CS4234: Optimisation Algorithms Tutorial 9

SLS Part 2; Mini-Project Last Discussion

V1.6: Steven Halim *October 31, 2022*

Preliminaries

During Lecture 8, 9, 10, and the previous Tutorial 8, you have been exposed with a new search paradigm: (Stochastic) Local Search (SLS) followed by 4 (FOUR) more established SLS algorithms (also called '**Meta**[1]-heuristics'): SA, TS, ILS, MA, that are reported to be successful in attacking various (NP-)hard COPs A, B, C, ... in various research papers.

By now, we have seen two and a half weeks worth of SLS ideas. Hopefully you have realized that when one is presented with a new (NP-)hard COP Z, or even COP A again but with different constraints (e.g., the classic TSP/MWVC but with very limited run time limit), one cannot simply take 'any favorite' SLS algorithm from a book/lecture note/research paper/one's own experience and apply that SLS algorithm verbatim with 'default parameters' on problem Z and hopes to get a good result out of the box. In this tutorial, we will apply what we have learned in Lecture 10 and/or in our Mini Project experimentations so far. This is the last moment where you can discuss Mini Project ideas freely before we freeze the results for this semester on Thursday, 03 November 2022, 11.59am.

Discussion Points

Q1: Statements About SLS (up to Lecture 10) For each statement below about Stochastic Local Search (SLS) algorithm, determine if it is More Towards True/More Towards False/It depends and give a short explanation.

1. **All** SLS algorithms, if run for **extremely long time**, e.g., $\approx \infty$, will **always** encounter a GO of a COP instance during its long search run although it cannot stop immediately after encountering such GO (see the previous statement).

2. We can make *any SLS algorithm* for Metric No-Repeat TSP to have a 2-approximation ratio.

[1]This word 'meta' was suddenly become very popular LAST year (precisely on 28 October 2021) due to `https://about.facebook.com/meta/`.

3. Hybrid SLS algorithms (that combines two, or more, simpler SLS algorithms) is **always better** than its individual SLS algorithm working individually on its own.

4. Tabu Search (TS) algorithm is a better SLS algorithm than Simulated Annealing (SA).

5. In Tabu Search algorithm, setting high Tabu Tenure value/setting encourages diversification search strategy.

6. If we use Tabu Search for TSP, the best parameter setting for Tabu Tenure is a fixed constant 7, i.e., that is, forbid the last 7 local moves that Tabu Search has just performed.

Q2: In slide 8 of `https://www.comp.nus.edu.sg/~stevenha/cs4234/lectures/10.SLS-DTP.pdf`, Steven has outlined the list of potential parameters (type-1 of SLS DTP), components (type-2 of SLS DTP), and search strategies (type-3 of SLS DTP) of Tabu Search (TS) Meta-heuristic. Now please do the same for Iterated Local Search (ILS) Meta-heuristic (assuming that you use ILS for the TSP). You can refer to `https://www.comp.nus.edu.sg/~stevenha/cs4234/lectures/09.Meta-heuristics.pdf` for the **bold red text** parts of ILS or other resources to give a more complete view.

Q3: Past paper (AY2019/20) hidden MCQs:

1. Which statement about Stochastic Local Search (SLS) algorithm is correct?

   (a) Albeit more difficult, we can analyze the worst case time complexity of an SLS algorithm
   (b) SLS algorithm terminates upon finding Global Optima
   (c) We should use SLS algorithm when we are given an NP-hard optimization problem
   (d) We can do pre-processing to make any SLS algorithm for (M-NR-)TSP has 2-approximation bound
   (e) It is easy to design a good SLS algorithm for a given NP-hard optimization problem

2. Which statement about Tabu Search is incorrect?

   (a) It has an optional component called Aspiration Criteria
   (b) It uses cooling function
   (c) One of its most important tunable parameter is Tabu Tenure
   (d) Lowering Tabu Tenure value makes Tabu Search perform more intensification
   (e) It is more efficient to forbid recent local moves instead of recently found solutions

3. Which of the following animal-inspired SLS/metaheuristic has never appeared in at least one scientific article before?

   (a) Ants Colony Optimization
   (b) Bat algorithm
   (c) Cuttlefish optimization algorithm
   (d) Killer Whale algorithm
   (e) Actually, all metaheuristics a-d above have appeared in at least one scientific article before

4. Which part of SLS algorithm engineering process is usually the most time consuming?

   (a) Picking the correct SLS algorithm
   (b) Picking the correct components of the chosen SLS algorithm
   (c) Implementing the SLS algorithm into a working program
   (d) Debugging the SLS algorithm
   (e) Fine tuning the SLS algorithm

5. Context: Mini-Project 1: Traveling-Salesman-Problem (TSP) of up to 1000 vertices with limited 2s runtime (single thread; or $2/k$ second if we have k threads). Which of the following SLS/heuristic ideas has the highest chance to work well as discussed in the final project presentations? You can assume that 2-edges-exchange (2-opt) Neighborhood relation is used for all 5 options below.

   (a) Steepest Descent with random restart upon hitting any Local Optima
   (b) Simulated Annealing with aggressive cooling function
   (c) Iterated Local Search that only accepts the better of the two Local Optima after each perturbation+subsidiary local search steps
   (d) Evolutionary/Memetic Algorithm with a medium-size population
   (e) Tabu Search algorithm with high Tabu Tenure to encourage exploration of search space

6. Context: Mini-Project 2: MIN-WEIGHTED-VERTEX-COVER (MWVC) of up to 4000 vertices and 600 000 edges with limited 2s runtime (single thread; or $2/k$ second if we have k threads). Which statement is likely incorrect?

   (a) We can use add vertex to vertex cover and remove vertex from vertex cover local moves

   (b) We need to favor intensification to get a good performing SLS for MWVC

   (c) We can use gain/loss scoring heuristic to help the SLS picks the next local move

   (d) We need to take care of potential *infeasible* solutions that are not a vertex cover

   (e) If the graph is dense, we can complement the input graph and optimizing for MAX-CLIQUE or choose to optimize for MAX-INDEPENDENT-SET instead

Remarks Before we end our CS4234 tutorial sessions this semester, let's take a class photo as momento (for Steven's last TG2, as he is not in Singapore on Monday, 07 November 2022. Audrey's TG1 + TG3 can do this on week 13.