihološki proces uključen u usporedbu alternativa po parovima. Pretpostavljamo da se sve značajne perturbacije mogu svesti na općeniti oblik

$$a_{ij} = \frac{w_i}{w_j}\epsilon_{ij}.$$

Pa npr. za dodavanje broja svakom elementu imamo

$$\frac{w_i}{w_j} + \alpha_{ij} = \frac{w_i}{w_j}\left(1 + \frac{w_j}{w_i}\alpha_{ij}\right) = \frac{w_i}{w_j}\epsilon_{ij}.$$

Konzistentnost imamo kada je $\epsilon_{ij} = 1$.

Izvedimo sada nekoliko elementarnih, ali ključnih rezultata o konzistentnim matricama. Raspišemo li sustav $Aw = \lambda_{max}w$, za i-tu jednadžbu imamo

$$\lambda_{max} = \sum_{j=1}^{n} a_{ij}\frac{w_j}{w_i}.$$

Definirajmo

$$\mu = -\frac{1}{n-1}\sum_{i=2}^{n}\lambda_i$$

te uočimo da $\sum_{i=1}^{n}\lambda_i = n$ jer je $n = \mathrm{tr}(A)$. Ova činjenica implicira da je

$$\mu = \frac{\lambda_{max} - n}{n-1}, \quad \lambda_{max} = \lambda_1,$$

a kako

$$\lambda_{max} - 1 = \sum_{j \ne i} a_{ij}\frac{w_j}{w_i}$$

imamo

$$n\lambda_{max} - n = \sum_{1 \le i < j \le n}\left(a_{ij}\frac{w_j}{w_i} + a_{ji}\frac{w_i}{w_j}\right)$$

i dakle

$$\mu = \frac{\lambda_{max} - n}{n-1} = \frac{1}{n-1} - \frac{n}{n-1} + \frac{1}{n(n-1)} \sum_{1 \le i \le j \le n} \left(a_{ij}\frac{w_j}{w_i} + a_{ji}\frac{w_i}{w_j} \right).$$

Supstitucijom, $a_{ij} = \frac{w_i}{w_j}\epsilon_{ij}$, $\epsilon_{ij} > 0$ dobivamo sljedeću jednadžbu

$$\mu = -1 + \frac{1}{n(n-1)} \sum_{1 \le i \le j \le n} \left(\epsilon_{ij} + \frac{1}{\epsilon_{ij}} \right).$$

Uočimo da kada $\epsilon \to 1$, tj. kada se približavamo konzistentnosti, onda $\mu \to 0$. Kako je $\epsilon_{ij} + \frac{1}{\epsilon_{ij}}$ konveksna funkcija koja ima minimum u $\epsilon_{ij} = 1$ te kako je suma konveksnih funkcija konveksna, slijedi da je μ konveksna funkcija. Dakle, hoće li μ biti mal ili velik ovisi o tome je li ϵ_{ij} blizu ili daleko od 1.

Zaključno, ako zapišemo $\epsilon_{ij} = 1 + \delta_{ij}$, $\delta > -1$ imamo

$$\mu = \frac{1}{n(n-1)} \sum_{1 \le i \le j \le n} \left(\delta_{ij}^2 - \frac{\delta_{ij}^3}{1 + \delta_{ij}} \right)$$

Želimo da matrica teži ka konzistentnosti, tj. da μ bude što bliže nuli što za posljedicu ima razmjernu blizinu svojstvene vrijednosti λ_{max} dimenziji n matrice.

Definicija 2.4 (Indeks konzistentnosti)**.** Indeks konzistentnosti (*eng.* consistency index) matrice usporedbi dan je izrazom

$$C.I. = \frac{\lambda_{max} - n}{n-1}. \tag{11}$$

Ernest H. Forman[5] je izračunao prosječne indekse konzistentnosti (*eng.* random consistency index, *kr.* R.I.) za $n \le 7$ koristeći veliki uzorak matrica usporedbi. To je učinio na način da je generirao slučajnim odabirom recipročne matrice koristeći fundamentalnu ljestvicu $^1/_9, {}^1/_8, \ldots, 1, \ldots, 8, 9$ te računao aritmetičku sredinu njihovih najvećih svojstvenih vrijednosti. Na isti način moguće je izračunati i za matrice dimenzije veće od 7. Sljedećom tablicom dani su indeksi nasumične konzistentnosti za $n \le 13$.

n	1	2	3	4	5	6	7	8	9	10	11	12	13
$R.I.$	0	0	.52	.89	1.11	1.25	1.35	1.4	1.15	1.49	1.51	1.54	1.56

Tablica 2: Indeksi nasumične konzistentnosti [11]

Uočimo da je $R.I. = 0$ za $n = 1, 2$. Naime, pozitivna recipročna matrica dimenzija 1 je trivijalno konzistentna. O konzistentnosti matrice dimenzija 2×2 govori nam sljedeći teorem.

Teorem 2.9. *Neka je $A \in \mathbb{R}^{2\times 2}$ pozitivna recipročna matrica. Tada je A konzistentna.*

Dokaz. Neka je $A \in \mathbb{R}^{2\times 2}$ pozitivna recipročna matrica. Tada je A oblika

$$A = \begin{bmatrix} 1 & x \\ \frac{1}{x} & 1 \end{bmatrix}, \quad x > 0.$$

[5] http://professorforman.com/

Sada je karakteristični polinom

$$k_A(\lambda) = \det(A - \lambda I) = \det\left(\begin{bmatrix} 1-\lambda & x \\ \frac{1}{x} & 1-\lambda \end{bmatrix}\right) = \lambda(\lambda - 2)$$

te najveća svojstvena vrijednost $\lambda_{max} = 2$. Kako je λ_{max} jednako dimenziji matrice, prema Teoremu 2.3 slijedi da je A konzistentna. □

Definicija 2.5 (Omjer konzistentnosti)**.** Omjer konzistentnosti (*eng.* consistency ratio) definira se kao omjer indeksa konzistentnosti i indeksa nasumične konzistentnosti

$$C.R. = \frac{C.I.}{R.I.}. \tag{12}$$

Nekonzistentnost može biti znak da je potrebno prilagoditi prosudbe. Međutim, prilagodbe ne bi smjele biti toliko velike da sama prosudba gubi smisao, ali niti premala da nema učinka. Na ljestvici od 0 do 1, ukupna konzistentnost bi trebala iznositi oko 0.1 tj. oko 10% [8]. Obzirom da $R.I.$ ovisi o n, granicu prihvatljive konzistetnosti dobivamo tako da svaku $R.I.$ pomnožimo s 0.1. Konvencionalno se uzima da je prihvatljivo $C.R. = .05$ za $n = 3$, $C.R. = .08$ za $n = 4$ te $C.R. = .1$ za sve $n \geq 5$ [11].

