

Metoda promenljivih okolina Variable Neighborhood Search (VNS)

Tatjana Davidović,

Matematički institut SANU

14. januar 2016.



Sadržaj

1 Opis metode

2 Varijante VNS metode

3 Primeri primene



Opšte karakteristike

- Matematički zasnovana metaheuristika (Mladenović and Hansen, C&OR, 1997);



Opšte karakteristike

- Matematički zasnovana metaheuristika (Mladenović and Hansen, C&OR, 1997);
- Bazirana na jednom rešenju i pretraživanju njegovih okolina;



Opšte karakteristike

- Matematički zasnovana metaheuristika (Mladenović and Hansen, C&OR, 1997);
- Bazirana na jednom rešenju i pretraživanju njegovih okolina;
- Osnovni korak je funkcija lokalnog pretraživanja (Local Search, LS);



Opšte karakteristike

- Matematički zasnovana metaheuristika (Mladenović and Hansen, C&OR, 1997);
- Bazirana na jednom rešenju i pretraživanju njegovih okolina;
- Osnovni korak je funkcija lokalnog pretraživanja (Local Search, LS);
- Sistematska upotreba više okolina za povećanje efikasnosti pretrage;



Opšte karakteristike

- Matematički zasnovana metaheuristika (Mladenović and Hansen, C&OR, 1997);
- Bazirana na jednom rešenju i pretraživanju njegovih okolina;
- Osnovni korak je funkcija lokalnog pretraživanja (Local Search, LS);
- Sistematska upotreba više okolina za povećanje efikasnosti pretrage;
- VNS se oslanja na tri jednostavne činjenice:

Fact 1 Lokalni optimum u odnosu na jednu okolinu ne mora biti optimum u odnosu na neku drugu;

Fact 2 Globalni optimum je lokalni u odnosu na svaku okolinu;

Fact 3 Za većinu problema lokalni optimumi u odnosu na razne okoline su relativno blizu.



Opšte karakteristike

- Matematički zasnovana metaheuristika (Mladenović and Hansen, C&OR, 1997);
- Bazirana na jednom rešenju i pretraživanju njegovih okolina;
- Osnovni korak je funkcija lokalnog pretraživanja (Local Search, LS);
- Sistematska upotreba više okolina za povećanje efikasnosti pretrage;
- VNS se oslanja na tri jednostavne činjenice:

Fact 1 Lokalni optimum u odnosu na jednu okolinu ne mora biti optimum u odnosu na neku drugu;

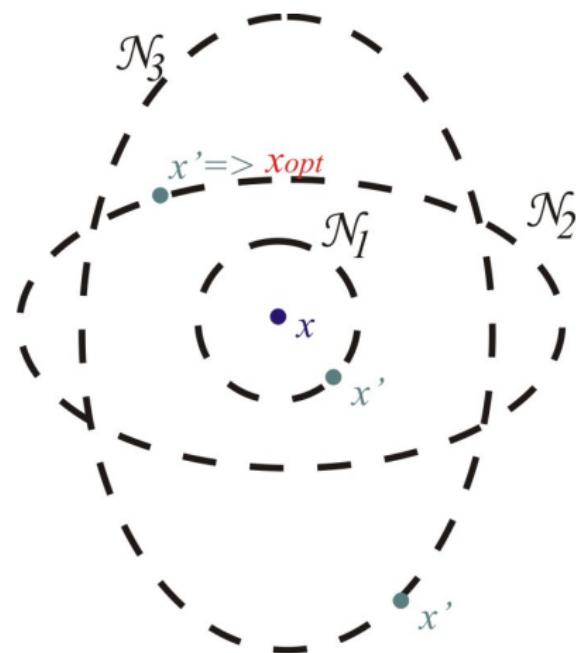
Fact 2 Globalni optimum je lokalni u odnosu na svaku okolinu;

Fact 3 Za većinu problema lokalni optimumi u odnosu na razne okoline su relativno blizu.

