

Metaheurísticas

Variable Neighborhood Search (VNS)

Felipe Augusto Lima Reis

felipe.reis@ifmg.edu.br



**INSTITUTO
FEDERAL**
Minas Gerais

Sumário

1 Introdução

VARIABLE NEIGHBORHOOD SEARCH (VNS)

- A Busca em Vizinhança Variável (VNS)¹ é um método de busca local proposto por Nenad Mladenović e Pierre Hansen em 1997 [Souza, 2011] [Mladenović and Hansen, 1997];
- O método explora o espaço de soluções “por meio de trocas sistemáticas de estruturas de vizinhança” [Souza, 2011]
 - O VNS não segue uma trajetória bem definida, mas explora vizinhos mais distantes da solução atual, na tentativa de gerar melhores resultados.

¹ VNS corresponde ao acrônimo de Variable Neighborhood Search.

- O VNS é inspirado em métodos métricos variáveis para solução de problemas de otimização contínua irrestrita
 - As primeiras publicações do tema surgiram no final dos anos 1950 e início dos anos 1960;
 - Os métodos buscavam variar a métrica (e, conseqüentemente, a vizinhança) a cada iteração de forma que a direção da busca se adapte melhor à função [Hansen et al., 2008];

- Segundo [Hansen et al., 2008], o VNS baseia-se fortemente em 3 observações:
 - ① Um mínimo local com respeito a uma estrutura da vizinhança não é necessariamente um mínimo local para outra estrutura da vizinhança;
 - ② Um mínimo global é um mínimo local com respeito a todas as estruturas de vizinhança possíveis;
 - ③ Para muitos problemas, os mínimos locais com respeito a uma ou várias vizinhanças são relativamente próximos uns dos outros².

²Segundo [Hansen et al., 2008], essa observação é empírica.

- O VNS foi projetado, originalmente, para solução aproximada de problemas de otimização combinatória
 - Foi estendido, ao longo dos anos, para solução de programação inteira, linear e não linear [Gendreau and Potvin, 2019].
- É composto de duas fases: [Gendreau and Potvin, 2019]
 - ① Fase Decrescente³: busca encontrar pontos de mínimos;
 - ② Fase de Perturbação⁴: as soluções são “pertubadas” (variadas) de modo a escapar desses pontos de mínimos locais .

³Tradução de “Descent Phase”, cujo objetivo corresponde a obter o valor mínimo em um problema de minimização [Gendreau and Potvin, 2019].

⁴Denominado nos algoritmos, muitas vezes, como *shaking* [Hansen et al., 2008] [Avanthay et al., 2003].

- O VNS original utiliza um algoritmo denominado **Variable Neighborhood Descent (VND)**⁵ [Gendreau and Potvin, 2019]
 - Esse algoritmo realiza mudanças nos vizinhos próximos de forma determinística;
 - Na versão original, o VNS fazia uso do VND para busca local [Souza, 2011] [Hansen et al., 2008].

```

Function VND ( $x, k'_{max}$ );
1 repeat
2    $k \leftarrow 1$ ;
3   repeat
4      $x' \leftarrow \arg \min_{y \in \mathcal{N}'_k(x)} f(y)$  /* Find the best neighbor in  $\mathcal{N}'_k(x)$  */;
5     NeighbourhoodChange ( $x, x', k$ ) /* Change neighbourhood */;
   until  $k = k'_{max}$ ;
until no improvement is obtained;

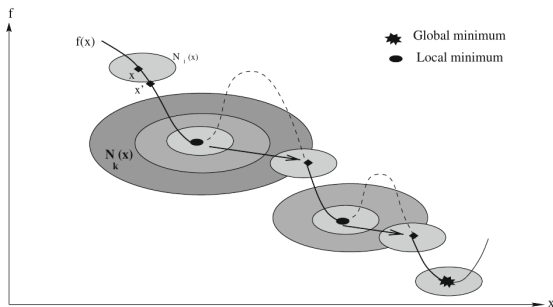
```

Fonte: [Hansen et al., 2008]

⁵Tradução direta: Vizinhança Decrescente Variável.