Primjer 2.4. *Odredimo indeks konzistentnosti i omjer konzistentnosti za matricu usporedbi iz Primjera 2.1. Najveća svojstvena vrijednost, λ_{max} matrice*

$$\begin{bmatrix} 1 & 7 & 8 \\ \frac{1}{7} & 1 & 3 \\ \frac{1}{8} & \frac{1}{3} & 1 \end{bmatrix}$$

je 3.1044*. Tada je indeks konzistentnosti prema (11) jednak*

$$C.I. = \frac{3.1044 - 3}{3 - 1} = .0522,$$

a omjer konzistentnosti prema (12) jednak

$$C.R. = \frac{.0522}{.52} = .1004.$$

Kako je C.R. veći od .05 *zaključujemo da konzistentnost nije zadovoljavajuća te je potrebno prilagoditi prosudbe.*

3 Implementacija koristeći grafičko korisničko sučelje MATLAB®-a

U ovom dijelu rada posvetit ćemo se programskom rješenju grafičkog sučelja AHP metode koristeći MATLAB®[6].

3.1 Korisničko sučelje MATLAB®-a

Korisničko sučelje (*eng.* user interface, *kr.* UI) je grafički prikaz jednog ili više prozora koji sadrže kontrole koje se nazivaju komponente. Komponente omogućavaju korisniku interakciju. Korisnik ne mora pisati naredbe kako bi izvršio zadatke. Za razliku od programiranja, za interakciju nije potrebno nikakvo predznanje osim osnovnog služenja računalom.

Neke od UI komponenata su izbornici, alatne trake, potisni gumbi, klizači itd. Korisnička sučelja kreirana koristeći MATLAB® mogu vršiti izračune, čitati i pisati u datoteke, komunicirati s drugim korisničkim sučeljima i prikazivati podatke tablično i grafički.

Uobičajeno, UI čeka da korisnik započne interakciju, a zatim odgovara na svaku akciju posebno. Svaka UI komponenta, a i sami UI, ima jednu ili više *callback* funkciju. Iz samog naziva „callback" može se naslutiti priroda takve funkcije. Naime, callback funkcija poziva natrag MATLAB da izvrši zadatak. Akcija korisnika kao npr. klik na gumb, je okidač za izvršavanje svake callback funkcije. UI tada odgovara na ove događaje (*eng.* event). Kreator korisničkog sučelja u callback funkcijama definira što komponente rade kako bi rukovodile događajima.

Ovakav način programiranja se često naziva *event-driven* programiranje. U event-driven programiranju, izvršavanje callback funkcija je asinkrono, tj. događaji koji nisu dio softvera pokreću njihovo izvršavanje. U slučaju MATLAB UI-a, većina događaja su korisnikova interakcija s UI-em, ali UI može odgovoriti drugim vrstama događaja kao npr. stvaranje datoteke ili spajanje uređaja na računalo. Moguće je pisati callback funkcije na dva različita načina

- kao MATLAB funkcije spremljene u datoteku i
- kao *stringove* koji sadrže MATLAB izraze ili naredbe (npr. `'R=qr(A);'`).

U implementaciji AHP metode koristili smo isključivo callback funkcije spremljene u zasebne datoteke.

3.2 Implementacija

Prije nego pređemo na samu implementaciju, u primjeru definirajmo problem koji ćemo unositi u program.

Primjer 3.1. *Pretpostavimo da časopis posvećen automobilskoj industriji bira automobil godine. U tu svrhu odabiru tri najprodavanija modela u toj godini te žele odabrati najbolji. Kako im je cijena nevažna stavka, odlučili su se na tri kriterija po kojima će uspoređivati modele: sigurnost (zračni jastuci, ABS[7], ESP[8]), udobnost (kožna sjedala, električni podizači stakala, atomatska klima) i stil (spušteno podvozje, moderni naplatci, kromirani detalji). Kriterije su procijenili na sljedeći način:*

[6] `https://www.mathworks.com/products/matlab.html`
[7] Sustav protiv blokiranja kotača vozila
[8] Elektronički sustav za sprječavanje zanošenja vozila

- *sigurnost umjereno važna u odnosu na udobnost,*
- *sigurnost jako važna u odnosu na stil i*
- *udobnost umjereno važna u odnosu na stil.*

Za svaku od alternativa iznijeli su nekoliko faktora za svaki kriterij:

- *Automobil A: Ovaj automobil ima najveći broj zračnih jastuka i ABS, ali nema ESP. Kožna sjedala su dio standardne opreme kao i električni podizači stakala. Ima moderne naplatke, spušteno podvozje i kromirane detalje.*
- *Automobil B: Ovaj automobil ima ABS i ESP. Paket udobnosti sadrži sve kao i kod A uz dodatno automatsku klimu. Ima samo spušteno podvozje.*
- *Automobil C: Ovaj automobil ima najmanji broj zračnih jastuka te ESP, ali nema ABS. Ima jednak paket udobnosti kao A. Nema stilskih značajki.*

Koristeći fundamentalnu ljestvicu, usporedbom po parovima obzirom na kriterije sigurnost, udobnost i stil dobili su procjene koje su prikazane u Tablici 3.