- Parametar k_{max} – maksimalni broj okolina.



Ilustracija upotrebe više okolina



Osnovni koraci metode

- Nalaženje početnog rešenja;



Osnovni koraci metode

- Nalaženje početnog rešenja;
- Lokalna popravka početnog rešenja;



Osnovni koraci metode

- Nalaženje početnog rešenja;
- Lokalna popravka početnog rešenja;
- Faza razmrdavanja (shaking) je diversifikacija, slučajno se bira rešenje u tekućoj okolini;



Osnovni koraci metode

- Nalaženje početnog rešenja;
- Lokalna popravka početnog rešenja;
- Faza razmrdavanja (shaking) je diversifikacija, slučajno se bira rešenje u tekućoj okolini;
- Faza popravljanja: Lokalno pretraživanje se obavlja počev od izabranog rešenja;



Osnovni koraci metode

- Nalaženje početnog rešenja;
- Lokalna popravka početnog rešenja;
- Faza razmrdavanja (shaking) je diversifikacija, slučajno se bira rešenje u tekućoj okolini;
- Faza popravljanja: Lokalno pretraživanje se obavlja počev od izabranog rešenja;
- Pomeraj: ukoliko je LS-om popravljeno trenutno najbolje rešenje pretraga se koncentriše oko njega inače se menja okolina za razmrdavanje;



Osnovni koraci metode

- Nalaženje početnog rešenja;
- Lokalna popravka početnog rešenja;
- Faza razmrdavanja (shaking) je diversifikacija, slučajno se bira rešenje u tekućoj okolini;
- Faza popravljanja: Lokalno pretraživanje se obavlja počev od izabranog rešenja;
- Pomeraj: ukoliko je LS-om popravljeno trenutno najbolje rešenje pretraga se koncentriše oko njega inače se menja okolina za razmrdavanje;
- Pretraga okoline može biti detaljna (BEST IMPROVEMENT) ili samo do prvog poboljšanja (FIRST IMPROVEMENT).



Osnovni koraci metode (nast.)

- U svakom trenutku postoje 3 rešenja: trenutno najbolje x_{best} , slučajno izabrano x' i lokalni optimum x'' ;



Osnovni koraci metode (nast.)

- U svakom trenutku postoje 3 rešenja: trenutno najbolje x_{best} , slučajno izabrano x' i lokalni optimum x'' ;
- Kada se stigne do poslednje okoline k_{max} postupak se ponavlja;



Osnovni koraci metode (nast.)

- U svakom trenutku postoje 3 rešenja: trenutno najbolje x_{best} , slučajno izabrano x' i lokalni optimum x'' ;
- Kada se stigne do poslednje okoline k_{max} postupak se ponavlja;
- Okoline se mogu ređati i od poslednje ka prvoj;



Osnovni koraci metode (nast.)

- U svakom trenutku postoje 3 rešenja: trenutno najbolje x_{best} , slučajno izabrano x' i lokalni optimum x'' ;
- Kada se stigne do poslednje okoline k_{max} postupak se ponavlja;
- Okoline se mogu ređati i od poslednje ka prvoj;
- Dodatni parametri: k_{min} , k_{step} , p_{plato} ;



Osnovni koraci metode (nast.)

- U svakom trenutku postoje 3 rešenja: trenutno najbolje x_{best} , slučajno izabrano x' i lokalni optimum x'' ;
- Kada se stigne do poslednje okoline k_{max} postupak se ponavlja;
- Okoline se mogu ređati i od poslednje ka prvoj;
- Dodatni parametri: k_{min} , k_{step} , p_{plato} ;
- Kriterijum zaustavljanja;



Osnovni koraci metode (nast.)

- U svakom trenutku postoje 3 rešenja: trenutno najbolje x_{best} , slučajno izabrano x' i lokalni optimum x'' ;
- Kada se stigne do poslednje okoline k_{max} postupak se ponavlja;
- Okoline se mogu ređati i od poslednje ka prvoj;
- Dodatni parametri: k_{min} , k_{step} , p_{plato} ;
- Kriterijum zaustavljanja;
- Najbolji lokalni optimum se prijavljuje korisniku.



