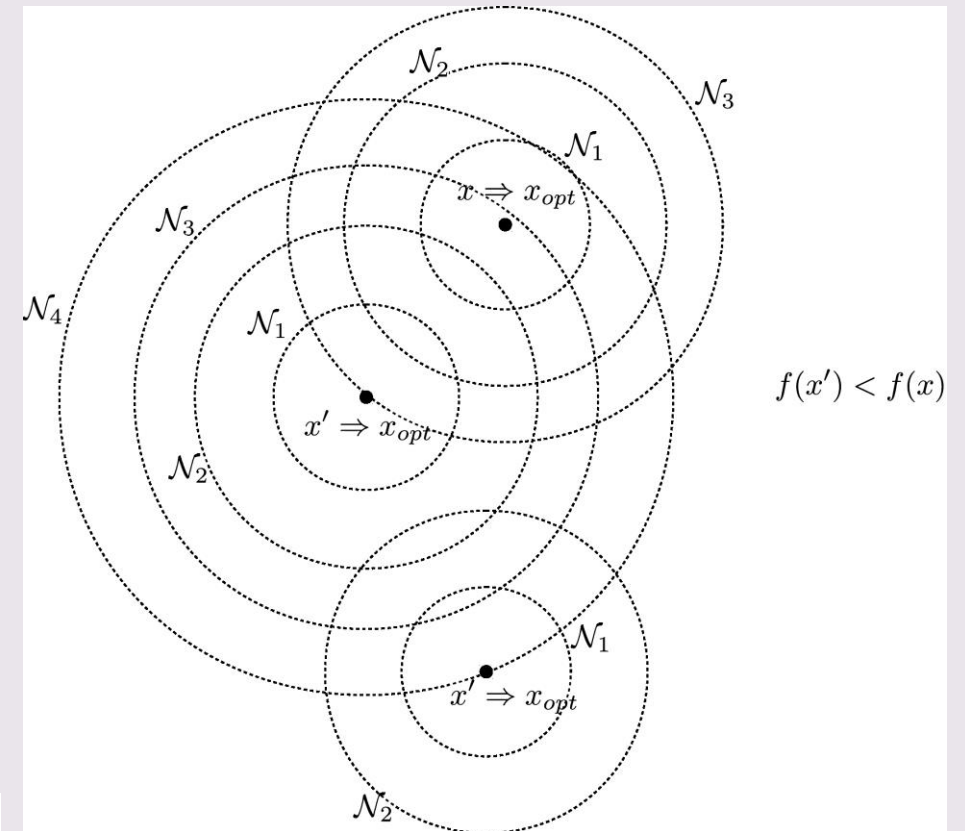


# Pretraga promenljivih okolina

## Variable Neighborhood Search

- P. Hansen i N. Mladenović, 1997
- Uzastopno pretraživanje skupa predefinisanih okolina u cilju pronalaženja boljeg rešenja
  - Lokalni optimum za jednu okolinu ne mora biti optimum i za drugu okolinu
  - Globalni optimum je optimum u odnosu na svaku okolinu
  - Empirijski dokazi pokazuju da su većina lokalni optimuma relativno blizu jedni drugima

$$N_1(x) \subset N_2(x) \subset \dots \subset N_k(x), \forall x \in S$$



# VNS

- Tri glavna koraka koja je ciklično ponavljaju dok se ne ispune uslovi zaustavljanja:
  1. *Shaking procedure*
  2. *Improvement procedure*
  3. *Neighborhood change step*

# VNS

## 1. Shaking procedure – procedura potresa

- Izbegavanje zamke lokalnog minimuma
- Neka je definisan skup operatora  $\mathcal{N} = \{\mathcal{N}_1, \dots, \mathcal{N}_{k_{max}}\}$ . Za svaki operator  $\mathcal{N}_k, 1 \leq k \leq max$ , datom rešenju  $x$  se dodeljuje predefinisana okolina  $\mathcal{N}_k(x)$ .
- Jednostavnom procedurom potresa iz  $k$ -te okoline se slučajno bira rešenje

---

**Algorithm 1: Shaking procedure**

---

```
Function Shake( $x, k, \mathcal{N}$ );  
choose  $x' \in \mathcal{N}_k(x)$  at random;  
return  $x'$ 
```