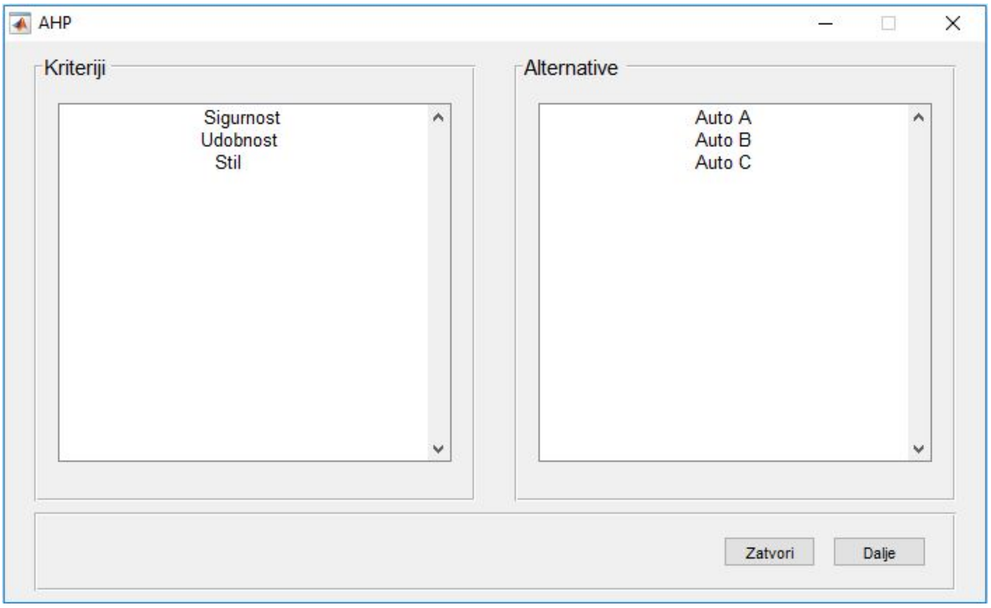
Sigurnost	*A umjereno preferiran u odnosu na B*
	A vrlo jako preferiran u odnosu na C
	B jako preferiran u odnosu na C
Udobnost	*A jednako preferiran u odnosu na C*
	B jako preferiran u odnosu na A
	B jako preferiran u odnosu na C
Stil	*A jako preferiran u odnosu na B*
	A ekstremno preferiran u odnosu na C
	B umjereno preferiran u odnosu na C

Tablica 3: Usporedbe po parovima alternativa u odnosu na kriterije

Želimo pri pokretanju programa početni prozor u koji ćemo unositi nazive kriterija i alternativa. Kreiranje prozora u koji možemo dodavati UI komponente postiže se korištenjem funkcije `figure()`. Funkcija `figure()` može primiti nekoliko parametara kao npr. *Name* za naziv prozora, *Color* za boju prozora, *Position* za poziciju, *Units* za mjernu jedinicu veličine itd. Za argument *Position* koristili smo vrijednost [200 200 700 400], a vrijednosti unutar vektora označavaju redom udaljenost od lijevog ruba ekrana, udaljenost od donjeg ruba ekrana, širinu i visinu samog prozora.

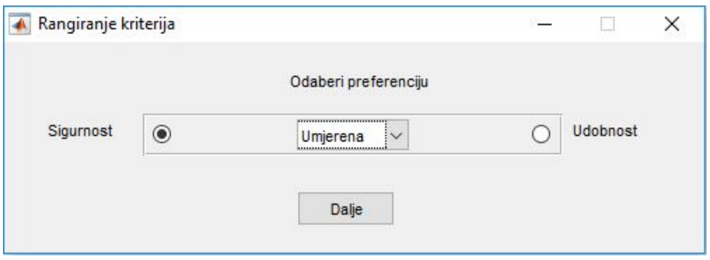
Za unos alternativa i kriterija potrebni su posebni objekti koji su nalijepljeni na prozor. Za unos teksta koristit ćemo MATLAB-ov generički objekt `UIControl()`. On ovisno o parametru `Style` može biti potisni gumb, naizmjenični gumb, okvir za izbor, uređivač teksta, tekst, lista itd. Za potrebe unosa naziva kriterija i alternativa koristili smo uređivač teksta, tj. *edit*. Callback funkcija poziva funkciju iz datoteke koja provjerava jesu li barem dva kriterija i dvije alternative unesene. Ukoliko ima manje od dva unosa, callback funkcija poziva dijaloški okvir pozivom ugrađenje MATLAB funkcije `errordlg()` s porukom upozorenja. U ovom slučaju broj redaka u unosu jednak je broju kriterija odnosno alternativa.

Na istom prozoru još su dva *UIControl* objekta. To su gumbi *Zatvori* koji zatvara i izlazi iz programa te gumb *Dalje* koji ide na novi prozor u kojemu se rangira kriterij. Potisni gumb ima za argument *Style* vrijednost *pushbutton*. Pritiskom na gumb *Zatvori* poziva se callback funkcija `btn_close_callback`, dok se pritiskom na gumb *Dalje* poziva callback funkcija `btn_next_callback` te joj se prosljeđuje glavni prozor, te uneseni nazivi kriterija i alternativa. Na Slici 4 prikazan je početni prozor za unos naziva kriterija i alternativa.



