

Operacijska istraživanja

12. predavanje: Problem dodjeljivanja
Opcionalni materijal

Sažetak predavanja

- Problem dodjeljivanja – ponavljanje
- Primjenjive obrađene metode
- Specijalizirana metoda
 - Inicijalizacija
 - Iteriranje do optimuma

Problema dodjeljivanja (engl. assignment)

- problem dodjeljivanja skupa zadataka ($i=1, \dots, m$) skupu agenata ($j=1, \dots, n$) tako da su troškovi minimalni
- 1-na-1 (bijektivno) dodjeljivanje agenata zadatcima
- trošak c_{ij} pridružen svakom potencijalnom uparivanju
- posebna vrsta problema transportnog problema za $m = n$
 - Neuravnoteženi problem se da svesti na uravnoteženi (umjetni entiteti)

$$\min z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

$$\sum_{j=1}^n x_{ij} = 1, \forall i$$

$$\sum_{i=1}^n x_{ij} = 1, \forall j$$

Obradene metode rješavanja

- Primjenjive dosadašnje metode:
 1. Simpleks!
 - Ogromna tablica ✗
 2. Svođenje na transportni problem (ponude i potražnje od 1)
 - MODI metoda
 - puno degeneracija! ✗
 - potreban totalni uređaj dodanih ϵ_i gdje je i broj bazičnih varijabli vrijednosti 0
 - Za obje metode dosta posla...
 - Opcija korištenja metode **grananja i ograđivanja** sa pohlepnom heuristikom (knjiga str. 144-146) – obradit ćemo na jednom budućem predavanju

Specijalizirani algoritmi

- Specijalizirani algoritmi
 - Primjenjivi samo na uski tip problema
 - „brži”, tj. manje posla
 - npr. MODI je specijalizirana metoda za zatvoreni transportni problem
 - Temelje se na simpleksnoj metodi, koriste se specifičnosti strukture problema

Specijalizirani algoritmi

- Specijalizirani algoritam dodjeljivanja (za uravnoteženi problem)
 - mađarska metoda (engl. **Hungarian method**)
 - Gradi se koračno dodjeljivanje uz pomoć najnižih cijena
 - Iterativno se mijenjaju cijene – vodi do drugih dodjela (monotono veće kardinalnosti ?)
 - **Završava** kada se uspije napraviti potpuno 1-na-1 dodjeljivanje

Specijalizirani algoritam dodjeljivanja - Init

- Inicijalna dodjela
 1. Izračun novih cijena
 - a) oduzeti minimalni element svakog retka od svih elemenata retka
 - b) oduzeti minimalni element svakog stupca od svih elemenata stupca
 2. Izgradnja dodjele
 - a) u svakom retku sa samo jednom nulom, napraviti dodjeljivanje
 - b) ako je u retku više od jedne nule, ostaviti taj redak
 - c) ako je u retku dodijeljena nula, u tom stupcu križamo preostale nule
 - d) postupak ponoviti sa stupcima
 3. Ostvaren uvjet terminiranja?
 - a) Potpuna dodjela -> optimum nađen!
 - b) Parcijalna dodjela -> provesti sljedeću iteraciju (upute za koji slide)

1. primjer dodjeljivanja

- dodijeliti poslove osobama
- Poslovi: J1...4
- Osobe: P1...4

	P1	P2	P3	P4
J1	5	9	3	6
J2	8	7	8	2
J3	6	10	12	7
J4	3	10	8	6

1. primjer dodjeljivanja: inicijalna dodjela

- oduzeti minimalni element svakog retka od svih elemenata retka
- oduzeti minimalni element svakog stupca od svih elemenata stupca

	P1	P2	P3	P4
J1	2	2	0	3
J2	6	1	6	0
J3	0	0	6	1
J4	0	3	5	3

1.primjer dodjeljivanja: inicijalna dodjela

- u svakom retku sa samo jednom nulom, napraviti dodjeljivanje
- ako je u retku više od jedne nule, ostaviti taj redak
- ako je u retku dodijeljena nula, u tom stupcu križamo preostale nule

	P1	P2	P3	P4
J1	2	2	0	3
J2	6	1	6	0
J3	8	0	6	1
J4	0	3	5	3

2. primjer: inicijalna dodjela

- u svakom retku sa samo jednom nulom, napraviti dodjeljivanje
- ako je u retku više od jedne nule, ostaviti taj redak
- ako je u retku dodijeljena nula, u tom stupcu križamo preostale nule
- postupak ponoviti sa stupcima

	P1	P2	P3	P4	P5
J1	11	7	10	17	10
J2	13	21	7	11	13
J3	13	13	15	13	14
J4	18	10	13	16	14
J5	12	8	16	19	10

2. primjer: inicijalna dodjela

- nakon reduciranja na nule

	P1	P2	P3	P4	P5
J1	4	0	3	10	2
J2	6	14	0	4	5
J3	0	0	2	0	0
J4	8	0	3	6	3
J5	4	0	8	11	1

2. primjer: inicijalna dodjela

	P1	P2	P3	P4	P5
J1	4	0	3	10	2
J2	6	14	0	4	5
J3	0	8	2	8	0
J4	8	8	3	6	3
J5	4	8	8	11	1

- Izgrađena inicijalna dodjela
- Nije potpuna – NIJE optimalna!
- Moramo preći u iteriranje