VNS Pseudokod

Initialization. Find an initial solution $x \in X$; improve it with the local search to obtain x_{best} ; choose stopping criterion; set STOP = 0.

Repeat

1. Set $k = 1$.
2. Repeat

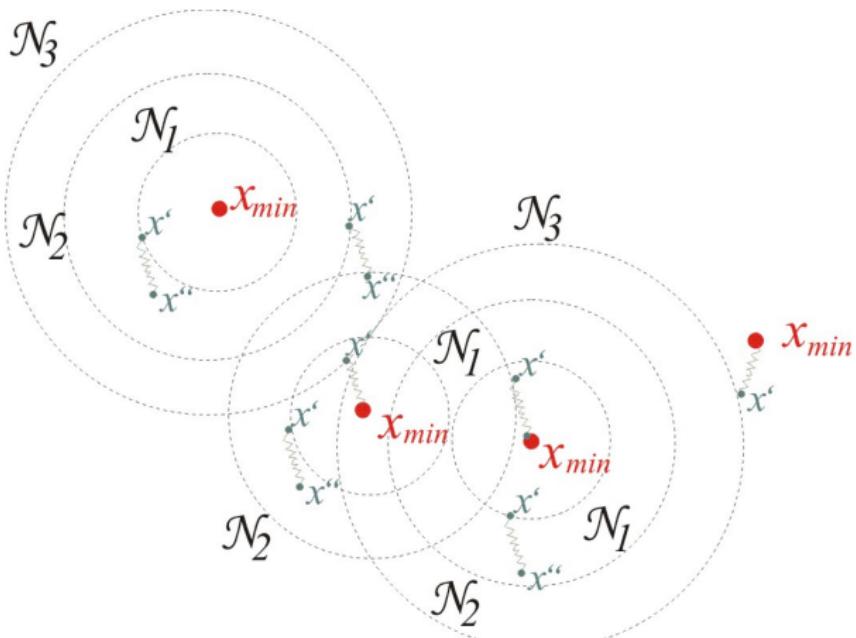
- (a) *Shake.* Generate a random point x' in the k^{th} neighborhood of x_{best} , ($x' \in N_k(x_{best})$).
- (b) *Improve.* Apply some LS method with x' as the initial solution; denote with x'' the obtained local optimum.
- (c) *Move.* If this local optimum is better than the current incumbent, move there ($x_{best} = x''$), and continue the search within \mathcal{N}_1 ($k = 1$); otherwise move to the next neighborhood ($k = k + 1$).
- (d) *Stopping criterion.* If the stopping condition is met, set STOP = 1.

until $k == k_{max}$ or STOP == 1.

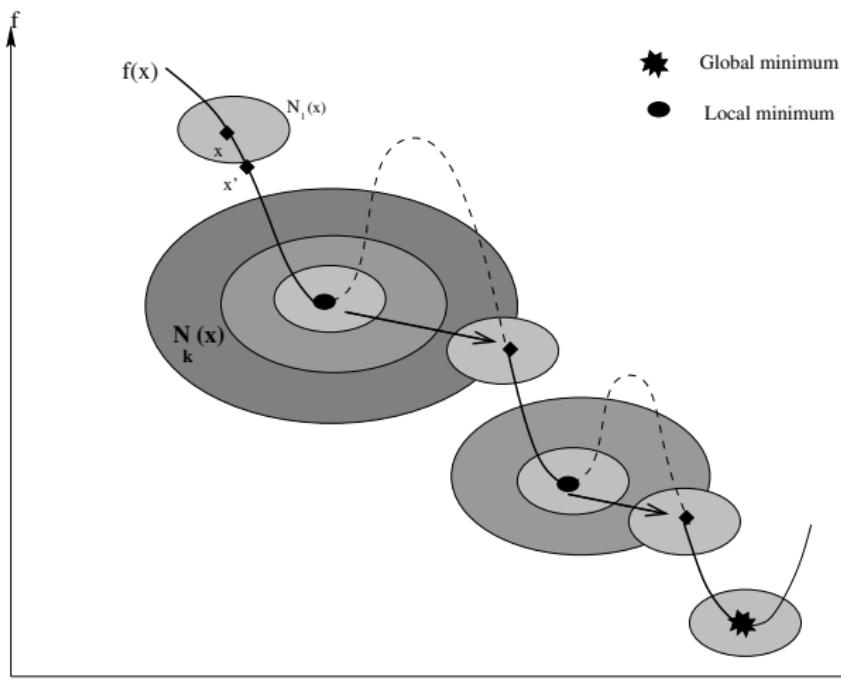
until STOP == 1.