Slika 4: Prozor za unos kriterija i alternativa

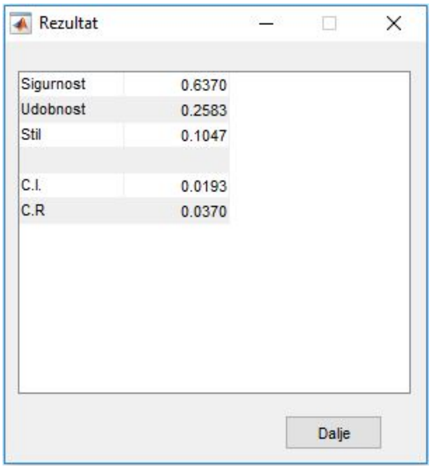
Pritiskom na gumb *Dalje* otvara se novi prozor za rangiranje kriterija. Prozor je stvoren na sličan način kao i na prozoru unosa. Nazivi kriterija dobiveni su pomoću *UIControl* objekta s argumentom *Style* vrijednosti *text*. Odabir elementa fundamentalne ljestvice izveden je na način da se odabire između ponuđenih koristeći *UIcontrol* argumenta *Style* vrijednosti *popup*. Odabirom intenziteta poziva se callback funkcija `{@put_in_cr_matrix,ii,jj}` koja unosi odabranu vrijednost u matricu. Obzirom da su ponuđene vrijednosti bez recipročnih vrijednosti potrebno je odrediti koji kriterij je preferiran. To smo postigli korištenjem objekta `uibuttongroup` na koji su nalijepljena dva radio gumba. Pritiskom na gumb pridružen jednom kriteriju, isključuje se gumb pridružen drugom kriteriju. *uibuttongroup* poziva callback funkciju `{@swap_pref_alt, cr, ii, jj}` koja u matricu unosi recipročnu vrijednost ukoliko je preferiran drugi kriterij. Dvostrukom *for* petljom osigurava se da se sve kombinacije parova bez ponavljanja kriterija rangiraju. Pritiskom na gumb *Dalje* rangira se sljedeći kriterij. Na Slici 5 prikazan je prozor za rangiranje jednog od kriterija. Kada se završi rangiranje kriterija, slijedi rangiranje alternativa koje je izvedeno na gotovo identičan način.



Slika 5: Prozor za rangiranje kriterija

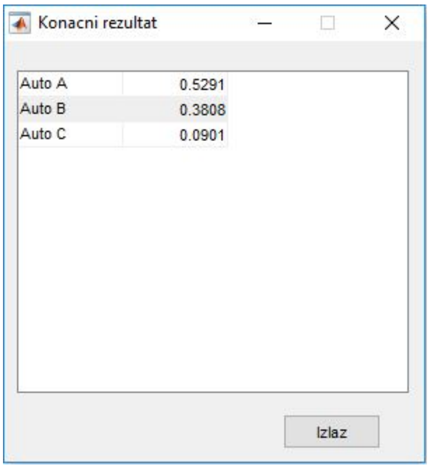
Nakon rangiranja posljednje alternative, callback funkcija poziva funkciju `powermethod()`

koja koristeći metodu potencija računa svojstveni vektor i svojstvene vrijednosti matrice rangiranih kriterija. Pritiskom na gumb *Dalje*, prikazuju se nazivi kriterija i pridružene težine te indeks i omjer konzistentnosti. Za prikaz rezultata korišten je dijaloški okvir objekta `dialog`, te nalijepljenog objekta `uitable`. Na Slici 6 prikazan je prozor rezultata rangiranja kriterija.



Slika 6: Rezultat rangiranja kriterija s pripadnim indeksima konzistentnosti

Na gotovo jednak način dobiveni su i rezultati za procjene relativne važnosti alternativa i pripadnih indeksa konzistentnosti za svaki od kriterija. Posljednji prozor prikazuje globalni vektor težina alternativa koje su dobivene kao umnožak matrice čiji su stupci vektori težina matrice alternativa u odnosu na svaki od kriterija i vektora težina matrice rangiranih kriterija. Na Slici 7 prikazan je posljednji prozor programa koji izlistava nazive alternativa i pripadane težine.



Slika 7: Prozor globalnog rješenja hijerarhije

Sa Slike 7 iščitavamo vektor težina globalnog rješenja problema iz Primjera 3.1. Uočimo kako je automobil A najbolji prema traženim kriterijima, a slijede ga redom automobili B i C.

4 Primjer: odabir kuće koristeći AHP metodu

Kako bismo ilustrirali AHP metodu razmotrit ćemo sljedeći primjer. Pretpostavimo da obitelj prosječnih primanja želi kupiti kuću. Moraju izabrati između tri alternative. Obitelj je identificirala osam faktora koji utječu na odabir kuće. Ti faktori se mogu podijeliti u tri kategorije: ekonomski, geografski i fizički. Iako se mogu uzimati relativne procjene tih kategorija, obitelj želi prioritizirati svih osam kriterija bez obzira kojoj kategoriji pripadaju. Koristeći AHP metodu, prvi korak je dekompozicija, tj. sktrukturiranje problema u hijerarhiju koja je vidljiva na Slici 8.

Slika 8: Dekompozicija problema u hijerarhiju

Na prvom (gornjem) nivou je ukupni cilj *Zadovoljstvo kućom*. Na drugom nivou je osam kriterija koji doprinose cilju i na trećem (donjem) su tri kuće kandidata koje će biti procijenjene u terminima kriterija drugog nivoa. Obitelj uzima u obzir osam kriterija po kojima će rangirati:

1. Veličina kuće: Prostor za ostavu, broj soba, veličina soba, ukupna površina kuće.

2. Transport: Udaljenost i dostupnost autobusnih stajališta.

3. Susjedstvo: Prometnost, sigurnost, porezi, stanje okolnih zgrada.

4. Starost kuće.

5. Veličina okućnice: Vanjski prostor oko kuće uključujući i dijeljeni prostor sa susjedima.

6. Moderni kućanski aparati: Perilica posuđa, klima uređaj, alarmni sustav.

7. Opće stanje: Potrebno ulaganje, stanje zidova, sagova, zavjesa, električnih instalacija.

8. Financiranje: Dostupnost kredita bilo od prodavatelja, banke ili neke druge vrste posudbe novca.

Idući korak je komparativna procjena kriterija. Elementi na drugom nivou su aranžirani u matricu i obitelj koja kupuje kuću procjenjuje relativnu važnosti elemenata u odnosu na ukupni cilj *Zadovoljstvo kućom*. Matrica usporedbi po parovima zajedno s rezultantnim vektorom dana je u Tablici 4.

