Local Search Based on a Local Utopia Point for the Multiobjective Travelling Salesman Problem

Krzysztof Michalak

Department of Information Technologies,
Institute of Business Informatics,
Wroclaw University of Economics, Wroclaw, Poland
krzysztof.michalak@ue.wroc.pl

Abstract. Performing a local search around solutions found by an evolutionary algorithm is a common practice. Local search is well known to significantly improve the solutions, in particular in the case of combinatorial problems. In this paper a new local search procedure is proposed that uses a locally established utopia point. In the tests in which several instances of the Travelling Salesman Problem (TSP) were solved using an evolutionary algorithm the proposed local search procedure outperformed a local search procedure based on Pareto dominance. Because the local search is focused on improving individual solutions and the multiobjective evolutionary algorithm can improve diversity, various strategies of sharing computational resources between the evolutionary algorithm and the local search are used in this paper. The results attained by the tested methods are compared with respect to computation time, which allows a fair comparison between strategies that distribute computational resources between the evolutionary optimization and the local search in various proportions.

Keywords: multiobjective optimization, combinatorial optimization, travelling salesman problem, local search

1 Introduction

Combinatorial problems arise in many practical applications what motivates the need for solving them efficiently. Unfortunately, exact algorithms, while guaranteed to find the optimal solution often suffer from the effects of combinatorial explosion, which makes them impractical, especially for large instances of optimization problems. To circumvent this limitation approximate methods are often used. Population-based methods are effective in solving such complex optimization problems and are also well suited for multiobjective optimization, because they maintain an entire population of solutions, which, in a multiobjective case, can approximate a Pareto front for a given problem. For combinatorial problems memetic algorithms are often used, in which a population-based algorithm is coupled with a separate individual learning or local search procedure [6]. Among local search approaches used in multiobjective problems local search procedures based on Pareto dominance are often used [4, 8]. In this paper a comparison is performed between a local search procedure based on a locally determined utopia point (ULS) and a local search procedure based on Pareto dominance (PLS). The results produced by these methods are compared using two commonly used indicators: the hypervolume and the Inverse Generational Distance (IGD). In the experiments performed on several instances of the TSP the ULS procedure produced better results than the Pareto-based procedure given equal time for computations using both methods.

This paper is structured as follows. Section 2 presents the local search procedure based on a local utopia point. In Section 3 the experiments are described and the results are discussed. Section 4 concludes the paper.

2 Local Search Procedures

In this paper a local search procedure based on a locally determined utopia point (ULS) is proposed. We assume, that the objective space is $\mathbb{R}^d$ and that we minimize all the objectives (application to the maximization case is straightforward). Thus, the optimization problem can be formalized as:

$$\begin{aligned} &\text{minimize } F(x) = (f_1(x), \ldots, f_d(x)) \\ &\text{subject to } x \in \Omega, \end{aligned} \tag{1}$$

where:

Ω - the decision space.

The ULS procedure improves a given solution x belonging to a population P as follows. First, a local utopia point $u(x) \in \mathbb{R}^d$ is established by setting each coordinate $u(x)_i$, $i = 1, \ldots, d$ to:

$$u(x)_i = max\left\{f_i(x') : x' \in P - \{x\} \wedge f_i(x') < f_i(x)\right\} \quad . \tag{2}$$

Thus, the obtained utopia point has all its coordinates set to maximum values of objectives represented in the population and less than those of x. If for a given coordinate i there is no $x' \in P - \{x\}$ satisfying $f_i(x') < f_i(x)$ then a preset special value MIN_VALUE is used (in this paper MIN_VALUE = -1 000 000 is used in this case). As presented in Algorithm 1 the ULS method is a best-improvement search, which replaces the current solution x with a new one x' from a neighbourhood $N(x)$ as long as $d(F(x'), u(x)) < d(F(x), u(x))$, where $d(\cdot)$ denotes the Euclidean distance. The neighbourhood $N(x)$ includes all solutions that can be generated from x using the 2-opt operator [1]. The search is performed until no improvement can be obtained.

To make a fair comparison the Pareto-based method used in this paper is also a best-improvement procedure which works on the same neighbourhoods $N(x)$ as the ULS and also stops when no improvement is obtained, but the acceptance criterion is Pareto dominance rather than the reduction of a distance to a given point. Therefore, the entire step 1. in Algorithm 1 is not used in PLS and in step 2.1. the current solution x is replaced if x' dominates x.