Ilustracija rada VNS metode



3-D ilustracija



Spisak varijanti

- Osnovna metoda promenljivih okolina (Basic VNS);
- Redukovana metoda promenljivih okolina (Reduced VNS);
- Metoda promenljivog spusta (Variable Neighborhood Descent, VND);
- Opšta metoda promenljivih okolina (General VNS);
- Metoda promenljivih okolina sa dekompozicijom (Variable Neighbourhood Decomposition Search, VNDS);
- Adaptivna metoda promenljivih okolina (Skewed VNS);
- Primal-dual VNS;
- Paralelni VNS, Metoda promenljivih formulacija (Variable Neighborhood Formulation Space Search), itd.



RVNS

- Stohastička varijanta;



RVNS

- Stohastička varijanta;
- Ima samo fazu razmrdavanja, jednu okolinu ali se menja rastojanje od najboljeg rešenja;



RVNS

- Stohastička varijanta;
- Ima samo fazu razmrdavanja, jednu okolinu ali se menja rastojanje od najboljeg rešenja;
- Na svakom od izabranih rastojanja bira se slučajno jedno rešenje;



RVNS

- Stohastička varijanta;
- Ima samo fazu razmrdavanja, jednu okolinu ali se menja rastojanje od najboljeg rešenja;
- Na svakom od izabranih rastojanja bira se slučajno jedno rešenje;
- Ako je ono bolje od trenutno najboljeg, selimo se u njega i vraćamo se u prvu okolinu (na rastojanje 1);



RVNS

- Stohastička varijanta;
- Ima samo fazu razmrdavanja, jednu okolinu ali se menja rastojanje od najboljeg rešenja;
- Na svakom od izabranih rastojanja bira se slučajno jedno rešenje;
- Ako je ono bolje od trenutno najboljeg, selimo se u njega i vraćamo se u prvu okolinu (na rastojanje 1);
- Ako izabrano rešenje nije bolje, povećavamo rastojanje;

