

METAHEURISTICAS

Ideas, Mitos, Soluciones

Irene Loiseau

2do Cuatrimestre 2017

Qué es un problema de optimización combinatoria?

Dado un conjunto finito E y una función de costo

$$c: 2^E \rightarrow \mathbb{R}$$

encontrar $S^* \in F$ tal que $c(S^*) \leq c(S) \quad \forall S \in F$, o sea
tal que el valor de $c(S^*)$ sea mínimo en F

donde $F \subseteq 2^E$ es el conjunto de soluciones válidas del problema.

F es un conjunto discreto de soluciones con un número finito de elementos

Ejemplo: Problema del viajante de comercio

E: conjunto de ejes

F: subconjuntos de E que forman un circuito hamiltoniano.

$$c(S) = \sum_{e \in S} c_e \quad \text{con} \quad c_e: \text{costo del eje } e.$$

Ejemplo: Problema de la mochila

E: conjunto de items

F: subconjuntos de E que satisfacen la restricción de capacidad de la mochila

$$\sum_{e \in S} a_e \leq b$$

$$c(S) = \sum_{e \in S} c_e$$

c_e : beneficio del elemento e

a_e : peso del elemento e

b: capacidad o peso de la mochila

Otros ejemplos de problemas de optimización combinatoria:

- *Problema de la suma de subconjuntos*
- *Determinación de caminos mínimos en grafos*
- *Flujo en redes*
- *Asignación de tareas*
- *Problemas de ruteo de vehículos. El problema del Viajante de comercio es un caso particular*
- *Diseño de redes de comunicaciones*
- *Ruteo en redes de comunicaciones*
- *VLSI*

- *Planificación de tareas*
- *Asignación de recursos y horarios en instituciones educativas*
- *Asignación de tripulaciones en líneas aéreas*
- *Minimizaron de desperdicios en el corte de materiales*
- *Localización de plantas*
- *Planificación financiera*
- *Problemas de energía*
- *Problemas de Biología Computacional (secuenciamiento de ADN, árboles filogenéticos, doblamiento de proteínas)*
- *etc.*

Cómo se modela matemáticamente un problema de optimización combinatoria?

Minimizar (o maximizar) $f(x)$

$$\begin{array}{ll} \text{sujeto a} & g(x_i) \geq b_i \quad i=1, \dots, m_k \\ & h(x_i) = c_i \quad i= m_{k+1}, \dots, M \\ & x_i \in Z \end{array}$$

-
- función objetivo
 - variables de decisión
 - restricciones
-
- *En la mayoría de los problemas que veremos en el curso f , g y h son funciones lineales.*
 - *No siempre se puede modelar exactamente así un problema de optimización combinatoria*

Cómo se modela como problema de programación lineal entera un problema de optimización combinatoria?

- Problema de la mochila
- Problema de flujo máximo
- Problema del viajante de comercio.

Cómo se resuelve un problema de optimización combinatoria?

- *Enumeración completa o “algoritmo de fuerza bruta”. Sirve?*

COMPLEJIDAD COMPUTACIONAL

Qué hacer?

- SOLUCIONES EXACTAS
- HEURISTICAS

HEURISTICAS

- Heurísticas clásicas
- Metaheurísticas o heurísticas “modernas” o sistemas inteligentes

Cuándo usarlas?

- Problemas para los cuales no se conocen “buenos” algoritmos exactos
- Problemas difíciles de modelar

Porqué usarlas?

- Adaptabilidad a modificaciones de los datos o del problema una vez que ya se obtuvo un resultado.
- Fáciles de implementar y programar
- Basadas en tener una gran capacidad de cálculo
- No sólo para problemas de optimización combinatoria

Cómo se evalúan?

- problemas test
- problemas reales
- problemas generados al azar
- cotas inferiores

ESQUEMA GENERAL DE UN ALGORITMO DE DESCENSO (O BUSQUEDA LOCAL)

S = conjunto de soluciones

$N(s)$ = soluciones “vecinas” de la solución s

Elegir una solución inicial $s_0 \in S$

Repetir

Elegir $s \in N(s_0)$ tal que $f(s) < f(s_0)$

Reemplazar s_0 por s

Hasta que $f(s) > f(s_0)$ para todos los $s \in N(s_0)$

Cómo se representan las soluciones de un problema?

*Cómo determinar las soluciones vecinas de una solución
s dada?*

Qué se obtiene con este procedimiento? Sirve?

Optimos locales y globales

Espacio de búsqueda

Problema de asignación de tareas:

Supongamos que tenemos el problema de asignar tareas a una sola máquina de modo a minimizar el tiempo total de ejecución.

Cada trabajo j tiene un tiempo de procesamiento p_j y una fecha de entrega d_j . El objetivo es entonces minimizar

$$T = \sum_j \max \{ (C_j - d_j), 0 \}$$

donde C_j es el momento en que se completa el trabajo j .