Algorithm 1 The ULS procedure.

```
IN:     d - the dimensionality of the objective space
        x - the solution to improve
        P - current population, x ∈ P

OUT:    x - the solution improved by the local search

for i = 1, . . . , d do                              # 1. Establish a local utopia point
    u(x)_i = MIN_VALUE
    for x' ∈ P − {x} do
        if f_i(x') < f_i(x) and f_i(x') > u(x)_i then
            u(x)_i = f_i(x')
        end if
    end for
end for

repeat                                               # 2. Improve the solution
    improved := false
    for x' ∈ N(x) do
        if d(F(x'), u(x)) < d(F(x), u(x)) then      # 2.1. Check if the new solution
            x := x'                                  #      is better than the old one
            improved := true
        end if
    end for
until (not improved)

return x
```

Both local search procedures are used in this paper in conjunction with the NSGA-II algorithm [2] for multiobjective optimization of the TSP. In this paper the inver-over operator [11] is used for generating offspring.

Because balancing the amount of computational resources between the evolutionary optimization and the local search can influence the effectiveness of the algorithm [5, 10], the Pareto-based method was used with several strategies of allocating computational resources to the evolutionary search and to the local search. Two tested approaches **PLS** ($\mathbf{P = 1.0}$) and **PLS** ($\mathbf{P = 0.5}$) involve applying the PLS with probability 1.0 and 0.5 respectively. Thus, in PLS (P=1.0) the local search is performed for every specimen in the population and in PLS (P=0.5) the local search is performed for every specimen with probability 0.5. In the latter approach the local search is performed at half the frequency, which may reduce computational overhead, but also may degrade the quality of the results. Another tested approach is not to perform the local search at all (the **none** strategy). Because the effectiveness of the local search may vary at different stages of the optimization, another two approaches were tested. In the **PLS (asc)** approach the probability of applying the local search to each specimen increases linearly from 0 at the beginning of the optimization process to 1 at the end of the optimization. In the **PLS (desc)** approach the probability of applying the local search is 1 at the beginning and it drops linearly to 0 during optimization. Clearly, in the PLS (asc) the local search is mainly applied to solutions from a population that underwent an evolutionary optimization for longer and thus can be expected to be better starting points. In the PLS (desc) the local search is applied most frequently at the beginning of the evolution and thus has to start with solutions that are far from optimal. On the other hand, in the PLS (desc) strategy the local search can positively influence the initial phase of the optimization.

3 Experiments and Results

In the experiments the ULS method was compared to four variants of the PLS using different strategies of allocating computational resources to the evolutionary algorithm and the local search, and to the evolutionary algorithm without the local search. In this paper bi-objective TSP instances kroAB*nnn* made available by Thibaut Lust [12] were used with the number of cities $nnn = 100$, 150, 200, 300, 400, 500, 750 and 1000. The parameters for the algorithm were set as follows. The population size was set to $N_{pop} = 100$ and the random inverse rate $\eta = 0.02$ was used. Algorithms based solely on the evolutionary optimization and those using local search procedures may achieve very different quality of the results in the same number of generations. Also, varying intensity of local search changes both the timing and the attained results in each generation. To allow a fair comparison the quality of the results obtained by different methods was compared after the same running time of $t = 300$ seconds rather than after a fixed number of generations. Pareto fronts attained by the tested methods within the available time limit were evaluated using the hypervolume [13] and Inverse Generational Distance (IGD) [7] indicators. The IGD indicator requires a set of reference points from which the distance of the obtained solutions is calculated. There are various techniques for obtaining such points, for example, in the case of optimization problems described analytically these points can be calculated directly from the equations describing the problem. In constrained problems one of the possible approaches is to solve a relaxed version of the problem and then use the solutions as idealized reference points. In this paper another approach was followed in which the set of reference points is obtained by taking all the nondominated points from solutions produced by the tested methods for a given problem instance. This approach has the advantage in that it can produce the reference points even for practical problems in which the points cannot be established analytically and no relaxation may be possible because of the complex nature of the problem. For each method and each test set 30 repetitions of the test were performed on a machine with 2.4 GHz Intel Core 2 Quad Q6600 CPU and 4 GB of RAM. From these runs mean, median and standard deviation values of both indicators were calculated and are presented in Tables 1-8.