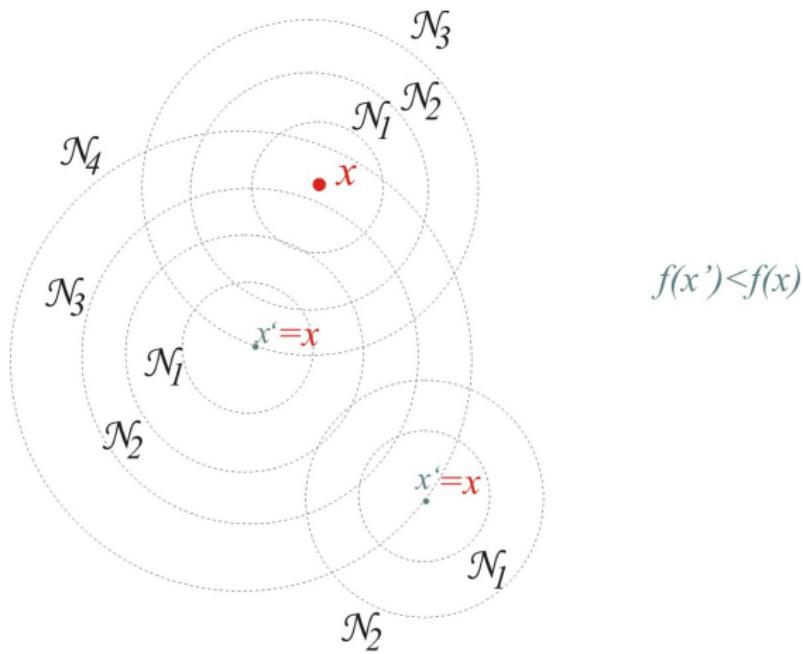


RVNS

- Stohastička varijanta;
- Ima samo fazu razmrdavanja, jednu okolinu ali se menja rastojanje od najboljeg rešenja;
- Na svakom od izabranih rastojanja bira se slučajno jedno rešenje;
- Ako je ono bolje od trenutno najboljeg, selimo se u njega i vraćamo se u prvu okolinu (na rastojanje 1);
- Ako izabrano rešenje nije bolje, povećavamo rastojanje;
- Kad stignemo do najvećeg rastojanja, vraćamo se u prvu okolinu (na rastojanje 1) sve dok nije zadovoljen kriterijum zaustavljanja.



Ilustracija RVNS-a



Metoda promenljivog spusta (Variable Neighborhood Descent, VND)

- Deterministička varijanta;



Metoda promenljivog spusta (Variable Neighborhood Descent, VND)

- Deterministička varijanta;
- Nema fazu razmrdavanja;



Metoda promenljivog spusta (Variable Neighborhood Descent, VND)

- Deterministička varijanta;
- Nema fazu razmrdavanja;
- Lokalno pretraživanje se obavlja u svakoj od izabranih okolina;



Metoda promenljivog spusta (Variable Neighborhood Descent, VND)

- Deterministička varijanta;
- Nema fazu razmrdavanja;
- Lokalno pretraživanje se obavlja u svakoj od izabranih okolina;
- Okoline su uredjene, a pri svakom poboljšanju vraćamo se u prvu okolinu;



Metoda promenljivog spusta (Variable Neighborhood Descent, VND)

- Deterministička varijanta;
- Nema fazu razmrdavanja;
- Lokalno pretraživanje se obavlja u svakoj od izabranih okolina;
- Okoline su uredjene, a pri svakom poboljšanju vraćamo se u prvu okolinu;
- Pretraga okolina može biti BI ili FI;



Metoda promenljivog spusta (Variable Neighborhood Descent, VND)

- Deterministička varijanta;
- Nema fazu razmrdavanja;
- Lokalno pretraživanje se obavlja u svakoj od izabranih okolina;
- Okoline su uredjene, a pri svakom poboljšanju vraćamo se u prvu okolinu;
- Pretraga okolina može biti BI ili FI;
- Zaustavljamo se kada u poslednjoj okolini više nema poboljšanja trenutno najboljeg rešenja;