---

# VNS

## 3. Neighborhood change step – promena okoline

- *Donošenje odluke koja okolina će biti pretraživana i koje rešenje će biti prihvaćeno*
- *Sekvencijalna promena okoline*

---

### Algorithm 2: Sequential neighborhood change step

---

```
Procedure Neighborhood_change_sequential( $x, x', k$ )  
  if  $f(x') < f(x)$  then  
    |  $x \leftarrow x'$ ;  
    |  $k \leftarrow 1$ ;  
  else  
    |  $k \leftarrow k + 1$ ;  
  end
```

---

# VNS

3. Neighborhood change step – promena okoline
  - *Ciklična promena okoline*

---

**Algorithm 3: Cyclic neighborhood change step**

---

```
Procedure Neighborhood_change_cyclic( $x, x', k$ )  
   $k \leftarrow k + 1$ ;  
  if  $f(x') < f(x)$  then  
    |  $x \leftarrow x'$ ;  
  end
```

---

# VNS

3. Neighborhood change step – promena okoline
  - „Cevasta“ promena okoline

---

**Algorithm 4: Pipe neighborhood change step**

---

```
Procedure Neighborhood_change_pipe( $x, x', k$ )  
  if  $f(x') < f(x)$  then  
    |  $x \leftarrow x'$ ;  
  else  
    |  $k \leftarrow k + 1$ ;  
  end
```

---

# VNS

3. Neighborhood change step – promena okoline
  - „Iskrivljena“ promena okoline

---

## Algorithm 5: Skewed neighborhood change step

---

```
Procedure Skewed_Neighborhood_change( $x, x', k, \alpha$ )  
if  $f(x') - f(x) < \alpha d(x', x)$  then  
|  $x \leftarrow x'$ ;  
|  $k \leftarrow 1$ ;  
else  
|  $k \leftarrow k + 1$ ;  
end
```

---

# VNS

2. Improvement procedure
  - *Lokalna pretraga*

---

**Algorithm 6:** Local search using the first improvement search strategy.

---

Function LS\_FI( $x, N$ )

repeat

    Let  $N(x) = \{x^1, \dots, x^p\}$ ;

$i \leftarrow 0$ ;

$x' \leftarrow x$ ;

    repeat

$i \leftarrow i + 1$ ;

        if  $f(x^i) < f(x)$  then

$x \leftarrow x^i$ ;

            break;

        end

    until  $i = p$ ;

until  $f(x') \leq f(x)$ ;

return  $x'$ ;

---

---

**Algorithm 7:** Local search using the best improvement search strategy.

---

Function LS\_BI( $x, N$ )

repeat

$x' \leftarrow x$ ;

$x \leftarrow \operatorname{argmin}_{y \in N(x')} f(y)$ ;

until  $f(x') \leq f(x)$ ;

return  $x'$ ;

---



# VNS

## 2. Improvement procedure – popravljanje rešenja

- *Lokalna pretraga*
- *Variable neighborhood descent procedures (sekvencijalni, ciklični, cevasti, ugnježden, mešoviti)*
  - Ukoliko je rešenje lokalni minimum za više okolina, onda je dobar kandidat za globalni minimum
- *Simulirano kaljenje, Tabu pretraga, Iterativna lokalna pretraga, Genetski algoritam, GRASP,...*

# Osnovni VNS

---

## Algorithm 12: Basic Variable Neighborhood Search.

---

```
Function Basic_VNS( $x, k_{max}, \mathcal{N}, N$  )
repeat
   $k \leftarrow 1$ ;
  while  $k \leq k_{max}$  do
     $x' \leftarrow \text{Shake}(x, k, \mathcal{N})$ ;
     $x'' \leftarrow \text{Local\_search}(x', N)$ ;
    Neighborhood_change_sequential( $x, x'', k$ );
  end
until stopping condition is fulfilled;
return  $x$ ;
```

---

# VNS - Varijante

- Fiksirana pretraga okoline = Iterativna lokalna pretraga
- Redukovani VNS – bez korake popravljanja rešenja
- Opšti VNS – umesto LS koristi VND
- „Iskrivljeni“ VNS – koristi „iskrivljenu“ promenu okoline
- Ugnježdeni VNS – VNS umesto LS

