

Búsqueda de Entorno Variable

2003-11-05

Pierre Hansen, Nenad Mladenović y José A. Moreno Pérez

INTRODUCCIÓN

La Búsqueda de Entorno Variable (conocida por sus siglas en inglés VNS de Variable Neighbourhood Search) es una metaheurística reciente para resolver problemas de optimización cuya idea básica es el cambio sistemático de entorno dentro de una búsqueda por entornos. Las Metaheurísticas son estrategias generales para diseñar procedimientos heurísticos para resolver un problema de optimización mediante un proceso de búsqueda en el espacio de soluciones. Los procesos de búsqueda heurística están generalmente basados en transformaciones de las alternativas que determinan una estructura de entornos en el espacio de soluciones. La VNS es una metaheurística que está basada en un principio simple cambiar la estructura de entornos cuando la búsqueda local se estanca en un óptimo local. Se han realizado muchas extensiones, principalmente para permitir la solución de problemas de gran tamaño, pero en la mayoría de ellas, se ha hecho un esfuerzo por mantener la simplicidad del esquema básico.

1. ESQUEMAS FUNDAMENTALES

Un problema de optimización consiste en encontrar, dentro de un conjunto X de soluciones factibles, la que optimiza una función $f(x)$. Si el problema es de minimización se formula como sigue:

$$\min \{ f(x) \mid x \in X \} \quad (1)$$

donde x representa una "solución" alternativa, f es la función "objetivo" y X es el "espacio" de soluciones factibles del problema. Una solución "óptima" x^* (o mínimo global) del problema es una solución factible donde se alcanza el mínimo de (1).

Una estructura de "entornos" en el espacio de soluciones X es una aplicación N que asocia a cada solución x del espacio X un subconjunto de soluciones $N(x)$ denominado entorno de x , cuyas soluciones se dicen "vecinas" de x . Las metaheurísticas de búsqueda local aplican una transformación o movimiento a la solución de búsqueda y por tanto utilizan, explícita o implícitamente, una estructura de entornos. Denotemos por $\{N_1, N_2, \dots, N_k, \dots, N_{k_{\max}}\}$, a un conjunto finito de estructuras de entornos en el espacio X . Los entornos N_k pueden ser inducidos por una o más métricas introducidas en el espacio de soluciones X . La mayoría de las heurísticas de búsqueda local usan sólo una estructura de entornos.

Una solución x^* de X es un "mínimo global" del problema (1) si no existe otra

solución x de X tal que $f(x) < f(x^*)$. Decimos que la solución x' es un "mínimo local" con respecto a N_k , si no existe otra solución x en $N_k(x')$ tal que $f(x) < f(x')$. Una búsqueda local descendente cambia la solución actual por otra solución mejor de su entorno, por tanto se queda atascada en un mínimo local. Las metaheurísticas basadas en procedimientos de búsqueda local aplican distintas formas de continuar la búsqueda después de encontrar el primer óptimo local.

La VNS está basada en tres hechos simples:

1. Un mínimo local con una estructura de entornos no lo es necesariamente con otra.
2. Un mínimo global es mínimo local con todas las posibles estructuras de entornos.
3. Para muchos problemas, los mínimos locales con la misma o distinta estructura de entornos están relativamente cerca.

Esta última observación, que es empírica, implica que los óptimos locales proporcionan información acerca del óptimo global. Puede ser, por ejemplo, que ambas soluciones tengan características comunes. Sin embargo, generalmente no se conoce cuales son esas características. Es procedente, por tanto, realizar un estudio organizado en las proximidades de este óptimo local, hasta que se encuentre uno mejor.

Los hechos 1 a 3 sugieren el empleo de varias estructura de entornos en las búsquedas locales para abordar un problema de optimización. El cambio de estructura de entornos se puede realizar de forma determinística, estocástica, o determinística y estocástica a la vez.

1.1. Búsqueda de entorno variable descendente

Una búsqueda local determina iterativamente una mejor solución del entorno de la solución actual. La clásica búsqueda "greedy" descendente consiste en reemplazar la solución actual por la mejor solución de su entorno, mientras se produzca mejora. Si se realiza un cambio de estructura de entornos de forma determinística cada vez que se llega a un mínimo local, se obtiene el método de "búsqueda de entorno variable descendente" (Variable Neighbourhood Descent, VND). Los pasos de la VND con estrategia greedy se muestran a continuación.