Results obtained in the experiments were processed using the Wilcoxon rank test [9]. This test was chosen because it does not assume the normality of the distributions, which may be hard to guarantee in case of hypervolume and IGD measurements. It was also recommended in a survey [3] in which methods suitable

Table 1. Hypervolume and IGD values attained for $N = 100$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$1.6023{\cdot}10^{10}$	$1.6196{\cdot}10^{10}$	$7.6683{\cdot}10^{8}$	$5.1648{\cdot}10^{4}$	$5.1055{\cdot}10^{4}$	$3.3267{\cdot}10^{3}$
PLS ($P = 0.5$)	$2.4334{\cdot}10^{10}$	$2.4342{\cdot}10^{10}$	$4.6754{\cdot}10^{8}$	$9.6821{\cdot}10^{3}$	$9.8788{\cdot}10^{3}$	$1.4756{\cdot}10^{3}$
PLS ($P = 1.0$)	$2.4880{\cdot}10^{10}$	$2.4904{\cdot}10^{10}$	$3.4252{\cdot}10^{8}$	$7.7639{\cdot}10^{3}$	$7.5605{\cdot}10^{3}$	$1.1032{\cdot}10^{3}$
PLS (asc)	$2.2010{\cdot}10^{10}$	$2.2146{\cdot}10^{10}$	$8.0293{\cdot}10^{8}$	$1.7071{\cdot}10^{4}$	$1.6824{\cdot}10^{4}$	$2.1464{\cdot}10^{3}$
PLS (desc)	$2.4837{\cdot}10^{10}$	$2.4878{\cdot}10^{10}$	$2.8748{\cdot}10^{8}$	$7.8428{\cdot}10^{3}$	$7.8020{\cdot}10^{3}$	$9.6257{\cdot}10^{2}$
ULS	$2.5601{\cdot}10^{10}$	$2.5832{\cdot}10^{10}$	$6.7932{\cdot}10^{8}$	$3.9768{\cdot}10^{3}$	$2.8019{\cdot}10^{3}$	$2.9563{\cdot}10^{3}$

Table 2. Hypervolume and IGD values attained for $N = 150$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$3.2621{\cdot}10^{10}$	$3.2362{\cdot}10^{10}$	$1.7880{\cdot}10^{9}$	$1.0080{\cdot}10^{5}$	$1.0186{\cdot}10^{5}$	$6.1111{\cdot}10^{3}$
PLS ($P = 0.5$)	$5.4549{\cdot}10^{10}$	$5.4433{\cdot}10^{10}$	$9.6264{\cdot}10^{8}$	$1.7823{\cdot}10^{4}$	$1.7760{\cdot}10^{4}$	$1.9285{\cdot}10^{3}$
PLS ($P = 1.0$)	$5.5720{\cdot}10^{10}$	$5.5644{\cdot}10^{10}$	$8.0822{\cdot}10^{8}$	$1.4891{\cdot}10^{4}$	$1.4926{\cdot}10^{4}$	$1.7976{\cdot}10^{3}$
PLS (asc)	$5.0417{\cdot}10^{10}$	$5.0145{\cdot}10^{10}$	$1.2348{\cdot}10^{9}$	$2.8112{\cdot}10^{4}$	$2.8549{\cdot}10^{4}$	$3.0057{\cdot}10^{3}$
PLS (desc)	$5.5453{\cdot}10^{10}$	$5.5454{\cdot}10^{10}$	$6.0025{\cdot}10^{8}$	$1.5162{\cdot}10^{4}$	$1.5086{\cdot}10^{4}$	$1.2078{\cdot}10^{3}$
ULS	$5.8936{\cdot}10^{10}$	$5.9314{\cdot}10^{10}$	$1.4208{\cdot}10^{9}$	$6.2998{\cdot}10^{3}$	$4.7199{\cdot}10^{3}$	$4.9699{\cdot}10^{3}$

