Local search (optimization)

From Wikipedia, the free encyclopedia

This article includes a list of references, related reading or external links, **but its sources remain unclear because it lacks inline citations.** Please improve this article by introducing more precise citations. *(May 2015)*

In computer science, **local search** is a metaheuristic method for solving computationally hard optimization problems. Local search can be used on problems that can be formulated as finding a solution maximizing a criterion among a number of candidate solutions. Local search algorithms move from solution to solution in the space of candidate solutions (the *search space*) by applying local changes, until a solution deemed optimal is found or a time bound is elapsed.

Local search algorithms are widely applied to numerous hard computational problems, including problems from computer science (particularly artificial intelligence), mathematics, operations research, engineering, and bioinformatics. Examples of local search algorithms are WalkSAT and the 2-opt algorithm for the Traveling Salesman Problem.

Contents [hide]

1 Examples
2 Description
3 See also
3.1 Real-valued search-spaces
4 Bibliography

Examples [edit]

Some problems where local search has been applied are:

1. The vertex cover problem, in which a solution is a vertex cover of a graph, and the target is to find a solution with a minimal number of nodes
2. The travelling salesman problem, in which a solution is a cycle containing all nodes of the graph and the target is to minimize the total length of the cycle
3. The boolean satisfiability problem, in which a candidate solution is a truth assignment, and the target is to maximize the number of clauses satisfied by the assignment; in this case, the final solution is of use only if it satisfies *all* clauses
4. The nurse scheduling problem where a solution is an assignment of nurses to shifts which satisfies all established constraints
5. The k-medoid clustering problem and other related facility location problems for which local search offers the best known approximation ratios from a worst-case perspective

Description [edit]

Most problems can be formulated in terms of search space and target in several different manners. For example, for the travelling salesman problem a solution can be a cycle and the criterion to maximize is a combination of the number of nodes and the length of the cycle. But a solution can also be a path, and being a cycle is part of the target.

A local search algorithm starts from a candidate solution and then iteratively moves to a neighbor solution. This is only possible if a neighborhood relation is defined on the search space. As an example, the neighborhood of a vertex cover is another vertex cover only differing by one node. For boolean satisfiability, the neighbors of a truth assignment are usually the truth assignments only differing from it by the evaluation of a variable. The same problem may have multiple different neighborhoods defined on it; local optimization with neighborhoods that involve changing up to k components of the solution is often referred to as **k-opt**.

Typically, every candidate solution has more than one neighbor solution; the choice of which one to move to is taken using only information about the solutions in the neighborhood of the current one, hence the name *local* search. When the choice of the neighbor solution is done by taking the one locally maximizing the criterion, the

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store
Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page
Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page
Print/export
Create a book
Download as PDF
Printable version
Languages
Deutsch
فارسی
Français
עברית
日本語
Русский
ไทย
Українська
中文
Edit links

metaheuristic takes the name hill climbing. When no improving configurations are present in the neighborhood, local search is stuck at a locally optimal point. This local-optima problem can be cured by using restarts (repeated local search with different initial conditions), or more complex schemes based on iterations, like iterated local search, on memory, like reactive search optimization, on memory-less stochastic modifications, like simulated annealing.

Termination of local search can be based on a time bound. Another common choice is to terminate when the best solution found by the algorithm has not been improved in a given number of steps. Local search is an anytime algorithm: it can return a valid solution even if it's interrupted at any time before it ends. Local search algorithms are typically approximation or incomplete algorithms, as the search may stop even if the best solution found by the algorithm is not optimal. This can happen even if termination is due to the impossibility of improving the solution, as the optimal solution can lie far from the neighborhood of the solutions crossed by the algorithms.

For specific problems it is possible to devise neighborhoods which are very large, possibly exponentially sized. If the best solution within the neighborhood can be found efficiently, such algorithms are referred to as very large-scale neighborhood search algorithms.

See also [edit]

Local search is a sub-field of:

- Metaheuristics
- Stochastic optimization
- Optimization

Fields within local search include:

- Hill climbing
- Simulated annealing (suited for either local or global search)
- Tabu search
- Reactive search optimization (combining machine learning and local search heuristics)

Real-valued search-spaces [edit]

Several methods exist for performing local search of real-valued search-spaces:

- Luus–Jaakola searches locally using a uniform distribution and an exponentially decreasing search-range.
- Random optimization searches locally using a normal distribution.
- Random search searches locally by sampling a hypersphere surrounding the current position.
- Pattern search takes steps along the axes of the search-space using exponentially decreasing step sizes.

Bibliography [edit]

- Battiti, Roberto; Mauro Brunato; Franco Mascia (2008). *Reactive Search and Intelligent Optimization*. Springer Verlag. ISBN 978-0-387-09623-0.
- Hoos, H.H. and Stutzle, T. (2005) Stochastic Local Search: Foundations and Applications, Morgan Kaufmann.
- Vijay Arya and Naveen Garg and Rohit Khandekar and Adam Meyerson and Kamesh Munagala and Vinayaka Pandit, (2004): Local Search Heuristics for *k*-Median and Facility Location Problems, Siam Journal of Computing 33(3).
- Juraj Hromkovič: Algorithmics for Hard Problems: Introduction to Combinatorial Optimization, Randomization, Approximation, and Heuristics (Springer)
- Wil Michiels, Emile Aarts, Jan Korst: Theoretical Aspects of Local Search (Springer)

This page was last modified on 8 May 2015, at 22:26.

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.

Privacy policy About Wikipedia Disclaimers Contact Wikipedia Developers Mobile view