Pasos de la VNS Descendente

Iniciación:

- Seleccionar el conjunto de estructuras de entornos N_k , $k = 1, \dots, k_{\max}$, que se usarán en el descenso;
- Encontrar una solución inicial x ;

Iteraciones:

- Repetir, hasta que no se obtenga mejora, la siguiente secuencia:

(1) Hacer $k = 1$;

(2) Repetir, hasta que $k = k_{\max}$, los pasos:

(a) Exploración del entorno: Encontrar la mejor solución x' del k -ésimo entorno de x , $N_k(x)$;

(b) Moverse o no: Si la solución obtenida x' es mejor que x , reemplazar x por x' y poner $k = 1$; en otro caso, hacer incrementar k por $k = k + 1$.

La solución final proporcionada por el algoritmo es un mínimo local con respecto a todas las k_{\max} estructuras de entornos, y por tanto la probabilidad de alcanzar un mínimo global es mayor que usando una sola estructura. La mayoría de las heurísticas de búsqueda local usan en sus descensos simplemente un entorno y algunas veces dos ($k_{\max} = 2$).

Además del orden "secuencial" de las estructuras de entornos en el VND anterior, se puede desarrollar una estrategia "anidada". Supongamos, por ejemplo, que $k_{\max} = 3$. Entonces una posible estrategia anidada es: ejecutar la VND de la figura 1 para las dos primeras estructuras de entornos, sobre cada x' que pertenezca al tercer entorno de x ; es decir a $N_3(x)$.

1.2. Búsqueda de entorno variable reducida

El método "reducido de búsqueda de entorno variable" (RVNS) se obtiene si se seleccionan soluciones aleatorias de $N_k(x)$, sin aplicarles a continuación un descenso. Los pasos de la RVNS se presentan de la forma siguiente.

Pasos de la VNS Reducida

Inicialización:

- Seleccionar el conjunto de estructuras de entornos N_k , $k = 1, \dots, k_{\max}$, que se usarán en la búsqueda;

- Encontrar una solución inicial x ;

- Elegir una condición de parada;

Iteraciones:

- Repetir, hasta que se cumpla la condición de parada, la siguiente secuencia:

- (1) Hacer $k = 1$;
 - (2) Repetir, hasta que $k = k_max$, los pasos:
 - (a) Agitación: Generar al azar una solución x' del k -ésimo entorno de x $N_k(x)$;
 - (b) Moverse o no: Si la solución obtenida x' es mejor que x , reemplazar x por x' y hacer $k = 1$; en otro caso, hacer $k = k + 1$.
-

La RVNS es útil para instancias muy grandes de problemas en las que la búsqueda local es muy costosa. Se observa que el mejor valor para el parámetro k_max es frecuentemente 2. Como condición de parada se usa generalmente el máximo número de iteraciones entre dos mejoras.

1.3. Búsqueda de entorno variable básica

El método "básico de búsqueda de entorno variable" (BVNS) combina cambios determinísticos y aleatorios de estructura de entornos. Los pasos de la VNS básica se muestran a continuación.

Pasos de la VNS Básica

Inicialización:

- Seleccionar un conjunto de estructuras de entornos N_k , $k = 1, \dots, k_max$, que se usarán en la búsqueda;
- Encontrar una solución inicial x ;
- Elegir una condición de parada;

Iteraciones:

- Repetir, hasta que se cumpla la condición de parada, la siguiente secuencia:

- (1) Hacer $k = 1$;
 - (2) Repetir, hasta que $k = k_max$, los pasos:
 - (a) Agitación: Generar al azar una solución x' del k -ésimo entorno de x $N_k(x)$;
 - (b) Búsqueda Local: Aplicar algún método de búsqueda local con x' como solución inicial; denótese con x'' el mínimo local así obtenido.
 - (c) Moverse o no: Si la solución obtenida x'' es mejor que x , reemplazar x por x'' y hacer $k = 1$; en otro caso, hacer $k = k + 1$.
-

La condición de parada puede ser, por ejemplo, el máximo tiempo de CPU permitido, el máximo número de iteraciones, o el máximo número de iteraciones entre dos mejoras. Frecuentemente los entornos N_k sucesivos están anidados. Obsérvese que la solución x' se genera al azar en el paso (2a) para evitar el ciclado, que puede ocurrir si se usa cualquier regla determinística.