Table 3. Hypervolume and IGD values attained for $N = 200$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$5.2585{\cdot}10^{10}$	$5.2471{\cdot}10^{10}$	$2.4659{\cdot}10^{9}$	$1.4720{\cdot}10^{5}$	$1.4830{\cdot}10^{5}$	$6.5569{\cdot}10^{3}$
PLS ($P = 0.5$)	$9.4544{\cdot}10^{10}$	$9.4447{\cdot}10^{10}$	$1.5599{\cdot}10^{9}$	$2.7332{\cdot}10^{4}$	$2.7231{\cdot}10^{4}$	$2.4114{\cdot}10^{3}$
PLS ($P = 1.0$)	$9.6240{\cdot}10^{10}$	$9.6652{\cdot}10^{10}$	$1.8580{\cdot}10^{9}$	$2.4363{\cdot}10^{4}$	$2.3533{\cdot}10^{4}$	$2.7024{\cdot}10^{3}$
PLS (asc)	$8.7223{\cdot}10^{10}$	$8.7472{\cdot}10^{10}$	$1.9099{\cdot}10^{9}$	$3.9923{\cdot}10^{4}$	$3.9520{\cdot}10^{4}$	$2.9927{\cdot}10^{3}$
PLS (desc)	$9.6239{\cdot}10^{10}$	$9.6039{\cdot}10^{10}$	$1.1415{\cdot}10^{9}$	$2.3962{\cdot}10^{4}$	$2.3868{\cdot}10^{4}$	$1.7321{\cdot}10^{3}$
ULS	$1.0480{\cdot}10^{11}$	$1.0573{\cdot}10^{11}$	$2.7560{\cdot}10^{9}$	$7.5180{\cdot}10^{3}$	$4.5561{\cdot}10^{3}$	$7.8807{\cdot}10^{3}$

Table 4. Hypervolume and IGD values attained for $N = 300$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$9.8025{\cdot}10^{10}$	$9.9744{\cdot}10^{10}$	$5.7400{\cdot}10^{9}$	$2.5395{\cdot}10^{5}$	$2.5251{\cdot}10^{5}$	$8.3855{\cdot}10^{3}$
PLS ($P = 0.5$)	$1.9927{\cdot}10^{11}$	$1.9914{\cdot}10^{11}$	$3.5869{\cdot}10^{9}$	$4.5034{\cdot}10^{4}$	$4.4860{\cdot}10^{4}$	$3.9594{\cdot}10^{3}$
PLS ($P = 1.0$)	$2.0028{\cdot}10^{11}$	$2.0087{\cdot}10^{11}$	$2.2556{\cdot}10^{9}$	$4.3155{\cdot}10^{4}$	$4.2478{\cdot}10^{4}$	$2.5737{\cdot}10^{3}$
PLS (asc)	$1.8516{\cdot}10^{11}$	$1.8466{\cdot}10^{11}$	$2.8529{\cdot}10^{9}$	$6.1549{\cdot}10^{4}$	$6.2227{\cdot}10^{4}$	$3.5908{\cdot}10^{3}$
PLS (desc)	$2.0187{\cdot}10^{11}$	$2.0224{\cdot}10^{11}$	$1.9251{\cdot}10^{9}$	$4.1816{\cdot}10^{4}$	$4.1515{\cdot}10^{4}$	$1.8804{\cdot}10^{3}$
ULS	$2.2953{\cdot}10^{11}$	$2.3295{\cdot}10^{11}$	$5.5661{\cdot}10^{9}$	$1.0190{\cdot}10^{4}$	$3.5550{\cdot}10^{3}$	$1.0179{\cdot}10^{4}$

for evaluating evolutionary and swarm intelligence algorithms were discussed. The results of the statistical tests are presented in Tables 9 and 10 which summarize comparisons of two the best performing methods ULS and PLS (desc) with other algorithms. In Table 9 a sign in a given row and column represents

