Evolutionary algorithm with a directional local search for multiobjective optimization in combinatorial problems

Krzysztof Michalak
Department of Information Technologies, Institute of Business Informatics, Wroclaw University of Economics
Komandorska 118-120
Wroclaw, Poland 53-345
krzysztof.michalak@ue.wroc.pl

ABSTRACT

This abstract summarizes the results reported in the paper [5]. In this paper a new method of performing a local search in multiobjective optimization problems is proposed. The proposed method uses a solution acceptance criterion based on aggregation of the objectives using adaptively adjusted weight vectors. A weight vector for performing the search starting from an initial solution is determined using directions in which objective improvements have been achieved in the vicinity of the initial solution.

In the paper the proposed method is tested on 2-, 3- and 4-objective instances of the Travelling Salesman Problem (TSP) and the Quadratic Assignment Problem (QAP). In the experiments the proposed method outperformed two other local search methods.

The proposed method focuses on solution acceptance criterion and thus can be combined with various methods of solution neighbourhood construction in the local search as well as various global search algorithms.

CCS CONCEPTS

•Mathematics of computing → Evolutionary algorithms; •Applied computing → Multi-criterion optimization and decision-making;

KEYWORDS

Multiobjective optimization, Evolutionary computing, Local search, Decomposition, TSP, QAP

ACM Reference format:
Krzysztof Michalak. 2017. Evolutionary algorithm with a directional local search for multiobjective optimization in combinatorial problems. In *Proceedings of GECCO '17 Companion, Berlin, Germany, July 15-19, 2017,* 2 pages.
DOI: http://dx.doi.org/10.1145/3067695.3084380

1 INTRODUCTION

One of the approaches to finding good solutions in optimization problems is to use local search methods which aim at improving initial solutions of the given problem by exploring their neighbourhoods. There are several components of a local search method, such as the initialization strategy, the structure of the neighborhoods and the strategy used to explore them, etc. [2].

An important element that has to be decided upon is the solution acceptance criterion. In the case of optimization problems with $m > 2$ objectives it is not always obvious which of two given solutions is better. The two most common approaches to address this issue are either to use Pareto dominance for deciding if one solution outperforms the other, or to aggregate the objectives of the solutions into scalar values and compare them.

In this paper an approach is presented in which the objectives are aggregated using weight vectors modified adaptively during the run of the algorithm based on directions in which improvements have been observed in previous iterations.

2 THE DIRECTIONAL LOCAL SEARCH

The Directional Local Search method proposed in this paper uses weighted aggregation to determine if a new solution improves over the existing one. This approach to aggregating the m objectives uses a weight vector λ satisfying the conditions:

$$\forall_{j\in\{1,\ldots,m\}} : \lambda_j \geq 0, \quad \sum_{j=1}^{m} \lambda_j = 1 \ . \tag{1}$$

A scalar objective for a solution x is then obtained, for example, by calculating the weighted sum $\sum_{j=1}^{m} \lambda_j f_j(x)$, or using the Tchebycheff aggregation $\max\limits_{1\leq j\leq m}\{\lambda_j|f_j(x) - z_j^*|\}$, where z_j^* is the utopia point with coordinates equal to the best known values of each objective.

The DirLS builds a list L that keeps track of the improvements observed while solving a multiobjective optimization problem. When a local search is performed around a solution x and an improved solution x' is found, a pair $\langle pos, dir\rangle$ is stored in which $pos = F(x')$ is a vector of the objectives of the new solution x' and dir equals $\Delta F = F(x') - F(x)$ which is the direction in the objective space along which the improvement occurred. Subsequently, the elements of the list L are used to calculate weight vectors used for aggregating the objectives during local search. When local search is to be performed around a solution x_0, N_{dir} elements are found in the list L for which pos elements are the closest to $F(x_0)$. The corresponding dir vectors are averaged and the result is normalized so that the conditions in the equation (1) hold, and in this way a weight vector λ_0 is obtained. The λ_0 vector is then used for aggregating the objectives when performing the local search around x_0. The concepts discussed above are presented in Figure 1.

GECCO '17 Companion, July 15-19, 2017, Berlin, Germany

ACM ISBN 978-1-4503-4939-0/17/07…$15.00
DOI: http://dx.doi.org/10.1145/3067695.3084380

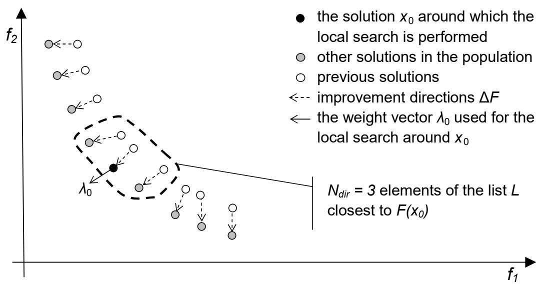

Figure 1: The elements of the DirLS method performing a biobjective minimization with $N_{dir} = 3$.

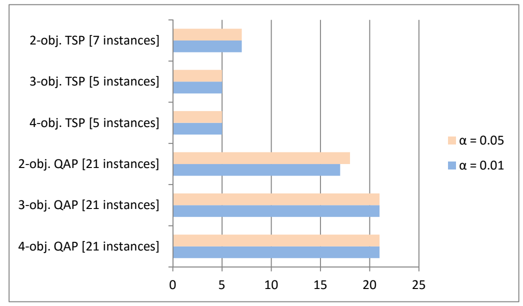

Figure 2: The number of times the DirLS was found to be significantly better than the other methods at the significance levels of $\alpha = 0.05$ and 0.01.