Problema de asignación de tareas:

Supongamos que tenemos el problema de asignar tareas a una sola máquina de modo a minimizar el tiempo total de ejecución.

Cada trabajo j tiene un tiempo de procesamiento p_j y una fecha de entrega d_j . El objetivo es entonces minimizar

$$T = \sum_j \max \{ (C_j - d_j), 0 \}$$

donde C_j es el momento en que se completa el trabajo j .

Como elegir las soluciones iniciales?. A priori se puede tomar cualquier permutación de las tareas.

Determinación de los vecinos de una solución dada: en este caso podemos tomar los que se obtengan de la solución actual cambiando la posición de un trabajo con otro.

En un problema con 4 trabajos por ejemplo los vecinos de (1,2,3,4) serán:

$$N(s) = \{(1,3,2,4), (3,2,1,4), (1,2,4,3), \\ (1,4,3,2), (2,1,3,4), (4,2,3,1)\}$$

Problema del viajante de comercio (TSP)

Dado un grafo G con longitudes asignadas a los ejes queremos determinar un circuito hamiltoniano de longitud mínima.

No se conocen algoritmos polinomiales para resolver el problema del viajante de comercio.

Tampoco se conocen algoritmos ε -aproximados polinomiales para el TSP general (si se conocen cuando las distancias son euclidianas)

Es “el” problema de optimización combinatoria más estudiado.

Heurística del vecino más próximo (heurística golosa)

Empezar

Elegir un nodo v cualquiera de G

Poner $l(v) = 0$

Inicializar $i = 0$

Mientras haya nodos sin marcar hacer:

poner $i = i + 1$

*elegir el eje (v, w) “más barato” tal que w no esté
marcado.*

poner $l(w) = i$

poner $v = w$

Fin

Cuál es la complejidad de este algoritmo?.

Heurísticas de mejoramiento

Cómo podemos mejorar la solución obtenida por alguna heurística constructiva?

Heurística 2-opt de Lin y Kernighan

- *Obtener una solución inicial H (o sea un circuito hamiltoniano H), por ejemplo con alguna heurística.*
- *Mientras sea posible hacer:*
 - . Elegir dos ejes de G tal que al sacarlos de H y reemplazarlos por los dos necesarios para reconstruir un nuevo circuito hamiltoniano H' obtengamos un H' de longitud menor a la de H .*
 - . $H = H'$*
- . Fin*

Cuándo para este algoritmo?. Se obtiene la solución óptima del TSP de este modo?

Algoritmo de búsqueda local

- En vez de elegir para sacar de H un par de ejes que nos lleve a obtener un circuito de menor longitud podemos elegir el par que nos hace obtener el menor H' entre todos los pares posibles. (*más trabajo computacional*)
- Esta idea se extiende en las heurísticas k -opt donde se hacen intercambios para cualquier k . O sea en vez de sacar dos ejes, sacamos k ejes de H y vemos cual es la mejor forma de reconstruir el circuito. En la práctica se usa sólo 2-opt o 3-opt.

METAHEURISTICAS

NO HAY UNA DEFINICION UNICA PARA ESTE TERMINO

- *“Una metaheurística es un conjunto de conceptos que pueden ser usados para definir algoritmos heurísticos para un amplio espectro de problemas diferentes”.*
- *“Las metaheurísticas son estrategias de alto nivel que guían una heurística específica del problema a resolver para mejorar su performance”*

Principales características de las metaheurísticas:

- Las metaheurísticas son estrategias que guían un proceso de búsqueda.
- Las metaheurísticas no son técnicas para un problema específico. Sus conceptos básicos se pueden describir con un alto nivel de abstracción.
- El objetivo es explorar eficientemente el espacio de búsqueda para encontrar soluciones óptimas o casi óptimas.
- Las técnicas metaheurísticas van desde algoritmos simples de búsqueda local a complejos procesos de aprendizaje.
- Las metaheurísticas son en muchos casos algoritmos no-determinísticos.
- Las metaheurísticas pueden usar conocimiento del dominio específico de aplicación manejando heurísticas controladas por ellas.
- Algunas metaheurísticas hace uso de la “memoria” de la búsqueda para guiar los pasos futuros.