Table 5. Hypervolume and IGD values attained for $N = 400$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$1.4252 \cdot 10^{11}$	$1.4400 \cdot 10^{11}$	$7.2263 \cdot 10^{9}$	$3.5728 \cdot 10^{5}$	$3.5593 \cdot 10^{5}$	$8.9753 \cdot 10^{3}$
PLS ($P = 0.5$)	$3.2235 \cdot 10^{11}$	$3.2327 \cdot 10^{11}$	$4.5732 \cdot 10^{9}$	$7.1950 \cdot 10^{4}$	$7.1490 \cdot 10^{4}$	$4.0573 \cdot 10^{3}$
PLS ($P = 1.0$)	$3.2250 \cdot 10^{11}$	$3.2260 \cdot 10^{11}$	$2.3309 \cdot 10^{9}$	$7.1416 \cdot 10^{4}$	$7.0661 \cdot 10^{4}$	$2.1962 \cdot 10^{3}$
PLS (asc)	$3.0254 \cdot 10^{11}$	$3.0231 \cdot 10^{11}$	$4.4301 \cdot 10^{9}$	$8.9644 \cdot 10^{4}$	$9.0201 \cdot 10^{4}$	$4.0627 \cdot 10^{3}$
PLS (desc)	$3.2774 \cdot 10^{11}$	$3.2797 \cdot 10^{11}$	$3.8239 \cdot 10^{9}$	$6.6905 \cdot 10^{4}$	$6.6328 \cdot 10^{4}$	$3.1366 \cdot 10^{3}$
ULS	$3.8699 \cdot 10^{11}$	$3.9092 \cdot 10^{11}$	$9.4632 \cdot 10^{9}$	$9.6561 \cdot 10^{3}$	$3.7527 \cdot 10^{3}$	$1.4027 \cdot 10^{4}$

Table 6. Hypervolume and IGD values attained for $N = 500$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$1.9950 \cdot 10^{11}$	$2.0019 \cdot 10^{11}$	$8.5470 \cdot 10^{9}$	$4.7972 \cdot 10^{5}$	$4.7940 \cdot 10^{5}$	$9.7379 \cdot 10^{3}$
PLS ($P = 0.5$)	$4.9077 \cdot 10^{11}$	$4.9095 \cdot 10^{11}$	$3.7994 \cdot 10^{9}$	$8.6907 \cdot 10^{4}$	$8.6609 \cdot 10^{4}$	$2.5111 \cdot 10^{3}$
PLS ($P = 1.0$)	$4.8834 \cdot 10^{11}$	$4.8770 \cdot 10^{11}$	$3.6278 \cdot 10^{9}$	$8.8264 \cdot 10^{4}$	$8.8483 \cdot 10^{4}$	$2.7402 \cdot 10^{3}$
PLS (asc)	$4.6317 \cdot 10^{11}$	$4.6356 \cdot 10^{11}$	$5.5284 \cdot 10^{9}$	$1.0467 \cdot 10^{5}$	$1.0475 \cdot 10^{5}$	$3.6158 \cdot 10^{3}$
PLS (desc)	$4.9502 \cdot 10^{11}$	$4.9455 \cdot 10^{11}$	$3.1035 \cdot 10^{9}$	$8.3402 \cdot 10^{4}$	$8.3579 \cdot 10^{4}$	$1.9820 \cdot 10^{3}$
ULS	$6.0132 \cdot 10^{11}$	$6.0288 \cdot 10^{11}$	$4.0029 \cdot 10^{9}$	$6.9318 \cdot 10^{3}$	$5.8373 \cdot 10^{3}$	$3.4575 \cdot 10^{3}$

Table 7. Hypervolume and IGD values attained for $N = 750$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$3.5962 \cdot 10^{11}$	$3.5960 \cdot 10^{11}$	$1.8593 \cdot 10^{10}$	$7.5376 \cdot 10^{5}$	$7.5480 \cdot 10^{5}$	$1.1816 \cdot 10^{4}$
PLS ($P = 0.5$)	$1.0891 \cdot 10^{12}$	$1.0905 \cdot 10^{12}$	$4.9247 \cdot 10^{9}$	$2.3941 \cdot 10^{5}$	$2.3891 \cdot 10^{5}$	$2.8395 \cdot 10^{3}$
PLS ($P = 1.0$)	$1.0887 \cdot 10^{12}$	$1.0898 \cdot 10^{12}$	$5.5977 \cdot 10^{9}$	$2.3955 \cdot 10^{5}$	$2.4000 \cdot 10^{5}$	$2.9521 \cdot 10^{3}$
PLS (asc)	$1.0529 \cdot 10^{12}$	$1.0507 \cdot 10^{12}$	$1.6065 \cdot 10^{10}$	$2.5560 \cdot 10^{5}$	$2.5590 \cdot 10^{5}$	$7.2508 \cdot 10^{3}$
PLS (desc)	$1.0936 \cdot 10^{12}$	$1.0931 \cdot 10^{12}$	$6.0725 \cdot 10^{9}$	$2.3758 \cdot 10^{5}$	$2.3807 \cdot 10^{5}$	$2.8925 \cdot 10^{3}$
ULS	$1.2795 \cdot 10^{12}$	$1.3003 \cdot 10^{12}$	$6.7021 \cdot 10^{10}$	$5.3185 \cdot 10^{4}$	$3.2315 \cdot 10^{4}$	$5.4241 \cdot 10^{4}$