	1	2	3	4	5	6	7	8	Vektor težina
1	1	5	3	7	5	7	$1/3$	$1/3$	.179
2	$1/5$	1	$1/3$	5	3	3	$1/5$	$1/7$	.054
3	$1/3$	3	1	5	3	3	7	$1/5$	.204
4	$1/7$	$1/5$	$1/5$	1	$1/3$	$1/5$	$1/7$	$1/9$	.017
5	$1/5$	$1/3$	$1/3$	3	1	$1/3$	$1/5$	$1/5$	.031
6	$1/7$	$1/3$	$1/3$	5	3	1	$1/5$	$1/5$	.043
7	3	5	$1/7$	7	5	5	1	1	.179
8	3	7	5	9	5	5	1	1	.292

$C.I = .289, C.R. = .207$

Tablica 4: Matrica usporedbi po parovima kriterija

Procjene su unesene koristeći fundamentalnu ljestvicu. Vektor prioriteta je svojstveni vektor pridružen najvećoj svojstvenoj vrijednosti. Prioriteti su jedinstveni do na množenje pozitivnom konstantom, ali ako zahtijevamo da su u sumi 1, onda su uvijek jedinstveni. U ovom slučaju financiranje ima najveću važnost s doprinosom od 33%

Sada prelazimo na usporedbe po parovima kuća na donjem nivou, uspoređujući svaku sa svakom koliko je jedna bolja od druge u odnosu na svaki od kriterija. Imamo 8 matrica dimenzija 3×3 obzirom da je osam elemenata na drugom nivou i tri kuće koje treba usporediti za svaki element. Zbog lakšeg razumijevanja matrica ukratko opišimo svaku od procjena:

- Kuća A: Ova kuća je najveća od svih promatranih. Nalazi se u dobrom susjedstvu s malo prometa i niskim porezima. Okućnica je usporedivo veća nego kod kuće B i C. Međutim, opće stanje nije jako dobro te zahtijeva čišćenje i ličenje. Kreditiranje je moguće samo u banci uz visoke kamate.

- Kuća B: Ova kuća je manja nego kuća A i nije blizu autobusnomj stajalištu. Susjedstvo odaje osjećaj nesigurnosti zbog loše prometne regulacije. Okućnica je mala i nedostaju osnovni moderni kućanski aparati. U drugu ruku, opće stanje je jako dobro. Moguće je dobiti dobar kredit uz nisku kamatnu stopu. Nekoliko sličnih kuća nalazi se u susjedstvu.

- Kuća C: Ova kuća je jako mala, ali ima nekoliko modernih kućanskih aparata. U susjedstvu su visoki porezi, ali je u dobrom stanju i djeluje sigurno. Okućnica je veća nego kod kuće C, ali puno manja nego kod koće A. Opće stanje je dobro te ima lijep sag i zavjese. Financiranje je bolje nego kod kuće A, ali lošije nego kod kuće B.

	A	B	C	Vektor težina	Idealizirane težine
Veličina kuće[a]					
A	1	7	7	.766	1.000
B	$1/7$	1	3	.158	.206
C	$1/7$	$1/3$	1	.076	.099
Transport[b]					
A	1	7	$1/5$	.227	.314
B	$1/7$	1	$1/9$	.051	.071
C	5	9	1	.722	1.000
Susjedstvo[c]					
A	1	7	5	.731	1.000
B	$1/7$	1	$1/3$	.081	.111
C	$1/5$	3	1	.188	.257
Starost kuće[d]					
A	1	$1/7$	$1/9$	.056	.090
B	7	1	3	.620	1.000
C	9	$1/3$	1	.324	.523
Veličina okućnice[e]					
A	1	5	3	.637	1.000
B	$1/5$	1	$1/3$	.105	.165
C	$1/3$	3	1	.258	.405
Moderni kućanski aparati[f]					
A	1	7	5	.715	1.000
B	$1/7$	1	$1/5$	.067	.094
C	$1/5$	5	1	.219	.306
Opće stanje[g]					
A	1	$1/3$	$1/3$	.143	.333
B	3	1	1	.429	1.000
C	3	1	1	.429	1.000
Financiranje[h]					
A	1	$1/7$	$1/5$	.072	.111
B	7	1	3	.650	1.000
C	5	$1/3$	1	.279	.430

[a] $C.I. = .068$, $C.R. = .130$
[b] $C.I. = .104$, $C.R. = .201$
[c] $C.I. = .032$, $C.R. = .062$
[d] $C.I. = .103$, $C.R. = .198$
[e] $C.I. = .019$, $C.R. = .037$
[f] $C.I. = .091$, $C.R. = .176$
[g] $C.I. = .000$, $C.R. = .000$
[h] $C.I. = .032$, $C.R. = .062$

Tablica 5: Usporedbe po parovima za drugi nivo hijerarhije

U Tablici 5 dane su matrice kuća i njihovi lokalni prioriteti u odnosu na elemente drugog nivoa. Posljednji korak je sinteza prioriteta, a moguća je na dva načina: distributivni i idealizirani. U distributivnom načinu matrica čiji su stupci vektori težina matrica relativnih procjena alternativa, tj. svojstveni vektori pomnožena vektorom težina matrice procjena kriterija daju globalno rješenje. Globalni vektor prioriteta koristeći distributivni način dan je sljedećim umnoškom

$$\begin{bmatrix} .766 & .227 & .731 & .056 & .637 & .715 & .143 & .072 \\ .158 & .051 & .081 & .620 & .105 & .067 & .429 & .650 \\ .076 & .722 & .188 & .324 & .258 & .219 & .429 & .279 \end{bmatrix} \begin{bmatrix} .179 \\ .054 \\ .204 \\ .017 \\ .031 \\ .043 \\ .179 \\ .292 \end{bmatrix} = \begin{bmatrix} .397 \\ .331 \\ .273 \end{bmatrix}.$$

Drugi način sinteze prioriteta je idealizirani način. U ovom slučaju vektor težina se prvo podijeli najvećim elementom. Tada alternativa kojoj pripada najveći element postaje idealna i dobiva vrijednost 1. Dalje se nastavlja na isti način kao i kod distributivnog. Globalni vektor prioriteta koristeći idealizirani način dan je sljedećim umnoškom

$$\begin{bmatrix} 1.000 & .314 & 1.000 & .090 & 1.000 & 1.000 & .333 & .111 \\ .206 & .071 & .111 & 1.000 & .165 & .094 & 1.000 & 1.000 \\ .099 & 1.000 & .257 & .523 & .405 & .306 & 1.000 & .430 \end{bmatrix} \begin{bmatrix} .179 \\ .054 \\ .204 \\ .017 \\ .031 \\ .043 \\ .179 \\ .292 \end{bmatrix} = \begin{bmatrix} .568 \\ .561 \\ .463 \end{bmatrix}.$$

Kuća A je preferirana koristeći oba načina sinteze prioriteta. U slučaju da distributivni način preferira kuću A, a idealizirani kuću B, rješenje bismo interpretirali na sljedeći način. Kuća A je preferirana u slučaju da slične kuće B u susjedstvu uzimamo u obzir te se stoga koristi distributivni način. Kada se dodaje novi broj alternativa (u ovom slučaju kuću B), a želimo sačuvati prioritete, onda koristimo idealizirani način pa je preferirana kuća B [9]. Npr. slučaju kada je 10 kriterija i 3 alternative, oba načina daju u 92% slučajeva jednaku najbolju alternativu [11].