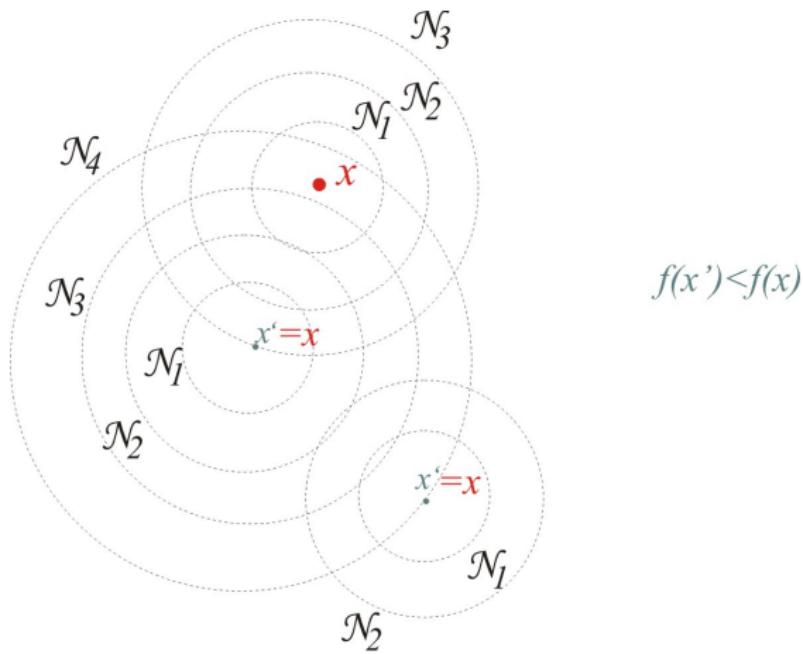


Metoda promenljivog spusta (Variable Neighborhood Descent, VND)

- Deterministička varijanta;
- Nema fazu razmrdavanja;
- Lokalno pretraživanje se obavlja u svakoj od izabranih okolina;
- Okoline su uredjene, a pri svakom poboljšanju vraćamo se u prvu okolinu;
- Pretraga okolina može biti BI ili FI;
- Zaustavljamo se kada u poslednjoj okolini više nema poboljšanja trenutno najboljeg rešenja;
- Najbolji lokalni optimum se prijavljuje korisniku.



Ilustracija VND-a



Ostale metode promenljivih okolina

- GVNS je kombinacija BVNS-a i VND-a: Umesto običnog lokalnog pretraživanja primenjuje se VND. Tipovi okolina mogu biti različiti, a VND-u se može ograničiti vreme za rad.



Ostale metode promenljivih okolina

- GVNS je kombinacija BVNS-a i VND-a: Umesto običnog lokalnog pretraživanja primenjuje se VND. Tipovi okolina mogu biti različiti, a VND-u se može ograničiti vreme za rad.
- VNDS primenjuje neku od varijanti VNS-a na podproblem datog problema: polazni problem se podeli na manje celine i svaka od njih se rešava VNS-om, a onda se konačno rešenje sklapa od komponenti koje su rešenja podproblema.



Ostale metode promenljivih okolina

- GVNS je kombinacija BVNS-a i VND-a: Umesto običnog lokalnog pretraživanja primenjuje se VND. Tipovi okolina mogu biti različiti, a VND-u se može ograničiti vreme za rad.
- VNDS primenjuje neku od varijanti VNS-a na podproblem datog problema: polazni problem se podeli na manje celine i svaka od njih se rešava VNS-om, a onda se konačno rešenje sklapa od komponenti koje su rešenja podproblema.
- SVNS dozvoljava prelazak u lošija rešenja sa određenom verovatnoćom. Ta verovatnoća je novi parametar metode. Ova varijanta ima neku vrstu restarta.



Ostale metode promenljivih okolina

- GVNS je kombinacija BVNS-a i VND-a: Umesto običnog lokalnog pretraživanja primenjuje se VND. Tipovi okolina mogu biti različiti, a VND-u se može ograničiti vreme za rad.
- VNDS primenjuje neku od varijanti VNS-a na podproblem datog problema: polazni problem se podeli na manje celine i svaka od njih se rešava VNS-om, a onda se konačno rešenje sklapa od komponenti koje su rešenja podproblema.
- SVNS dozvoljava prelazak u lošija rešenja sa određenom verovatnoćom. Ta verovatnoća je novi parametar metode. Ova varijanta ima neku vrstu restarta.
- Primal-dual VNS koristi MIP formulaciju osnovnog i dualnog problema i rešava relaksacije heuristički.