1.4. Búsqueda de entorno variable general

La búsqueda local del paso (2b) de la BVNS puede ser sustituida por la VND. El método de búsqueda resultante aplica dos series (posiblemente distintas) de estructuras de entornos; una para la agitación y otra para el descenso. Los pasos de este método "general de búsqueda de entorno variable" (GVNS) se muestran a continuación.

Pasos de la VNS General

Inicialización:

- Seleccionar un conjunto de estructuras de entornos N_k , para $k = 1, \dots, k_{\max}$, que se usarán en la agitación;
- Seleccionar un conjunto de estructuras de entornos N'_j , para $j = 1, \dots, j_{\max}$, que se usarán en el descenso;
- Encontrar una solución inicial x ;
- Elegir una condición de parada;

Iteraciones:

- Repetir, hasta que se cumpla la condición de parada, la siguiente secuencia:

(1) Hacer $k = 1$;

(2) Repetir, hasta que $k = k_{\max}$, los pasos:

(a) Agitación: Generar al azar una solución x' del entorno $N_k(x)$;

(b) Búsqueda Local: Aplicar la VND con las estructuras de entornos N'_j , para $j = 1, \dots, j_{\max}$; denótese con x'' la solución así obtenida.

(c) Moverse o no: Si la solución obtenida x'' es mejor que x , reemplazar x por x'' y hacer $k = 1$; en otro caso, hacer $k = k + 1$.

El uso de la búsqueda de entorno variable general (GVNS) ha dado lugar a las aplicaciones más exitosas aparecidas recientemente.

2. EXTENSIONES

Se han propuesto en la literatura diversas formas de extender la VNS para dotarlo de algunas características adicionales. Describimos en primer lugar varias sencillas extensiones de la VNS obtenidas como versiones de la VNS básica. A continuación se describen tres extensiones que constituyen mejoras prácticas de la VNS que han permitido resolver con éxito problemas muy grandes: la búsqueda de entorno variable con descomposición (VNDS), la búsqueda de entorno variable sesgada (SVNS) y la búsqueda de entorno variable paralela (PVNS).

2.1. Extensiones básicas

Las primeras extensiones se derivan directamente de la VNS básica. La BVNS es un método descendente de la primera mejora con aleatorización. Sin mucho esfuerzo adicional se transforma en un método ascendente-descendente: en el paso (2c) hacer también $x = x''$ con alguna probabilidad, incluso si la solución es peor que la actual (o que la mejor solución encontrada hasta el momento). También se puede transformar en un método de la mejor mejora: aplicar un movimiento al mejor entorno k^* entre todos los k_{\max} entornos. Otras extensiones de la VNS son encontrar la solución x' en el paso (2a) como la mejor entre b (un parámetro) soluciones generadas aleatoriamente en el k -ésimo entorno, o introducir k_{\min} y k_{paso} , dos parámetros que controlan el proceso de cambio de entorno: en el algoritmo VNS básico, en vez de $k = 1$ hacer $k = k_{\min}$ y en vez de $k = k + 1$ hacer $k = k + k_{\text{paso}}$.

2.2. Búsqueda de entorno variable con descomposición

El método de "búsqueda de entorno variable con descomposición" (VNDS de Variable Neighbourhood Decomposition Search) extiende la VNS en un esquema de entorno variable en dos niveles basado en la descomposición del problema. Sus pasos son los presentados a continuación.

Pasos de la VNS con Descomposición

Iniciación:

- Seleccionar un conjunto de estructuras de entornos N_k , para $k = 1, \dots, k_{\max}$, que se usarán en el descenso;
- encontrar una solución inicial x ;
- Elegir una condición de parada;

Iteraciones:

- Repetir, hasta que se cumpla la condición de parada, la siguiente secuencia:

(1) Hacer $k = 1$;

(2) Repetir, hasta que $k = k_{\max}$, los pasos:

(a) Agitación: Generar al azar una solución x' del k -ésimo entorno de x $N_k(x)$; denotemos por y el conjunto de los atributos de la solución presentes en x' pero no en x ($y = x' \setminus x$).