Table 8. Hypervolume and IGD values attained for $N = 1000$ using different local search methods in 300 seconds running time.

Algorithm	Hypervolume			IGD		
	Mean	Median	Std. dev.	Mean	Median	Std. dev.
none	$5.2986 \cdot 10^{11}$	$5.2949 \cdot 10^{11}$	$1.4122 \cdot 10^{10}$	$1.0302 \cdot 10^{6}$	$1.0302 \cdot 10^{6}$	$1.2143 \cdot 10^{4}$
PLS ($P = 0.5$)	$1.9127 \cdot 10^{12}$	$1.9131 \cdot 10^{12}$	$9.5281 \cdot 10^{9}$	$4.4548 \cdot 10^{5}$	$4.4488 \cdot 10^{5}$	$4.1141 \cdot 10^{3}$
PLS ($P = 1.0$)	$1.9139 \cdot 10^{12}$	$1.9133 \cdot 10^{12}$	$8.3681 \cdot 10^{9}$	$4.4464 \cdot 10^{5}$	$4.4448 \cdot 10^{5}$	$3.8338 \cdot 10^{3}$
PLS (asc)	$1.8727 \cdot 10^{12}$	$1.8731 \cdot 10^{12}$	$2.0848 \cdot 10^{10}$	$4.5905 \cdot 10^{5}$	$4.5878 \cdot 10^{5}$	$6.7310 \cdot 10^{3}$
PLS (desc)	$1.9204 \cdot 10^{12}$	$1.9185 \cdot 10^{12}$	$7.9682 \cdot 10^{9}$	$4.4170 \cdot 10^{5}$	$4.4259 \cdot 10^{5}$	$4.3471 \cdot 10^{3}$
ULS	$2.0328 \cdot 10^{12}$	$2.0865 \cdot 10^{12}$	$2.2730 \cdot 10^{11}$	$1.2851 \cdot 10^{5}$	$9.6405 \cdot 10^{4}$	$9.3073 \cdot 10^{4}$

the result of the Wilcoxon test comparing the ULS with a method specified in the column using the results for the test set specified in the row. The null hypothesis of the statistical test is that the median values of a given indicator are equal for ULS and the compared method. The interpretation of the signs is as follows. The "–" sign is placed in the table when the ULS method attained a worse median value than the compared method. The "+" sign denotes a test in which the ULS method attained a better median value than the compared method and the p-value obtained in the statistical test was not higher than 0.05. If the ULS method attained a better median value than the compared method, but the p-value obtained in the statistical test was higher than 0.05 the "#" sign is inserted, which represents a better performance of the ULS method, but with no confirmed statistical significance. Table 10 summarizes the performance of the PLS (desc) method in the same way as described above. The last row (F-W) in each of the two tables shows, if the family-wise probability of type I error calculated over all test sets is smaller than 0.05 (the "+" sign) or not (the "#" sign). If the method presented in the table produced a worse result for one or more sets the "–" sign is placed in the last row.

Table 9. Results of the statistical analysis comparing hypervolume and IGD values obtained using the ULS method with other tested methods.

N	Hypervolume					IGD				
	none	PLS ($P = 0.5$)	PLS ($P = 1.0$)	PLS (asc)	PLS (desc)	none	PLS ($P = 0.5$)	PLS ($P = 1.0$)	PLS (asc)	PLS (desc)
100	+	+	+	+	+	+	+	+	+	+
150	+	+	+	+	+	+	+	+	+	+
200	+	+	+	+	+	+	+	+	+	+
300	+	+	+	+	+	+	+	+	+	+
400	+	+	+	+	+	+	+	+	+	+
500	+	+	+	+	+	+	+	+	+	+
750	+	+	+	+	+	+	+	+	+	+
1000	+	+	+	+	+	+	+	+	+	+
F-W	+	+	+	+	+	+	+	+	+	+

Table 10. Results of the statistical analysis comparing the results obtained using the PLS (desc) method with other tested methods.

