CMA–VNS: A Short Description

Fan Xue and Geoffrey Q.P. Shen*
{fan.xue, bsqpshen}@polyu.edu.hk

July 8, 2015

1 Introduction

The CMA-VNS (Covariance Matrix Adaptation Variable Neighborhood Search) solver is a competitor for the first Combinatorial Black-Box Optimization Competition (CBBOC 2015[1]). CMA-VNS is a hyper-heuristic which employs a CMA-ES (Covariance Matrix Adaptation Evolution Strategy) [Hansen et al., 2003] evolution which is followed by an iterated local search. The reason of employing the CMA-ES is the efficient estimation of variables in CMA-ES. Hence the local search, variable neighborhood search in this context, can be facilitated against expensive (and/or limited) evaluations.

2 Main Procedure

Figure 1 shows the pseudo code of CMA-VNS. There are two main blocks in the pseudo code: CMA block (line 4-7) and VNS block (line 8-15). The proportion of the two block is controlled in a hyper-heuristic mechanism which tested the overall performance of a number of modes before-hand and trained a set of rules from the results.

The CMA block trains a bi-population CMA-ES[2] to predict promising solutions in a given black-box problem. One of the most important parameter LAMBDA is determined by the pre-determined rules. If any new best known solutions are found, they can be appended into the *elite* set except for the premature solutions at very beginning.

The VNS block employs three mechanisms. First, a *backbone* (common bits of *elite* sets, see [Zhang and Looks, 2005]) is constructed to strengthen the intensification of searching. Then an estimated promising solution is generated by CMA-ES for diversification then combined with the *backbone*. At last, a VNS procedure tries to find improvements of the generated solution. If the new result is better than the best known value, the new result is appended to the *elite* set.

*Corresponding author. Tel.: (852) 2766 5817; Addr.: Department of Building and Real Estate, The Hong Kong Polytechnic University, Hunghom, Kowloon, Hong Kong S.A.R., China P.R.

[1] See `http://web.mst.edu/~tauritzd/CBBOC/`.

[2] From LIBCMAES, see `https://github.com/beniz/libcmaes`.

```
procedure CMA–VNS(dimension, evaluations)                    ▷ Note: Main procedure
    params ← configure_by_pre-trained_rules (dimension, evaluations)
    elites ← ∅
    repeat                                                    ▷ Note: CMAES goes first
        solutions ← BIPOPCMAES(params.lambda)
        elites += solutions.new_best_knowns
    until params.cma_eval is met
    repeat                                                    ▷ Note: Then VNS
        backbone ← common_bits(elites)
        s0 ← bipopCMAES.predict with backbone                 ▷ Note: By CMAES and elites
        solution ← VNS (s0)                                   ▷ Note: With a flipping tabu list
        if solution is new best known then
            elites += solution
        end if
    until params.vns_eval is met
end procedure
```

Figure 1: Pseudo code of CMA–VNS

3 Additional features

Solution cache A hashtable cache is employed to map bit vectors to objective function values. Before any evaluation request, the bit vector is tested first in the cache. A successful hit can save time and increase effective evaluations.

Tabu list of flipping In the procedure of VNS, a tabu list of flipping is used to avoid try bit flips which failed recently. When the list usually increases from empty set to all bits quickly, i.e. no bit can be changed. And the list will be cleared once until no improvements found since last clearance. Experiments show that the tabu list makes evaluation of solutions more effective.

Training free The parameters of CMA-VNS are determined by pre-trained rules. The *dimension* and *maximum evaluations* are the two main inputs for using the rules. In addition, the rules are found effective enough without re-running and re-training for each NK problem set.

References

[Hansen et al., 2003] Hansen, N., Müller, S. D., and Koumoutsakos, P. (2003). Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma–es). *Evol. Comput.*, 11(1):1–18.

[Zhang and Looks, 2005] Zhang, W. and Looks, M. (2005). A novel local search algorithm for the traveling salesman problem that exploits backbones. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, IJCAI'05, pages 343–348, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.