Primjer je preuzet iz [12] s tim da su odabiri prilagođeni i odluke za distributivni način su izračunate danim programom iz dodatka obzirom na preferencije koje su navedene u primjeru.

5 Dodatak

```
% ./ahp.m
close all; clear; clc;
h0 = figure('Position', [200 200 700 400],...
        'NumberTitle', 'off',...
        'Name', 'AHP',...
        'ToolBar', 'none',...
        'MenuBar', 'none',...
        'Resize', 'off');
cr_panel = uipanel('Parent', h0,...
               'Position', [.03 .18 .45 .8],...
               'Title', 'Kriteriji',...
               'FontSize', 11);
alt_panel = uipanel('Parent', h0,...
                'Position', [.52 .18 .45 .8],...
                'Title', 'Alternative',...
                'FontSize', 11);
cr_input = uicontrol('Style', 'edit',...
                   'Units', 'normal',...
                   'FontSize', 10,...
                   'Parent', cr_panel,...
                   'Position', [.05 .09 .9 .85],...
                   'Min', 1,...
                   'Max', 3,...
                   'Callback', @cr_input_callback);
alt_input = uicontrol('Style', 'edit',...
                   'Units', 'normal',...
                   'FontSize', 10,...
                   'Parent', alt_panel,...
                   'Position', [.05 .09 .9 .85],...
                   'Min', 1,...
                   'Max', 3,...
                   'Callback', @alt_input_callback);
button_panel = uipanel('Parent', h0,...
                    'Position', [.03 .02 .94 .14],...
                    'Title', '',...
                    'FontSize', 11);
button_next = uicontrol('Style', 'pushbutton',...
                   'String', 'Dalje',...
                   'Units', 'normal',...
                   'Position', [.87 .3 .1 .4],...
                   'Parent', button_panel,...
                   'Callback', {@btn_next_callback,h0,cr_input,
                     alt_input});
button_close = uicontrol('Style', 'pushbutton',...
                     'String', 'Zatvori',...
                     'Units', 'normal',...
```

```
                         'Position', [.75 .3 .1 .4],...
                         'Parent', button_panel,...
                         'Callback', @btn_close_callback);

% ./alt_input_callback.m
function alt_input_callback(src,event)
    alt_input = src.String;
    size_alt = size(alt_input);
    length_size_alt = size_alt(1,1);
    if length_size_alt < 2
        errordlg('Broj alternativa mora biti barem dva',...
                 'Greska unosa');
    end
end

% ./btn_close_callback.m
function btn_close_callback(src, event)
    close all;
    clear;
    clc;
end

% ./btn_next_callback.m
function btn_next_callback(src,event,h0,cr_input,alt_input)
    cr = cr_input.String;
    size_cr = size(cr);
    length_size_cr = size_cr(1,1);
    alt = alt_input.String
    size_alt = size(alt);
    length_size_alt = size_alt(1,1);
    cr_matrix = ones(length_size_cr, length_size_cr);
    alt_matrix = ones(length_size_alt,length_size_alt,
       length_size_cr);
    if length_size_cr<2
        errordlg('Broj kriterija mora biti barem dva',...
                 'Greska unosa');
    elseif length_size_alt<2
        errordlg('Broj alternativa mora biti barem dva',...
                 'Greska unosa');
    else
        close(h0);
        cell_of_cr = cellstr(cr);
        cell_of_alt = cellstr(alt);
        for ii=1:length_size_cr
            for jj=ii+1:length_size_cr
                d = dialog('Position', [350 350 500 150],...
                           'Name', 'Rangiranje kriterija');
                title = uicontrol('Parent', d,...
```

```
                        'Style', 'text',...
                        'Position', [150 65 210 65],...
                        'String', 'Odaberi preferenciju')
                          ;
        left_cr = uicontrol('Parent', d,...
                        'Style', 'text',...
                        'Position', [5 70 100 25],...
                        'String', cell_of_cr(ii));
        right_cr = uicontrol('Parent', d,...
                        'Style', 'text',...
                        'Position', [400 70 60 25],...
                        'String', cell_of_cr(jj));
        bg = uibuttongroup('Visible', 'on',...
                           'Units', 'pixels',...
                           'Position', [100 70 300 30],
                              ...
                           'SelectionChangedFcn', {@
                              swap_pref_cr, ii, jj});
        r2 = uicontrol(bg, 'Style',...
                      'radiobutton',...
                      'Units', 'pixels',...
                      'Position',[5 5 20 20],...
                      'HandleVisibility', 'off');
        r3 = uicontrol(bg, 'Style',...
                      'radiobutton',...
                      'Units', 'pixels',...
                      'Position', [276 5 20 20],...
                      'HandleVisibility', 'off');
        rangs = uicontrol('Parent', d,...
                          'Style', 'popup',...
                          'Position', [210 71 80 25],...
                          'String', {'Jednaka'; 'Umjerena
                             ';'Jaka';'Vrlo jaka';'
                             Ekstremna'},...
                          'Callback', {@put_in_cr_matrix,
                             ii,jj});
        btn = uicontrol('Parent', d,...
               'Position', [210 20 70 25],...
               'String', 'Dalje',...
               'Callback', 'delete(gcf)');
        uiwait();
    end
end

for cr=1:length_size_cr
    for ii=1:length_size_alt
        for jj=ii+1:length_size_alt
            d = dialog('Position', [350 350 500 150],...
```

```
                  'Name', 'Rangiranje alternativa');
    title = uicontrol('Parent', d,...
                      'Style', 'text',...
                      'Position', [150 65 210
                         65],...
                      'String', cell_of_cr(cr));
