# A Local Search Meta-Heuristic for Non-Specialists*

**Serdar Kadioglu and Meinolf Sellmann**


## Abstract

Based on our recent paper [2], we present dialectic search, a local search meta-heuristic inspired by Hegel and Fichte's Dialectic. We illustrate the simplicity and the efficiency of dialectic search on the set covering problem and propose to apply it to the problem of where to place a limited number of sensors in a water distribution network such that the damage incurred by accidental or intentional contamination is minimized.


## 1 Introduction

*Computational Sustainability* is a newly emerging field; its first international conference was held in June, 2009 where challenging problems related to natural resource management, species distribution planning and energy policy models were discussed. Without exception the method of choice of all non-experts in optimization for their problems was simulated annealing which is known to work slowly without sophisticated techniques outside the core method. Therefore, we argue that a simple yet powerful new local search method is needed. To this end, we present *dialectic search* which, by design, draws the users attention to the most important aspects of any efficient local search procedure.

## 2 A Meta-Heuristic Inspired by Philosophy

An inherent problem of local search is the dilemma of having to balance the wish for improving solutions with the need to diversify the search. This is known as the exploitation-exploration trade-off. We find a local search paradigm where exploration and exploitation are tightly connected yet clearly separated from each other in philosophy: Hegel and Fichte's Dialectic. Their concept of intellectual discovery works as follows: The current model is called the *thesis*. Based on it, we formulate an *antithesis* which negates (parts of) the thesis. Finally, we merge thesis and antithesis to achieve the *synthesis*. The synthesis then becomes the new thesis and the process is iterated.

Dialectic strikes an appealing balance between exploration and exploitation. In essence, the formulation of an antithesis enforces search space exploration, while the optimization of thesis and antithesis allows us to exploit and improve. We believe that this makes dialectic search easy to use for non-experts in optimization.

### 2.1 Dialectic Search

A detailed outline of dialectic search meta-heuristic is given in [2]. Briefly, we begin by greedily improving a given assignment (the thesis). Then we try to improve the solution further by generating randomized modifications (an antithesis) of the current assignment, greedily improving it, and then combining the two assignments to form a new assignment, which is also greedily improved (the synthesis). If this new assignment is at least as good, we consider it the new current assignment. If this process does not result in improvements for a while, then the search moves to the modified assignment and continues searching from there.

We need to specify how the thesis is transformed into an antithesis, how an assignment is greedily improved, and how thesis and antithesis are combined to form the synthesis for each individual problem. The contribution of dialectic search is that it manages the balance between exploitation and exploration so that the user can focus on both tasks separately.

## 3 The Set Covering Problem

We evaluate dialectic search paradigm on one of the most studied NP-hard combinatorial optimization problems, the set cover problem (SCP): Given a finite set $S := \{1, \dots, m\}$ of items, and a family $F := \{S_1, \dots, S_n \subseteq S\}$ of subsets of $S$, and a cost function $c : F \to \mathbb{R}^+$, the objective is to find a subset $C \subseteq F$ such that $S \subseteq \bigcup_{S_i \in C} S_i$ and $\sum_{S_i \in C} c(S_i)$ is minimized.

**Initial Solution:** A simple greedy construction for SCP is to pick sets one by one until a cover is found. [7] compare 7 different criteria how the next set is chosen and suggest to choose one of the criteria at random in each step of the greedy construction. Run around 30 times, this randomized approach was reported to yield good solutions, and we use this method to initialize our search.

**Antithesis and Synthesis:** As antithesis, we pick a randomized subset $T$ of the current selection $C$, whereby we choose the size of this subset randomly between one half, one third, and one quarter of the cardinality of $C$. $T$ is empty first and

* This work was supported by the National Science Foundation through the Career: Cornflower Project (award number 0644113).

| | AvgSol | | BestSol | | AvgTime | | |
|---|---|---|---|---|---|---|---|
| Class | TS | Dialectic | TS | Dialectic | TS | Dialectic | Speedup |
| a | **38.66** | 38.74 | **38.4** | 38.6 | 4.3 | **1.78** | 2.4 |
| b | 22.02 | **22.00** | 22 | 22 | 7.02 | **0.49** | 14 |
| c | 43.5 | **43.45** | 43 | 43 | 7.86 | **2.97** | 2.6 |
| d | 25 | **24.81** | 24.8 | **24.4** | 14.4 | **1.07** | 13.4 |
| 4 | **37.92** | 38.20 | **37.7** | 37.8 | **0.67** | 1.63 | 0.4 |
| 5 | 34.36 | **34.28** | 34.1 | 34.1 | 1.87 | **1.85** | 1 |
| 6 | 20.78 | **20.66** | 20.6 | 20.6 | **0.26** | 0.72 | 0.3 |
| nre | 17.14 | **16.98** | 17 | **16.6** | 5.94 | **0.50** | 11.7 |
| nrf | 10.62 | **10** | 10 | 10 | 31.4 | **1.31** | 23.8 |
| nrg | 62.7 | **62.25** | 61.8 | **61.2** | 32.0 | **4.33** | 7.3 |
| nrh | 34.88 | **34.03** | 34 | **33.8** | 22.4 | **3.49** | 6.4 |