Ostale metode promenljivih okolina

- GVNS je kombinacija BVNS-a i VND-a: Umesto običnog lokalnog pretraživanja primenjuje se VND. Tipovi okolina mogu biti različiti, a VND-u se može ograničiti vreme za rad.
- VNDS primenjuje neku od varijanti VNS-a na podproblem datog problema: polazni problem se podeli na manje celine i svaka od njih se rešava VNS-om, a onda se konačno rešenje sklapa od komponenti koje su rešenja podproblema.
- SVNS dozvoljava prelazak u lošija rešenja sa određenom verovatnoćom. Ta verovatnoća je novi parametar metode. Ova varijanta ima neku vrstu restarta.
- Primal-dual VNS koristi MIP formulaciju osnovnog i dualnog problema i rešava relaksacije heuristički.
- VNFSS osim okolina menja i formulacije problema.



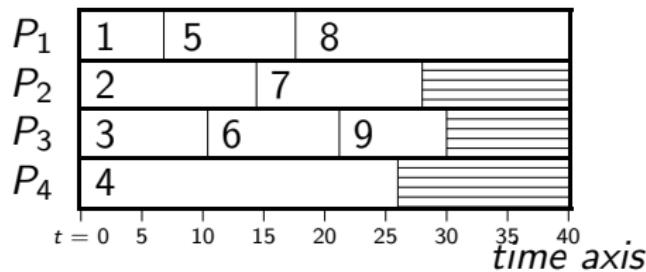
Raspoređivanje nezavisnih zadataka na identične mašine

$T = \{1, 2, \dots, n\}$ - Skup zadataka,

$M = \{1, 2, \dots, m\}$ - Skup mašina,

l_i - dužina izvršavanja svakog zadatka i ($i = 1, 2, \dots, n$).

Cilj: Minimizacija ukupnog vremena izvršavanja svih zadataka (*makespan*).



Gantt diagram–raspodela zadataka po mašinama



VNS implementacija

- Rešenje je struktura (S, A, Y) .

$S_{m \times n}$ je dinamička matrica.

Element s_{ji} predstavlja indeks i -tog zadatka dodeljenog procesoru j .

Vektor $A = \{a_1, \dots, a_m\}$ sadrži broj zadataka dodeljenih svakom procesoru.

Elementi vektora $Y = \{y_1, \dots, y_m\}$ su dužine rada procesora, tj.

$$y_j = \sum_{i=1}^{a_j} l_{s_{ji}}.$$



VNS implementacija

- Rešenje je struktura (S, A, Y) .

$S_{m \times n}$ je dinamička matrica.

Element s_{ji} predstavlja indeks i -tog zadatka dodeljenog procesoru j .

Vektor $A = \{a_1, \dots, a_m\}$ sadrži broj zadataka dodeljenih svakom procesoru.

Elementi vektora $Y = \{y_1, \dots, y_m\}$ su dužine rada procesora, tj.

$$y_j = \sum_{i=1}^{a_j} l_{s_{ji}}.$$

- Okoline:

- ➊ premeštanje zadatka (shift),
- ➋ dva zadatka zamene mesta (interchange).



VNS implementacija

- Rešenje je struktura (S, A, Y) .

$S_{m \times n}$ je dinamička matrica.

Element s_{ji} predstavlja indeks i -tog zadatka dodeljenog procesoru j .

Vektor $A = \{a_1, \dots, a_m\}$ sadrži broj zadataka dodeljenih svakom procesoru.

Elementi vektora $Y = \{y_1, \dots, y_m\}$ su dužine rada procesora, tj.

$$y_j = \sum_{i=1}^{a_j} l_{s_{ji}}.$$

- Okoline:

- ① premeštanje zadatka (shift),
- ② dva zadatka zamene mesta (interchange).

- Ažuriranje rešenja u $O(1)$ koraka.



VNS implementacija

- Rešenje je struktura (S, A, Y) .

$S_{m \times n}$ je dinamička matrica.

Element s_{ji} predstavlja indeks i -tog zadatka dodeljenog procesoru j .

Vektor $A = \{a_1, \dots, a_m\}$ sadrži broj zadataka dodeljenih svakom procesoru.

Elementi vektora $Y = \{y_1, \dots, y_m\}$ su dužine rada procesora, tj.

$$y_j = \sum_{i=1}^{a_j} l_{s_{ji}}.$$

- Okoline:

- ① premeštanje zadatka (shift),
- ② dva zadatka zamene mesta (interchange).