    left_cr = uicontrol('Parent', d,...
                    'Style', 'text',...
                    'Position', [5 70 100 25],...
                    'String', cell_of_alt(ii));
    right_cr = uicontrol('Parent', d,...
                         'Style', 'text',...
                         'Position', [400 70 60
                            25],...
                         'String', cell_of_alt(jj
                            ));
    bg = uibuttongroup('Visible', 'on',...
                       'Units', 'pixels',...
                       'Position', [100 70 300
                          30],...
                       'SelectionChangedFcn', {@
                          swap_pref_alt, cr, ii,
                          jj});
    r2 = uicontrol(bg, 'Style',...
                  'radiobutton',...
                  'Units', 'pixels',...
                  'Position',[5 5 20 20],...
                  'HandleVisibility', 'off');
    r3 = uicontrol(bg, 'Style',...
                   'radiobutton',...
                   'Units', 'pixels',...
                   'Position', [276 5 20 20],...
                   'HandleVisibility', 'off');
   rangs = uicontrol('Parent', d,...
          'Style', 'popup',...
          'Position', [210 39 80 58],...
          'String', {'Jednaka';'Umjerena';'Jaka';
             'Vrlo jaka';'Ekstremna'},...
          'Callback', {@put_in_alt_matrix, cr, ii
             , jj});
    btn = uicontrol('Parent', d,...
                    'Position', [210 20 70 25],
                       ...
                    'String', 'Dalje',...
                    'Callback', 'delete(gcf)');
    uiwait(d);
  end
end
```

```
end
%% uvjeti za metodu potencija
tol = 1e-6;
maxiter = 200;

%% metoda potencija za kriterije
v0 = zeros(length_size_cr, 1);
v0(1) = 1;
[lambda,w,~,~] = powermethod(cr_matrix, v0, tol, maxiter)
   ;

%% rezultat za kriterije
cr_res = cell_of_cr;
cr_res(:,2) = num2cell(w);
cr_res(end+1,:) = {'',''};
CI = (lambda-length_size_cr)/(length_size_cr-1);
cr_res(end+1,:) = {'C.I.', CI};
cr_res(end+1,:) = {'C.R', calculate_cr(CI,length_size_cr)
   };
d = dialog('Position', [350 350 300 300],...
           'Name', 'Rezultat');
t = uitable(d, 'Position', [10 50 280 230],...
            'RowName', {},...
            'ColumnName', {},...
            'Data', cr_res);
btn = uicontrol('Parent', d,...
               'Position', [200 10 70 25],...
               'String', 'Dalje',...
               'Callback', 'delete(gcf)');
uiwait(d);

%% rezultat za alternative
S = zeros(length_size_alt);
for ii=1:length_size_cr
 % metoda potencija za alternative
    v0 = zeros(length_size_alt,1);
    v0(1) = 1;
    [lambda,ys,~,~] = powermethod(alt_matrix(:,:,ii), v0,
        tol, maxiter);
    S(:,ii)=ys;
    alt_res = cell_of_alt;
    alt_res(:,2) = num2cell(ys);
    alt_res(end+1,:) = {'',''};
    CI = (lambda-length_size_alt)/(length_size_alt-1);
    alt_res(end+1,:) = {'C.I.', CI};
    alt_res(end+1,:) = {'C.R', calculate_cr(CI,
       length_size_alt)};
    d = dialog('Position', [350 350 300 300],...
```

```
                    'Name', cell_of_cr{ii});
        t = uitable(d, 'Position', [10 50 280 230],...
                     'RowName', {},...
                     'ColumnName', {},...
                     'Data', alt_res);
        btn = uicontrol('Parent', d,...
                        'Position', [200 10 70 25],...
                        'String', 'Dalje',...
                        'Callback', 'delete(gcf)');
        uiwait(d);
    end
end
v = S*w;
final_weights = cell_of_alt;
final_weights(:,2)=num2cell(v);
d = dialog('Position', [350 350 300 300],...
                'Name', 'Konacni rezultat');
    t = uitable(d, 'Position', [10 50 280 230],...
                 'RowName', {},...
                 'ColumnName', {},...
                 'Data', final_weights);
    btn = uicontrol('Parent', d,...
                    'Position', [200 10 70 25],...
                    'String', 'Izlaz',...
                    'Callback', 'delete(gcf)');
    uiwait(d);

function put_in_cr_matrix(src, event, ii, jj)
    if src.Value == 1
        cr_matrix(ii,jj) = 1;
    elseif src.Value == 2
        cr_matrix(ii,jj) = 3;
        cr_matrix(jj,ii) = 1/3;
    elseif src.Value == 3
        cr_matrix(ii,jj) = 5;
        cr_matrix(jj,ii) = 1/5;
    elseif src.Value == 4
        cr_matrix(ii,jj) = 7;
        cr_matrix(jj,ii) = 1/7;
    elseif src.Value == 5
        cr_matrix(ii,jj) = 9;
        cr_matrix(jj,ii) = 1/9;
    end
end

function swap_pref_cr(src, event, ii, jj)
    cr_matrix(ii,jj) = 1/cr_matrix(ii,jj);
    cr_matrix(jj,ii) = 1/cr_matrix(ii,jj);
```

```
    end

    function put_in_alt_matrix(src, event, cr, ii, jj)
        if src.Value == 1
            alt_matrix(ii,jj,cr) = 1;
        elseif src.Value == 2
            alt_matrix(ii,jj,cr) = 3;
            alt_matrix(jj,ii,cr) = 1/3;
        elseif src.Value == 3
            alt_matrix(ii,jj,cr) = 5;
            alt_matrix(jj,ii,cr) = 1/5;
        elseif src.Value == 4
            alt_matrix(ii,jj,cr) = 7;
            alt_matrix(jj,ii,cr) = 1/7;
        elseif src.Value == 5
            alt_matrix(ii,jj,cr) = 9;
            alt_matrix(jj,ii,cr) = 1/9;
        end
    end

    function swap_pref_alt(src, event, cr, ii, jj)
        alt_matrix(ii,jj,cr) = 1/alt_matrix(ii,jj,cr);
        alt_matrix(jj,ii,cr) = 1/alt_matrix(jj,ii,cr);
    end
end

% ./powermethod.m
function[lambda,vk,iter,err]=powermethod(A,v0,tol,maxiter)
v0=v0/norm(v0);
w=A*v0;
vk=w/norm(w);

lambda=vk'*A*vk;
err=norm(A*vk-lambda*vk)/abs(lambda);
iter=1;

while(iter<=maxiter && err>=tol)
    vk=A*vk;
    vk=vk/norm(vk);

    lambda=vk'*A*vk;
    err=norm(A*vk-lambda*vk)/abs(lambda);
    iter=iter+1;
end
vk=vk/norm(vk,1);
end
```

```matlab
% ./calculate_cr.m
function[cr]=calculate_cr(ci, n)
    RI=[0,0,.52,.89,1.11,1.25,1.35,1.4,1.45,1.49,1.51,1.54,1.56];
    if n <= 2
        cr='Nije definirano';
    else
        cr=ci/RI(n);
    end
end
```