N	Hypervolume					IGD				
	none	PLS ($P = 0.5$)	PLS ($P = 1.0$)	PLS (asc)	ULS	none	PLS ($P = 0.5$)	PLS ($P = 1.0$)	PLS (asc)	ULS
100	+	+	−	+	−	+	+	−	+	−
150	+	+	−	+	−	+	+	−	+	−
200	+	+	−	+	−	+	+	−	+	−
300	+	+	+	+	−	+	+	#	+	−
400	+	+	+	+	−	+	+	+	+	−
500	+	+	+	+	−	+	+	+	+	−
750	+	+	+	+	−	+	#	+	+	−
1000	+	+	+	+	−	+	+	+	+	−
F-W	+	+	−	+	−	+	#	−	+	−

4 Conclusion

In this paper a local search method based on a local utopia point was proposed. This method was compared to a Pareto-based local search on several instances of the multiobjective Travelling Salesman Problem. The proposed method was found to outperform the Pareto-based approach when given equal computational time.

For the Pareto-based method five different strategies of invoking the local search procedure were tested. Out of these strategies the best one performs the local search with probability $P = 1.0$ at the beginning of the optimization and the reduces this probability linearly to $P = 0.0$.

Further work may include using different method of establishing the reference point for the ULS method as well as a research on various strategies of balancing computational resources between the evolutionary optimization and the ULS.

References

1. Chiang, C.W., Lee, W.P., Heh, J.S.: A 2-opt based differential evolution for global optimization. Applied Soft Computing 10(4), 1200–1207 (2010)
2. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6, 182–197 (2002)
3. Derrac, J., Garca, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Comput. 1(1), 3–18 (2011)
4. Dubois-Lacoste, J., et al.: A hybrid TP+PLS algorithm for bi-objective flow-shop scheduling problems. Comput. Oper. Res. 38(8), 1219–1236 (2011)
5. Goldberg, D.E., Voessner, S.: Optimizing global-local search hybrids. In: Banzhaf, W., Daida, J., Eiben, A.E., Garzon, M.H., Honavar, V., Jakiela, M., Smith, R.E. (eds.) Proceedings of the Genetic and Evolutionary Computation Conference. vol. 1, pp. 220–228. Morgan Kaufmann, Orlando, Florida, USA (1999)
6. Ishibuchi, H.: Memetic algorithms for evolutionary multiobjective combinatorial optimization. In: Computers and Industrial Engineering (CIE), 2010 40th International Conference on. pp. 1–2 (2010)
7. Li, M., Zheng, J.: Spread assessment for evolutionary multi-objective optimization. In: Proceedings of the 5th International Conference on Evolutionary Multi-Criterion Optimization. pp. 216–230. EMO '09, Springer-Verlag, Berlin (2009)
8. Lust, T., Teghem, J.: Two-phase pareto local search for the biobjective traveling salesman problem. Journal of Heuristics 16(3), 475–510 (2010)
9. Ott, L., Longnecker, M.: An Introduction to Statistical Methods and Data Analysis. Brooks/Cole Cengage Learning (2010)
10. Sinha, A., Goldberg, D.E.: Verification and extension of the theory of global-local hybrids. In: Proceedings of GECCO 2001
11. Tao, G., Michalewicz, Z.: Inver-over operator for the TSP. In: Eiben, A.E., et al. (eds.) Parallel Problem Solving from Nature - PPSN V, Lecture Notes in Computer Science, vol. 1498, pp. 803–812. Springer Berlin Heidelberg (1998)
12. Thibaut Lust: Multiobjective TSP. `https://sites.google.com/site/thibautlust/research/multiobjective-tsp` (2015), online: accessed 2015.01.29
13. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., da Fonseca, V.G.: Performance assessment of multiobjective optimizers: An analysis and review. IEEE Transactions on Evolutionary Computation 7, 117–132 (2002)