(b) Búsqueda Descompuesta: Buscar el óptimo local en el espacio de y por inspección o por alguna heurística; denótese por y' a la mejor solución encontrada y por x'' a la correspondiente solución en el espacio completo X ($x'' = (x' \cup y) \cup y'$);

(c) Moverse o no: Si la solución obtenida x'' es mejor que x , reemplazar x por x'' y hacer $k = 1$; en otro caso, hacer $k = k + 1$.

Nótese que la única diferencia entre la VNS básica y la VNDS está en el paso (2b): en vez de aplicar algún método de búsqueda local en el espacio completo X (empezando desde x' de $N_k(x)$), en la VNDS se resuelve en cada iteración un subproblema en un subespacio X_k incluido en $N_k(x)$ con x' de X_k . Cuando la búsqueda local utilizada en este paso es también la VNS, aparece un esquema VNS en dos niveles.

La VNDS puede verse como la inclusión del esquema de aproximación sucesiva, que ha sido históricamente muy utilizada en optimización combinatoria, en la estructura de la VNS.

2.3. Búsqueda de entorno variable sesgada

Una vez que se ha encontrado la mejor solución en una gran región es necesario alejarse bastante para obtener una mejor. El método de "búsqueda de entorno variable sesgada" (SVNS de Skewed Variable Neighbourhood Search) afronta la exploración de valles alejados de la solución actual. Las soluciones generadas al azar en entornos muy lejanos pueden diferenciarse substancialmente de la solución actual; por lo que la VNS degenera, en algún sentido, en una heurística de arranque múltiple (en la que se realizan iterativamente descensos desde soluciones generadas al azar). Por tanto, la VNS sesgada incorpora una compensación por la distancia desde la solución actual para evitar este inconveniente. Sus pasos son presentados a continuación.

Pasos de la VNS Sesgada

Inicialización:

- Seleccionar un conjunto de estructuras de entornos N_k , para $k = 1, \dots, k_{\max}$, que se usarán en la búsqueda;
- Encontrar una solución inicial x y su valor $f(x)$;
- Hacer $x^* = x$, $f^* = f(x)$;
- Elegir una condición de parada y un parámetro a ;

Iteraciones:

- Repetir, hasta que se cumpla la condición de parada, la siguiente secuencia:

(1) Hacer $k = 1$;

(2) Repetir, hasta que $k = k_{\max}$, los pasos:

(a) Agitación: Generar al azar una solución x' del k -ésimo entorno de x $N_k(x)$;

(b) Búsqueda local: Aplicar algún método de búsqueda local con x' como solución inicial; denótese por x'' al óptimo local obtenido;

(c) Mejorar o no: Si $f(x'') < f^*$ hacer $f^* = f(x'')$ y reemplazar x^* por x'' ;

(d) Moverse o no: Si $f(x'') - a d(x, x'') < f(x)$ reemplazar x por x'' y $k = 1$; en otro caso, hacer $k = k + 1$.

La SVNS usa una función $d(x, x'')$ para medir la distancia entre la solución actual x y el óptimo local encontrado x'' . La distancia usada para definir los entornos N_k puede también utilizarse con este propósito. La elección del parámetro a debe permitir la exploración de valles lejanos a x cuando $f(x'')$ es algo mayor que $f(x)$, pero no demasiado (en otro caso siempre se abandonaría la solución x) y evitar movimientos frecuentes desde x a una solución cercana para volver a x . Un buen valor para a tiene que determinarse experimentalmente en cada caso. Se pueden realizar elecciones más sofisticadas para la función de $a(x, x'')$ mediante algún proceso de aprendizaje.

2.4. Búsqueda de entorno variable paralela

Los métodos de "búsqueda de entorno variable paralela" (PVNS de Parallel Variable Neighbourhood Search) constituyen la tercera extensión. Se han propuesto diversas formas de paralelizar la VNS que han sido aplicadas al problema de la p -mediana. Tres de estas posibilidades son: (i) paralelizar la búsqueda local, (ii) aumentar el número de soluciones obtenidas del entorno actual y realizar búsquedas locales en paralelo desde cada una de ellas y (iii) hacer lo mismo que en (ii) pero actualizando la información sobre la mejor solución encontrada. La segunda paralelización, cuyos pasos se muestran a continuación, es la que ha dado mejores resultados.