Literatura

[1] D. Bakić, *Linearna algebra*, Školska knjiga, Zagreb, 2008.

[2] T. Heath, *Aristarchus of Samos, the Ancient Copernicus: A History of Greek Astronomy to Aristarchus, Together with Aristarchus's Treatise on the Sizes and Distances of the Sun and Moon*, Cambridge University Press, Cambridge, 2013.

[3] R.A. Horn, C.R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 2013.

[4] MathWorks®, *MATLAB® Creating Graphical User Interfaces, 2000-2015.*

[5] E. Mu, M. Pereya-Royas, *Practical Decision Making: An Introduction to the Analytic Hierarchy Process (AHP) Using Super Decisions V2*, Springer International Publishing, 2017.

[6] M. Panju, *Iterative Methods for Computing Eigenvalues and Eigenvectors*, The Waterloo Mathematics Review, **1**(2011), 9-18.

[7] A. Quarteroni, R. Sacco, F. Saleri, *Numerical Mathematics*, Springer-Verlag New York, Inc., New York, 2000.

[8] T.L. Saaty, *How to make a decision: The Analytic Hierarchy Process*, European Journal of Operational Research, **48**(1990), 9-26.

[9] T.L. Saaty, *Rank, Normalization and Idealization in the Analytic Hierarchy Process*, The 7th International Symposium on Analytic Hierarchy Process, Bali, Indonesia, 2003.

[10] T.L. Saaty, *Decision making with the analytic hierarchy process*, International Journal of Services Sciences, **1**(2008), 83-98.

[11] T.L. Saaty, *Fundamentals of Decision Making and Priority Theory with the Analytic Hierarchy Process*, RWS Publications, Pittsburgh, 2013.

[12] T.L. Saaty, L.G. Vargas, *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*, Springer, New York, 2012.

[13] S. Shiraishi, T. Obata, M. Daigo, *Properties of a Positive Reciprocal Matrix and Their Application to AHP*, Journal of the Operations Research Society of Japan, **41**(1998), 404-414.

[14] N. Truhar, *Numerička linearna algebra*, Sveučilište J.J. Strossmayera, Odjel za matematiku, Osijek, 2010.

Sažetak

Rad se temelji na osnovnim rezultatima Thomasa L. Saatyja iz područja teorije odlučivanja. Predstavljena je potrebna teorijska podloga AHP metode kao i osnovni rezultati i njihovi dokazi koji su potrebni za njenu praktičnu upotrebu. U radu je obrađena, kao jedna od mogućih, metoda svojstvenog vektora. Za praktično rješavanje navedenom metodom uvedena je metoda potencija za numerički izračun svojstvenog vektora i svojstvene vrijednosti. Nakon što su ostvarene sve pretpostavke, u radu je predstavljeno programsko rješenje za rješavanje problema odlučivanja u obliku grafičkog sučelja koristeći AHP metodu za hijerarhiju s tri nivoa. Sinteza rada je primjer odabira kuće od nekoliko ponuđenih u odnosu na kriterije. Na samom kraju rada umetnut je programski kod praktičnog dijela rada.

Ključne riječi

AHP metoda, teorija odlučivanja, hijerarhijsko odlučivanje, konzistentnost, fundamentalna ljestvica, metoda svojstvenog vektora, metoda potencija, indikator konzistentnosti, indeks konzistentnosti, omjer konzistentnosti, omjer nasumične konzistentnosti, grafičko korisničko sučelje

Title, summary and keywords

Title

Hierarchical Decision Making Using AHP Method

Summary

This paper is based on Thomas L. Saaty's contributions to the decision theory. A necessary theoretical basis for AHP method is introduced, including general results and proofs. This paper presents the principal eigenvalue method as one of the given possibilities. Power method has been introduced for numerical calculating eigenvector and eigenvalue of interest. After all conditions have been fulfilled, in this paper we show software solution using graphical interface for solving decision problems for three-level hierarchy. Paper synthesis is an example of choosing a house among several with respect to the criterions. At last, we included program code for the practical part of the paper in an appendix.

Keywords

AHP method, decision theory, hierarchical decision making, consistency, fundamental scale, eigenvector method, power method, consistency indicators, consistency index, consistency ratio, random consistency ratio, graphical user interface

Životopis

Želimir Piljić svoje obrazovanje započinje u Osnovnoj školi Ivana Kozarca u Županji. Nakon završetka osnovne škole, upisuje prirodoslovno-matematički smjer u Gimnaziji Županja u Županji. 2011. godine upisuje preddiplomski studij matematike na Odjelu za matematiku, Sveučilišta J.J. Strossmayera u Osijeku. Preddiplomski studij završava 2015. godine s temom završnog rada Kriptografija javnog ključa pod mentorstvom doc.dr.sc. Ivana Matića. Iste godine upisuje diplomski studij matematike i računarstva na Odjelu za matematiku. Tijekom diplomskog studija radio je u osječkoj tvrtci UHP Digital d.o.o. kao backend developer.