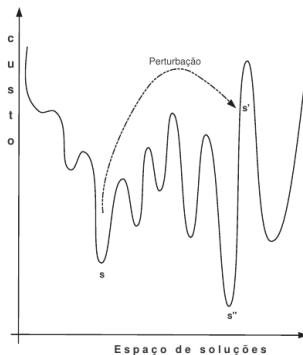
- No VNS, o algoritmo parte uma solução inicial e seleciona vizinhos aleatoriamente a cada iteração
 - Um vizinho s' é selecionado na vizinhança $N^{(k)}(s)$ da solução s corrente;
 - Uma busca local é feita nesse vizinho s' ;
 - Se s' for melhor que s , a busca recomeça na vizinhança de s' ;
 - Caso contrário, uma busca é feita na vizinhança $N^{(k+1)}(s)$;
 - O procedimento é encerrado quando um critério de parada é atingido [Souza, 2011].

- A imagem abaixo ilustra o funcionamento do VNS básico.



Fonte: [Hansen et al., 2008]

- A imagem abaixo ilustra o mecanismo de perturbação da solução no VNS.



Fonte: [Souza, 2011]

Em [Souza, 2011], a imagem ilustra o processo de perturbação no algoritmo ILS. Porém, a ideia por trás da perturbação da solução atual é semelhante nos algoritmos ILS e VNS.

ALGORITMO

- O pseudo-algoritmo simplificado do VNS pode ser visto abaixo:

1. Initialisation

- Determine an initial solution s
- Set $t = 1$

2. Repeat the following until a stopping condition is met

- 2.a *Shaking*. Generate a point s' at random from the t th neighborhood of s ($s' \in N^{(t)}(s)$)
 - 2.b *Local search*. Apply some local search method with s' as initial solution; let s'' be the so obtained local optimum
 - 2.c *Move or not*. If s'' is better than the incumbent s , move there (i.e. set $s = s''$), and continue the search with $N^{(1)}$ (i.e. set $t = 1$); otherwise set $t = (t \bmod t_{\max}) + 1$
-

Fonte: Adaptado de [Avanthay et al., 2003]

Algoritmo

- O pseudo-algoritmo VNS pode ser visto abaixo:

```

procedimento VNS()
1  Seja  $s_0$  uma solução inicial;
2  Seja  $r$  o número de estruturas diferentes de vizinhança;
3   $s \leftarrow s_0$ ;           {Solução corrente}
4  enquanto (Critério de parada não for satisfeito) faça
5       $k \leftarrow 1$ ;       {Tipo de estrutura de vizinhança corrente}
6      enquanto ( $k \leq r$ ) faça
7          Gere um vizinho qualquer  $s' \in N^{(k)}(s)$ ;
8           $s'' \leftarrow \text{BuscaLocal}(s')$ ;
9          se ( $f(s'') < f(s)$ )
10             então
11                  $s \leftarrow s''$ ;
12                  $k \leftarrow 1$ ;
13             senão
14                  $k \leftarrow k + 1$ ;
15         fim-se;
16     fim-enquanto;
17 fim-enquanto;
18 Retorne  $s$ ;
fim VNS;
  
```

Fonte: [Souza, 2011]

VANTAGENS E DESVANTAGENS

Vantagens e Desvantagens

- **Vantagens:**
 - Fácil de ser implementado;
 - Pode ser utilizado como base para construção de métodos mais sofisticados;
 - Pode ser usado em conjunto com outros algoritmos.
- **Desvantagens:**
 - Algoritmo pode iterar repetidas vezes em vizinhanças ruins;
 - Pode não explorar efetivamente o espaço de soluções, uma vez que pode ficar preso em pequenas melhorias na vizinhança.

Referências I



Avanthay, C., Hertz, A., and Zufferey, N. (2003).
A variable neighborhood search for graph coloring.
European Journal of Operational Research, 151(2):379–388.



Gendreau, M. and Potvin, J.-Y. (2019).
Handbook of Metaheuristics (Third Edition).
Springer, Cham, 3 edition.



Hansen, P., Mladenović, N., and Pérez, J. A. M. (2008).
Variable neighbourhood search: methods and applications.
4OR, 6(4):319–360.



Mladenović, N. and Hansen, P. (1997).
Variable neighborhood search.
Computers & Operations Research, 24(11):1097–1100.



Souza, M. J. F. (2011).
Inteligência computacional para otimização.
[Online]; acessado em 12 de Maio de 2021. Disponível em: <http://www.decom.ufop.br/prof/marcone/ Disciplinas/InteligenciaComputacional/InteligenciaComputacional.pdf>.