Specijalizirani algoritam dodjeljivanja - iteriranje

- Iteriranje (kreće se iz postojeće dodjele)
 1. Priprema tablice
 - a) obilježiti nedodijeljene retke
 - b) Za svaki obilježeni redak koji ima nulu, obilježiti odgovarajuće stupce
 - c) Ako u označenim stupcima već postoje dodjeljivanja, označiti odgovarajuće retke
 - d) Ponavljati korake b-c dok ima promjena
 - e) Prekrižiti **NEoznačene** retke i **označene** stupce
 2. Korigiranje cijena
 - a) odabrati najmanju vrijednost od neprekriženih brojeva, θ
 - b) oduzeti θ od svih vrijednosti koje nisu prekrižene
 - c) dodati θ na sve vrijednosti dvostruko prekrižene
 - d) ne mijenjati vrijednosti koje su jednostruko prekrižene
 3. Izgradnja dodjele sa korigiranim cijenama (kao kod inicijalne dodjele)
 4. Ostvaren uvjet terminiranja?

2.Primjer – iteracija 1

Označavanje redaka i stupaca

	P1	P2	P3	P4	P5	
J1	4	0	3	10	2	✓
J2	6	14	0	4	5	
J3	0	0	2	0	0	
J4	8	0	3	6	3	✓
J5	4	0	8	11	1	✓

✓

2.Primjer – iteracija 1

Križanje redaka i stupaca

	P1	P2	P3	P4	P5	
J1	4	0	3	10	2	✓
J2	6	14	0	4	5	
J3	0	0	2	0	0	
J4	8	0	3	6	3	✓
J5	4	0	8	11	1	✓

✓

- $\theta = 1$

2. Primjer – iteracija 1

Nakon provođenja θ – korekcija cijena

	P1	P2	P3	P4	P5
J1	3	0	2	9	1
J2	6	15	0	4	5
J3	0	1	2	0	0
J4	7	0	2	5	2
J5	3	0	7	10	0

2. Primjer – iteracija 1

Ponovna faza dodjeljivanja

	P1	P2	P3	P4	P5
J1	3	0	2	9	1
J2	6	15	0	4	5
J3	0	8	2	3	0
J4	7	0	2	5	2
J5	3	8	7	10	0

2. Primjer – iteracija 2

Ponovna faza označavanja redaka i stupaca

	P1	P2	P3	P4	P5	
J1	3	0	2	9	1	✓
J2	6	15	0	4	5	
J3	0	×	2	×	×	
J4	7	×	2	5	2	✓
J5	3	×	7	10	0	

✓

- $\theta = 1$

2. Primjer – iteracija 2

Ponovna faza θ

	P1	P2	P3	P4	P5
J1	2	0	1	8	0
J2	6	16	0	4	5
J3	0	2	2	0	0
J4	6	0	1	4	1
J5	3	1	7	10	0

- $\theta = 1$

2. Primjer – iteracija 2

Ponovna faza dodjeljivanja

	P1	P2	P3	P4	P5
J1	2	8	1	8	8
J2	6	16	0	4	8
J3	0	2	2	8	8
J4	6	0	1	4	1
J5	3	1	7	10	0

2. Primjer – iteracija 3

Ponovna faza označavanja

	P1	P2	P3	P4	P5	
J1	2	1	1	8	0	✓
J2	6	16	0	4	3	
J3	0	2	2	3	2	
J4	6	0	1	4	1	✓
J5	3	1	7	10	0	✓
		✓			✓	

- $\theta = 1$

2. Primjer – iteracija 3

Ponovno korigiranje cijena

	P1	P2	P3	P4	P5
J1	1	0	0	7	0
J2	6	17	0	4	5
J3	0	3	2	0	0
J4	5	0	0	3	1
J5	2	1	6	9	0

2. Primjer – iteracija 3

Ponovna faza dodjeljivanja

	P1	P2	P3	P4	P5
J1	1	0	8	7	0
J2	6	17	0	4	5
J3	0	3	2	8	0
J4	5	0	8	3	1
J5	2	1	6	9	0

2. Primjer – iteracija 4

Ponovna faza označavanja

	P1	P2	P3	P4	P5	
J1	1	0	5	7	0	✓
J2	6	17	0	4	5	✓
J3	0	3	2	8	0	
J4	5	0	8	3	1	✓
J5	2	1	6	9	0	✓
		✓	✓		✓	

- $\theta = 1$

2. Primjer – iteracija 4

Ponovno faze korigiranja cijena i dodjeljivanja

	P1	P2	P3	P4	P5
J1	0	1	1	6	1
J2	5	17	0	3	5
J3	1	4	3	0	1
J4	4	0	1	2	1
J5	1	1	6	8	0

- dodjeljivanje završeno jer je nađen optimum: $11 + 7 + 13 + 10 + 10 = 51$