

# Rješavanje problema raspoređivanja medicinskog osoblja korištenjem metaheuristika

Izradio: Ivan Navratil

Mentor: doc. dr. sc. Marko Đurasević

Datum: petak, 30. rujna 2022.

# O problemu

- problem u kojem je medicinskim sestrama potrebno dodijeliti smjene prema njihovim zahtjevima, uzevši u obzir niz drugih uvjeta
- rezultat rješenja problema – tablica dimenzija  $n \times m$ 
  - $n$  – broj medicinskih sestara
  - $m$  – period planiranja
- razne definicije problema => različita tvrda i meka ograničenja

# Tvrda ograničenja problema

- Ograničenje jedne smjene dnevno
- Ograničenje iduće smjene
- Maksimalan broj određene smjene po radniku
- Minimalni i maksimalni broj radnih minuta
- Minimalni i maksimalni broj uzastopnih smjena
- Minimalni broj uzastopnih slobodnih dana
- Maksimalni broj radnih vikenda
- Slobodni dani

SECTION\_SHIFTS

# ShiftID, Length in mins, Shifts which cannot follow this shift |

↔ separated

E, 480,

L, 480, E

```
# ID, MaxShifts, MaxTotalMinutes, MinTotalMinutes,  
  ↪ MaxConsecutiveShifts, MinConsecutiveShifts,  
  ↪ MinConsecutiveDaysOff, MaxWeekends  
A, E=14 | L=14, 4320, 3360, 5, 2, 2, 1
```

SECTION\_DAYS\_OFF

# EmployeeID, DayIndexes (start at zero)

A, 3

# Meka ograničenja problema

- Zahtjev za smjenom i zahtjev protiv smjene
- Pokriće smjena

SECTION\_SHIFT\_ON\_REQUESTS

# EmployeeID, Day, ShiftID, Weight

A, 5, L, 1

...

SECTION\_SHIFT\_OFF\_REQUESTS

# EmployeeID, Day, ShiftID, Weight

G, 3, E, 2



SECTION\_COVER

# Day, ShiftID, Requirement, Weight for under, Weight for over  
0,E,4,100,1

# Korišteni algoritmi

- GRASP (Greedy randomized adaptive search procedure)
  - konstrukcijska faza
    - pronalazak rješenja koje zadovoljava sva tvrda ograničenja
  - faza lokalne pretrage
    - implementacija algoritma VND
    - poboljšanje kvalitete rješenja
- VND (Variable Neighbourhood Descent)
  - iterativan prolazak kroz susjedstva do
    - pronalaska boljeg rješenja kada postupak prolaska kroz susjedstva počinje iz početka
    - zadnjeg susjedstva te ako nije pronađeno bolje rješenje, algoritam je završio sa izvođenjem

# GRASP – konstrukcijska faza

- izrada rasporeda kreće od praznog rasporeda te se izvršava paralelna potraga za rasporedima radnika
- zajednička memorija poslova za pronalaz rješenja s manjom kaznom pokrića
- u svakoj iteraciji, za svaki dan, pronalaze se sve smjene koje radnik može raditi (može imati i slobodni dan)

# GRASP – konstrukcijska faza

- od pronađenih smjena, u nekim slučajevima, stvaraju se blokovi rješenja
  - rješenja koja zadovoljavaju sva tvrda ograničenja
  - rješenja koja zadovoljavaju sva tvrda ograničenja osim minimalnog broja minuta
  - rješenja koja zadovoljavaju sva tvrda ograničenja osim minimalnog broja minuta i minimalnog broja uzastopnih smjena

# GRASP – konstrukcijska faza

- odabir rješenja
  - ako u kandidatima postoji rješenje koje zadovoljava sva tvrda ograničenja, razmatraju se samo takva ograničenja
  - samo ako ne postoji rješenje koje zadovoljava sva tvrda ograničenja, uzimaju se kandidati koji imaju veći broj radnih minuta od rješenja od kojeg su nastali
    - jedino tvrdo ograničenje koje može biti prekršeno je ograničenje minimalnog broja minuta
  - samo ako je pretraživanje ponovno pokrenuto “malen” broj puta ili ako je pronađeno rješenje koje zadovoljava sva tvrda ograničenja vrši se filtriranje po kazni po broju koraka u odnosu na osnovno rješenje

# GRASP – faza lokalne pretrage

- implementirani algoritmi lokalne pretrage
  - prvi algoritam lokalne pretrage
  - drugi algoritam lokalne pretrage
  - vertikalna zamjena
  - horizontalna zamjena

# Prvi algoritam lokalne pretrage

- fokusira se na zadovoljavanje pokrića smjena
- algoritam radniku koji je slobodan ili radi neku smjenu na kojoj taj dan radi previše radnika dodjeljuje smjenu koja nije dodijeljena dovoljnom broju radnika
  - provjerava je li tim potezom prekršeno neko tvrdo ograničenje