In cases when there are fewer elements in the list L than the number of averaged vectors N_{dir} and also randomly with the probability $1-|L|/N_{pop}$ (where N_{pop} is the population size), the DirLS performs the local search using a weight vector $\lambda^{\#}$ defined as follows. For calculating $\lambda^{\#}$ the nadir point $n^{\#} \in R^m$ is used that consists of the worst values of the objectives found during the search. For a specimen x_0 with the objectives vector $F(x_0) = \{f_1(x_0), \ldots, f_m(x_0)\}$ the weight vector $\lambda^{\#}$ is calculated based on a vector $\delta^{\#}$ of absolute differences between the objective values and nadir point coordinates:

$$\delta^{\#} = [|f_1(x_0) - n_1^{\#}|, \ldots, |f_m(x_0) - n_m^{\#}|] \ . \tag{2}$$

The $\delta^{\#}$ vector represents the direction from the nadir point $n^{\#}$ to the location of the objectives $F(x_0)$. The weights in $\lambda^{\#}$ are calculated by normalizing the coordinates in $\delta^{\#}$:

$$\lambda^{\#} = \delta^{\#} / \sum_{j=1}^{m} \delta_j^{\#} \ . \tag{3}$$

A local search method (DLS) using only weight vectors calculated using equations (2) and (3) is used in this paper as one of the reference methods to which the DirLS is compared.

3 EXPERIMENTS AND RESULTS

To verify the effectiveness of the proposed method, experiments were performed on 2-, 3- and 4-objective instances of the Travelling Salesman Problem (TSP) and the Quadratic Assignment Problem (QAP). In both problems solutions are permutations of a given length n. For the TSP the size of the test instances was $n = 100, \ldots, 400$ and for the QAP $n = 25$, 50 and 75. The number of 2-, 3- and 4-objective instances of the TSP was 7, 5 and 5 respectively and the number of the QAP instances was 21 for each number m of the objectives. Experiments on the biobjective TSP were performed using the kroAB$\langle n \rangle$ TSP instances made available by Thibaut Lust [4]. Experiments on the biobjective QAP were performed using instances with correlated flow matrices introduced in [3]. For three and four objectives three permutations π_2, π_3 and π_4 were randomly generated. For a solution (permutation) π the objective $f_1(\pi)$ was calculated using the kroA$\langle n \rangle$ cost matrix in the TSP and the flow matrix used for the first objective taken from [3] in the QAP. The objectives $f_j(\pi)$ for $j \geq 2$ were calculated as $f_j(\pi) = f_1(\pi_j(\pi))$.

In the experiments the DirLS was compared to the local search using Pareto dominance as a solution acceptance criterion (PLS) and the local search using decomposition (DLS) that was based on weight vectors calculated using equations (2) and (3). All three local search methods were used in conjunction with the NSGA-II as the global optimization algorithm. The local search was performed around the solutions found by the NSGA-II using neighbourhoods constructed using the 2-opt operator [1]. Pareto fronts obtained in the experiments were compared using the hypervolume indicator. The optimization was run 30 times for each algorithm and problem instance and the median hypervolume was calculated. The DirLS attained better hypervolume than the PLS and the DLS in all tests except one test instance of the 2-objective QAP problem with $n = 25$.

Statistical significance of the results was verified using the Wilcoxon test with the null hypothesis that the DirLS method gives the same median hypervolume value as the reference methods PLS and DLS. The number of times the DirLS was found to be significantly better than the other methods at the significance levels of $\alpha = 0.05$ and 0.01 is presented in Figure 2.

4 CONCLUSION

This paper presents the Directional Local Search (DirLS) method that utilizes the knowledge concerning promising search directions to construct weight vectors for the local search. In the experiments the DirLS method has proven to be more effective than the other two local search methods on both tested problems. The DirLS focuses on the solution acceptance criterion and thus can be used with various neighbourhood construction methods as well as different global optimization methods.

REFERENCES

[1] Cheng-Wen Chiang, Wei-Ping Lee, and Jia-Sheng Heh. 2010. A 2-Opt based differential evolution for global optimization. *Applied Soft Computing* 10, 4 (2010), 1200 – 1207.
[2] Arnaud Liefooghe et al. 2012. On dominance-based multiobjective local search: design, implementation and experimental analysis on scheduling and traveling salesman problems. *Journal of Heuristics* 18, 2 (2012), 317–352.
[3] Thomas Stutzle Luis Paquete. 2006. A study of stochastic local search algorithms for the biobjective QAP with correlated flow matrices. *European Journal of Operational Research* 169, 3 (2006), 943–959.
[4] Thibaut Lust. 2015. Multiobjective TSP. https://sites.google.com/site/thibautlust/research/multiobjective-tsp. (2015). Online: accessed 2017.04.03.
[5] Krzysztof Michalak. 2016. Evolutionary algorithm with a directional local search for multiobjective optimization in combinatorial problems. *Optimization Methods and Software* 31, 2 (2016), 392–404.