# VNS – napredne varijante

- Variable neighborhood decomposition search
- Primal-dual VNS
- VNS for nonlinear optimization (Gaus, Nelder-Mead umesto LS)
- VNS for Mixed Integer Non-Linear Programs
- Variable formulation space search
- ...
- Parallel VNS
- Hybrids

# VNS za SAT

1. Reprezentacija rešenja – vektor bitova
2. Funkcija cilja – broj klauza koje su zadovoljene pri interpretaciji
3. Inicijalno rešenje – slučajno generisano rešenje
  - *Radi unapređenja inicijalizacije može se slučajno generisati više potencijalnih rešenja, pa za inicijalno izabrati najbolje*
4. Okolina –  $k$ -tu okolinu predstavljaju vektori koji se od posmatranog rešenja razlikuju u  $k$  bitova
5. Procedura potresa – generisati rešenje koje se od trenutnog razlikuje u  $k$  bitova

# VNS za SAT

6. Lokalna pretraga – u datoj okolini traži se rešenje sa boljom funkcijom cilja
  - *Radi efikasnosti broje se klauze koje nisu zadovoljene i onog trenutka kada njihov broj prevaziđe broj kod trenutnog rešenja, rešenje se odbacuje*
7. Promena okoline
  - *Sekvencijalna*
  - *Ciklična*
  - *Cevasta*

# VNS za SAT

Назив примера	Број варијабли у формули	Број клауза у формули	Индекси искоришћених примера
uf20-91	20	91	[1,..., 10]
uf50-218	50	218	[1,..., 10]
uf75-325	75	325	[1,..., 10]
uf100-430	100	430	[1,..., 10]
uf150-645	150	645	[1,..., 5]

# VNS za SAT

Назив	Опис	Тип	Могуће вредности
maxK	Максимална вредност коју $k$ (околина) може да има.	Цео број	$x \in [2,5], x \in \mathbb{Z}$
searchMethod	Метода локалне претраге: 1. Локална претрага првог побољшања 2. Локална претрага најбољенг побољшања	Цео број	$x \in [1,2], x \in \mathbb{Z}$
neighborhoodMethod	Метода промене околине: 1. Секвенцијална промена околине 2. Кружна промена околине 3. Цеваста (енг. <i>Pipe</i> ) промена околине	Цео број	$x \in [1,3], x \in \mathbb{Z}$
maxDuration	Максимално дозвољено извршавање изражено у микросекундама.	Цео број	$x = 15000000$



# VNS za SAT

maxK	searchMet hode	neighborho odMethod	maxDuratio n	Просечни проценат задовољен их клауза	Укупно задовољен их формула	Просечно време извршава ња
4	2	2	15000000	0.994452	53	11718800
3	2	1	15000000	0.994429	54	11658400
3	2	3	15000000	0.994428	63	11169500
3	2	2	15000000	0.994335	59	11441200
4	2	3	15000000	0.994231	51	12000700
4	2	1	15000000	0.993968	59	11820500
5	2	3	15000000	0.993294	46	12512500
5	2	1	15000000	0.993251	48	12350200
5	2	2	15000000	0.993108	47	12581600
4	1	3	15000000	0.991789	31	12977900
4	1	1	15000000	0.991721	31	13015700

# VNS za PSAT

$k$	<i>best</i>	Просечно време извршавања	Просек решених примера	Процент решених примера
9	YES	2251.12637	27.61111111	92.04%
10	YES	2627.999259	27.55555556	91.85%
8	YES	2209.338981	27.27777778	90.93%
7	YES	1494.471259	26.72222222	89.07%
6	YES	853.9062778	26.33333333	87.78%
9	NO	1869.03063	26.33333333	87.78%
10	NO	1780.79537	26.05555556	86.85%
5	YES	609.2723889	25.83333333	86.11%
8	NO	1394.446667	25.5	85.00%
4	YES	511.7403148	25.11111111	83.70%
6	NO	451.4740185	24.66666667	82.22%
7	NO	865.3665185	24.61111111	82.04%
3	YES	354.6732963	24.55555556	81.85%
5	NO	275.220037	24.22222222	80.74%
2	YES	326.4263889	24.16666667	80.56%
1	YES	322.7114815	23.66666667	78.89%
4	NO	172.932963	23.05555556	76.85%