- Ažuriranje rešenja u $O(1)$ koraka.
- Redukcija okolina, sortiranje zadataka.



p-centar problem

- Dat je skup od n čvorova (lokacija, korisnika);
- Data je $D = [d_{ij}]_{n \times n}$ matrica rastojanja između čvorova;
- Cilj je locirati p uslužnih centara tako da se minimizira maksimalno rastojanje između korisnika i pridruženog centra;
- Korisnici se pridružuju najbližem uspostavljenom centru;
- Centri mogu biti locirani u bilo kom od n čvorova.



Implementacija VNS-a

- N. Mladenović, M. Labbe, P. Hansen, Solving the p-center problem with tabu search and variable neighborhood search, Networks 42 (1) (2003) 48–64.;



Implementacija VNS-a

- N. Mladenović, M. Labbe, P. Hansen, Solving the p-center problem with tabu search and variable neighborhood search, Networks 42 (1) (2003) 48–64.;
- Okolina: zamena centra ne-centrom (Vertex substitution, 1-interchange);



Implementacija VNS-a

- N. Mladenović, M. Labbe, P. Hansen, Solving the p-center problem with tabu search and variable neighborhood search, Networks 42 (1) (2003) 48–64.;
- Okolina: zamena centra ne-centrom (Vertex substitution, 1-interchange);
- Veličina okoline je $p(n - p)$;



Implementacija VNS-a

- N. Mladenović, M. Labbe, P. Hansen, Solving the p-center problem with tabu search and variable neighborhood search, Networks 42 (1) (2003) 48–64.;
- Okolina: zamena centra ne-centrom (Vertex substitution, 1-interchange);
- Veličina okoline je $p(n - p)$;
- Efikasna implementacija: prvo se doda ne-centar, a onda izbaci najgori centar;



Implementacija VNS-a

- N. Mladenović, M. Labbe, P. Hansen, Solving the p-center problem with tabu search and variable neighborhood search, Networks 42 (1) (2003) 48–64.;
- Okolina: zamena centra ne-centrom (Vertex substitution, 1-interchange);
- Veličina okoline je $p(n - p)$;
- Efikasna implementacija: prvo se doda ne-centar, a onda izbaci najgori centar;
- Smanjuje se veličina okoline za faktor p ;



Implementacija VNS-a

- N. Mladenović, M. Labbe, P. Hansen, Solving the p-center problem with tabu search and variable neighborhood search, Networks 42 (1) (2003) 48–64.;
- Okolina: zamena centra ne-centrom (Vertex substitution, 1-interchange);
- Veličina okoline je $p(n - p)$;
- Efikasna implementacija: prvo se doda ne-centar, a onda izbaci najgori centar;
- Smanjuje se veličina okoline za faktor p ;
- Koriste se posebne strukture podataka (heap);



Implementacija VNS-a

- N. Mladenović, M. Labbe, P. Hansen, Solving the p-center problem with tabu search and variable neighborhood search, Networks 42 (1) (2003) 48–64.;
- Okolina: zamena centra ne-centrom (Vertex substitution, 1-interchange);
- Veličina okoline je $p(n - p)$;
- Efikasna implementacija: prvo se doda ne-centar, a onda izbaci najgori centar;
- Smanjuje se veličina okoline za faktor p ;
- Koriste se posebne strukture podataka (heap);
- Složenost ažuriranja rešenja $O(n \log n)$.



VNS konferencije

- 18th Mini EURO Conference on VNS, Tenerife, Spain, Nov. 23-25, 2005;
- 28th Mini EURO Conference on VNS, Herceg-Novi, Montenegro, Oct. 04-07, 2012;
- 3rd International Conference on VNS, Djerba, Tunisia, Oct. 08-11, 2014;



Hvala na pažnji!

Pitanja?

Tatjana Davidović
tanjad@mi.sanu.ac.rs