Pasos de la VNS Paralela

Inicialización:

- Seleccionar un conjunto de estructuras de entornos N_k , $k = 1, \dots, k_{\max}$, que se usarán en la búsqueda;
- Encontrar una solución inicial x ;
- Elegir una condición de parada;

Iteraciones:

- Repetir, hasta que se cumpla la condición de parada, la siguiente secuencia:

(1) Hacer $k = 1$;

(2) Repetir en paralelo, hasta que $k = k_{\max}$, para cada procesador p los pasos:

(a) Agitación: Generar al azar una solución x'_p del k -ésimo entorno de x $N_k(x)$;

(b) Búsqueda Local: Aplicar algún método de búsqueda local con x'_p como solución inicial; denótese con x''_p el mínimo local así obtenido.

(c) Moverse o no: Si la solución obtenida x''_p es mejor que x , hacer $x = x''_p$ y $k = 1$; en otro caso, hacer $k = k + 1$.

Se ha mostrado que asignando diferentes entornos a cada procesador e interrumpiendo su trabajo tan pronto como se mejore la solución da muy buenos resultados.

3. HÍBRIDOS

Dado que el cambio sistemático de estructura de entornos es una herramienta simple y muy potente, otra forma de extender la VNS ha sido incorporarla a otras metaheurísticas. Estas propuestas han dado lugar a diversas metaheurísticas híbridas.

3.1. VNS y TS

La búsqueda tabú (TS de Tabu Search) generalmente usa una estructura de entornos con respecto a la que ejecuta movimientos de ascenso y descenso explotando diferentes tipos de memoria. En principio hay dos maneras de hacer híbridos de VNS y TS: usar algún tipo de memoria para orientar la búsqueda dentro de VNS o usar VNS dentro de TS. Varios híbridos de este tipo han

aparecido en la literatura.

3.2. VNS y GRASP

La metaheurística GRASP (de Greedy Randomized Adaptive Search Algorithm) consta de dos fases; en la primera fase se construyen soluciones usando un procedimiento greedy aleatorizado y en la segunda, las soluciones se mejoran por alguna búsqueda local o un método enumerativo. Una forma natural de hibridizar VNS con GRASP aplicado en la literatura es usar VNS en la segunda fase de GRASP.

3.3. VNS y MS

La búsqueda Multi-arranque (MS de MultiStart Search) es una metaheurística clásica que, para evitar el estancamiento de un procedimiento descendente en un óptimo local, sencillamente reinicia la búsqueda desde otra solución. Una metaheurística híbrida que combina VND y MS consiste en reiniciar la VND desde otra solución generada al azar del espacio X , cuando se estanca en una solución que es mínimo local con respecto a todos los entornos N_1, \dots, N_{k_max} .

Enlaces

<http://vnsheuristic.ull.es>

<http://www.gerad.ca/en/publications/>

Bibliografía

Mladenovic N, and Hansen P. (1997) "Variable Neighbourhood Search" *Computers and Operations Research*, 24 pp. 1097-1100. (también publicado en *Les Cahiers du GERAD G-96-49* (1996))

Hansen, P. and Mladenovic, N. (2001) "Variable Neighborhood Search: Principles and Applications" *European Journal of Operational Research*, 130 pp. 449-467.

Hansen, P. and Mladenovic, N. (2001) "Variable Neighbourhood Search" en *Handbook of Applied Optimization*, P. Pardalos and M. Resende (eds.), Oxford University Press, 2001.

Hansen, P. and Mladenovic, N. (2001) "Developments of variable neighborhood search" en *Essays and Surveys in Metaheuristics*, pp. 415-440, C. Ribeiro and P. Hansen (eds.), Kluwer, 2001.

Hansen, P. and Mladenovic, N. (2001) "Industrial Applications of Variable Neighbourhood Search" en *Decision and Control in Management Science*, pp. 261-274, G. Zaccour (ed.), Kluwer, 2001.

Hansen, P. and Mladenovic, N. (2003) "Variable Neighbourhood Search" Capítulo 6 en *Handbook of Metaheuristics*, F. Glover and G. Kochenberger, Kluwer (2003).

Pierre Hansen, Nenad Mladenović y José A. Moreno Pérez (2003) "Búsqueda de Entorno Variable". *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, Número 19, Volumen 2. Páginas 77-92.