TECNICAS METAHEURISTICAS

- Simulated annealing (primeros trabajos 1953, 1983)
- Tabú Search (primeras aplicaciones a optimización combinatoria en 1986, basado en algunas ideas de los 70)
- Algoritmos genéticos y evolutivos (primeras ideas en los 60, mayormente aplicaciones a problemas de IA).
- BRKGA (biased random keys genetic algorithms).
- GRASP (1989)
- Colonia de hormigas (1992),
- VNS (Variable Neighborhood Search)
- Iterated Local Search
- Scatter Search and Path Relinking
- Honey-bee mating optimization
- Particle Swarm Optimization
- Sistemas Inmunes Artificiales
- Redes neuronales (primeras ideas en los 60, resurgieron en los 80)
- **Etc.**
- Híbridos

- *Origen, motivación, exceso de nomenclatura, similitudes “forzadas” con problemas de la física y la biología por ejemplo, etc.*
- *Metaheurísticas de una solución y metaheurísticas de población*
- *Se usan en otros problemas, que no son de optimización combinatoria también.*
- *Se mencionan como “Sistemas inteligentes” en software comercial.*

BIBLIOGRAFIA

- *De que tratan los libros de metaheurísticas?*
- *De que tratan los papers?:*
- *“nuevas ideas” para las metaheurísticas tradicionales.*
- *nuevas metaheurísticas*
- *aplicaciones a una enorme cantidad de problemas*

REVISTAS:

- Journal of Heuristics, Kluwer
- Revistas de Operations Research, Sistemas Inteligentes, Inteligencia artificial, etc.

Congresos

Otros cursos y tutoriales de Metaheurísticas:

- "Metaheuristics: from Design to implementation", E. Talbi <http://www.lifl.fr/~talbi/metaheuristic/>
- "Introducción a los algoritmos metaheurísticos", F. Herrera
- "Metaheurísticas", C. Ribeiro, M. Poggi de Aragao

(bajarlos de la página de la materia)

Framework para metaheurísticas:

- PARADISEO <http://paradiseo.gforge.inria.fr/>

Bibliografía

- Aarts,E.,Korst,J.,"Simulated Annealing and Boltzmann Machines", Wiley, 1989.
- Aarts,E.,Lenstra,J.,(eds),"Local Search in Combinatorial Optimization", Wiley, 1997.
- Alba,E.,"Parallel Metaheuristics: A New Class of Algorithms", Wiley, 2005.
- Bonabeu,E., Dorigo, M.,Theraulaz, G:, "Swarm Intrelligence; from natural to artificial systems", Oxford Univesity Press,1999.
- Blum, C. Roli., A., "Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison", ACM Computing Suveys, vol 35 no 3, 2003, pp 268-308.
- Corne, D., Dorigo,M., Glover,F.,(eds)"New ideas in Optimization", McGraw Hill, 1999.
- Davis,L.(ed),"Handbook of genetic algorithms", Reinhold, 1991.
- Dorigo, M and Stutzle., "Ant colony optimization",MIT Press, 2004.
- Gendreau,M., Potvin,J., "Handbook of Metaheuristics", Springer, 2010

- Glover, F., De Werra, D., (eds), "Tabu search", Annals of Operations Research 41, Baltzer, 1993.
- Glover, F., Laguna, M., "Tabu Search", Kluwer Academic Pub., 1997.
- Glover, F., Kochenberger, G., "Handbook of Metaheuristics", Kluwer Academic Pub., 2003.
- Goldberg, D. "Genetic Algorithms in Search, Optimization and Machine learning", Addison-Wesley, 1989.
- Haupt, R., Haupt, S., "Practical Genetic Algorithms", Willey, 1998.
- Hertz, A., "Les Méta-heuristiques: quelques conseils pour en faire bon usage", en "Gestion de Production e Ressources Humaines: méthodes de planification dans les systèmes productifs", 2005, pp 205-222.
- Hertz, J., Krog, A., Palmer, R., "Introduction to the theory of neural computacion", Addison Wesley, 1991.
- Laporte, G., Osman, I., (eds), "Metaheuristics in Combinatorial Optimization", Annals of Operations Research 63, Baltzer, 1996.
- Michalewicz, Z., "Genetic algorithms + Data Structures = Evolution programs", Springer Verlag, 1996.

- Mitchell,M., "An introduction to genetic algorithms (complex adaptive systems)", MIT Press, 1996.
- Osman,I.H., Kelly,J.,(eds) "Metaheuristics: theory and applications", Kluwer Academic Pub., 1996.
- Pham,D.,Karaboga,D." Intelligent Optimization Techniques: Genetic Algorithms, Tabu Search, Simulated Annealing, and Neural Networks", Springer Verlag, 1998.
- Rayward-Smith,V.J., Osman,I.H., Reeves,C.R., "Modern Heuristic Search Methods", Wiley, 1996.
- Reeves, C. (ed), "Modern heuristics techniques for combinatorial Problems", Blackwell,1993.
- Resende, M., Ribeiro, C., "Optimization by GRASP", Springer, 2016.
- Talbi, E.G. "Metaheuristics:from design to implementation", Wiley, 2009.
- Van Laarhoven,P., Aarts,E. "Simulated Annealing: theory and applications", Kluwer, 1988.
- Voss, Martello, Osman, Rocairol, "Metaheuristics", Kluwer, 1999.