**Tablica 5.4:** Kazne za kršenje ograničenja pokrića prije primjene lokalne pretrage.

Smjena \ Dan	Dan		
	0	1	2
E	0	0	0
D	0	100	0
N	0	0	0

**Tablica 5.5:** Kazne za kršenje ograničenja pokrića nakon primjene lokalne pretrage.

Smjena \ Dan	Dan		
	0	1	2
E	0	0	0
D	0	0	0
N	0	0	0

**Tablica 5.6:** Primjer rasporeda prije primjene lokalne pretrage.

Radnik \ Dan	Dan		
	0	1	2
0			N
1	N	E	D
2	D		N
3	E	N	E

**Tablica 5.7:** Primjer rasporeda nakon primjene lokalne pretrage.

Radnik \ Dan	Dan		
	0	1	2
0		D	N
1	N	E	D
2	D		N
3	E	N	E



# Drugi algoritam lokalne pretrage

- fokusira se na zadovoljavanje pokrića smjena
- algoritam radniku kojemu je dodijeljena smjena na dan kada je previše radnika dodijeljena ista smjena, uklanja tu smjenu toga dana, a dodjeljuje smjenu kojoj nije dodijeljen dovoljan broj radnika, tog ili nekog drugog slobodnog dana tog radnika

**Tablica 5.8:** Kazne za kršenje ograničenja pokrića prije primjene lokalne pretrage.

Smjena \ Dan	Dan		
	0	1	2
E	0	0	0
D	0	100	0
N	0	0	1

**Tablica 5.9:** Kazne za kršenje ograničenja pokrića nakon primjene lokalne pretrage.

Smjena \ Dan	Dan		
	0	1	2
E	0	0	0
D	0	0	0
N	0	0	0

**Tablica 5.10:** Primjer rasporeda prije primjene lokalne pretrage.

Radnik \ Dan	Dan		
	0	1	2
0	N	N	
1	N	E	D
2	D		N
3	E	N	E

**Tablica 5.11:** Primjer rasporeda nakon primjene lokalne pretrage.

Radnik \ Dan	Dan		
	0	1	2
0	N	D	
1	N	E	D
2	D		N
3	E	N	E

# Vertikalna zamjena i horizontalna zamjena

- vertikalna zamjena
  - fokusirana na smanjenje kazne za ograničenja za i protiv smjene
  - za danu veličinu bloka (2, 3, 4, 5) zamjene se smjene dvaju radnika u određenom rasponu (+ zamjena isključivo vikenda)
- horizontalna zamjena
  - fokusirana na smanjenje kazni pokrića
  - za danu veličinu bloka (2, 3, 4, 5) zamjene se blokovi smjena istog radnika u određenom rasponu (+ zamjena isključivo vikenda)

# Konačni algoritam

- ponavljaju se vertikalna pretraga, prvi algoritam lokalne pretrage i drugi algoritam lokalne pretrage dok se rješenje poboljšava, a prvi algoritam lokalne pretrage i drugi algoritam lokalne pretrage se ponavljaju prema VND algoritmu
- ponavljanje horizontalne pretrage određen broj puta
- pretraživanje se zaustavlja nakon što se rješenje nije poboljšalo određen broj puta

# Konfiguracija programskog rješenja (1/2)

**instance:** 1

**instances\_path:** src/main/resources/instances/

**search\_algorithms:** GRASP, VND

**grasp:**

**solutions\_by\_employee\_path:**

↪ src/main/resources/solutions-by-employee/

**solutions\_write\_path:** src/main/resources/solutions-grasp-write/

**iterations\_before\_restart:** 10x

**iterations\_before\_ignoring\_penalty:** 10

**penalty\_alpha:** 0.25

**threads:** 6

**neighborhood:**

**searchType:** mid

**includeNoShift:** true

# Konfiguracija programskog rješenja (2/2)

## **local\_search:**

**solutions\_read\_path:** src/main/resources/solutions-read/

**solutions\_write\_path:**

↪ src/main/resources/solutions-grasp-ls-write/

## **horizontal\_search:**

**repeat:** 10

## **variable\_neighbor\_descent:**

**break\_after\_times\_not\_improved:** 25

# Usporedba rješenja

Instanca	Tjedana	Radnika	Tipova smjena	Pronađeno rješenje	Najbolje rješenje
1	2	8	1	609	607
2	2	14	2	944	828
3	2	20	3	1229	1001
4	4	10	2	1854	1716
5	4	16	2	1484	1143
6	4	18	3	2720	1950
7	4	20	3	1770	1056
8	4	30	4	3166	1300
9	4	36	4	1242	439
10	4	40	5	5864	4631
11	4	50	6	4726	3443
12	4	60	10	8232	4040
13	4	120	18	8118	1348
14	6	32	4	2858	1278
15	6	45	6	11105	3831
16	8	20	3	5523	3225
17	8	32	4	10222	5746
18	12	22	3	9083	4459
19	12	40	5	11016*	3149
20	26	50	6	N/A	4769
21	26	100	8	N/A	21133
22	52	50	10	N/A	30244
23	52	100	16	N/A	17428
24	52	150	32	N/A	42463

# Nadogradnje rješenja

- ubrzavanje postojeće implementacije
- dodavanje dodatnih lokalnih pretraga
- promjena poretka izvođenja lokalnih pretraga



# Zaključak

- algoritam se pokazao dobrim te pronalazi rješenje koje zadovoljava sva tvrda ograničenja u roku od nekoliko minuta, osim za najveće instance problema
- trenutnoj implementaciji algoritma mogu se povećati performanse
- faza lokalne pretrage traje znatno duže od faze izgradnje rješenja te ima najviše mjesta za napredak