**Table 1.** We present the average runtime in seconds for finding the best solution in each run, the average solution quality and the best solution quality. We run dialectic search 50 times on each instance using an AMD Athlon 2.0Ghz machine whereas TS was reported to run 10 times on each instance using a Pentium 4 2.4Ghz machine.

then augmented iteratively by selecting two sets whose removal would leave the fewest items uncovered which are still covered by $C \setminus T$. One of the two sets is chosen uniformly at random and added to $T$. We repeat this until $T$ has the desired size. If $A \leftarrow F \setminus T$ does not cover all items, we greedily add sets in $T$ to $A$ until it is a cover. $A$ becomes our antithesis.

To obtain a synthesis, we conduct a greedy walk from the thesis to the antithesis. This walk consists of two phases. In the first phase, we remove all sets in $C$ that are not part of $A$. In the second phase, we greedily select a set in $A$ which minimizes the cost over newly covered items and repeat until we obtain a cover which is returned as the synthesis.

**Numerical Results:** We consider 70 well-known benchmark instances from the OR library and compare this simple dialectic search with the tabu search, TS, from [1] which was tuned on and for each of the benchmark classes individually. Tables 1 summarizes the aggregated results for each of the different classes. Dialectic search, running with one set of parameters on all instances, clearly outperforms the state-of-the-art on 9 out of 11 classes while running up to 23 times faster, and improves the best known solution over 5% of all instances in one of the best studied benchmark sets in OR

We attribute the effectiveness of this simple algorithm to the fact that in each step both exploration and exploitation play their part, and yet, they can be addressed separately, which makes dialectic search so attractive to non-specialists.

## 4 Sensor Placement in Water Networks

We propose to use dialectic search for the placement of a limited number of sensors in a water distribution network to minimize the impact of accidental or intentional contamination. Strategic placement of sensors reduce the risk of pollution of vulnerable public water distribution systems. September 11 attacks have increased the concerns over terrorist attacks and early detection of contamination events is crucial to mitigate potentially catastrophic public health and economic consequences. A number of approaches have been developed to solve this problem, such as integer programming models and various heuristics [3; 4; 5]. It is notable that [3] and [5] solve the problem via solution of a corresponding set covering problem.

We model a water distribution network as a graph $G(V, E)$, where vertices in $V$ represent junctions, and edges in $E$ represent pipes. We assume network dynamics can be obtained by water simulation tools (e.g. EPANET [6]) as a set of steady-state flow patterns. Since we do not know a priori at which node an injection will occur, we consider a set of contamination scenarios, $A$, where each attack $a \in A$ has a likelihood $\alpha_a$ such that $\sum_{a \in A} \alpha_a = 1$. For each attack scenario, we compute $d_{ai}$, the impact of attack $a$ if it is witnessed by a sensor located at node $i$, using water analysis tools. The objective is to minimize the aggregate impact over all attack scenarios. We propose a simple dialectic search for the problem.

**Initial Solution and Greedy Improvement:** We obtain an initial solution by placing each sensor to a node that would minimize the objective the most. As greedy improvement heuristic, we consider changing the placement of each sensor and compute the objective delta that would result from the replacement. We commit the change that would decrease the objective the most and iterate until no more improvement is possible.

**Antithesis and Synthesis:** First, we determine randomly the fraction of sensors that must be replaced. Then, we compute an antithesis by iteratively picking a sensor whose removal would have the least impact on the objective value and assigning it to a random node which is not protected by a sensor in the current assignment. As synthesis, we return the best solution found while moving from thesis to antithesis in this iterative way.

**Numerical Results:** Execution of water simulations require non-trivial computation and the evaluation of the aforementioned dialectic search is our ongoing work. Our aim is to have preliminary results ready for the workshop.

## References

1. N. Musliu. Local Search Algorithm for Unicost Set Covering Problem. *IEA/AIE*, 302–311, 2006
2. S. Kadioglu and M. Sellmann. Dialectic Search. *To Appear in CP 2009*, 2009.
3. A. Kessler, A. Ostfeld, G. Sinai. Detecting Accidental Contaminations in Municipal Water Networks. *Journal of Water Resources Planning and Management*, 124(4):192-198, 1998.
4. A. Kumar, M.L. Kansal, G. Arora. Discussion of 'Detecting Accidental Contaminations in Mmunicipal Water Networks'. *Jo of Water Resources Planning and Management*, 125(4):308-310, 1999.
5. A. Ostfeld and E. Salomons. Optimal Layout of Early Warning Detection Stations for Water Distribution Systems Security. *Journal of Water Resources Planning and Management*, 130(5):377-385, 2004.
6. L.A. Rossman. The EPANET Programmer's Toolkit for Analyis of Water Distribution Systems. *In Proceedings of the Annual Water Resources Plannnig and Management Conference*, 1999.
7. F.J. Vasko and G.R. Wilson. An Efficient Heuristic for Large Set Covering Problems. *Naval Research Logistics Quarterly*, 31:163–171, 1984.