Supplementary Materials: Some Multi-Objective Local Search Algorithms Are Better than Others

Anonymous submission

Introduction

This supplementary document provides additional material for our main submission titled *"Some Multi-Objective Local Search Algorithms Are Better than Others"* to the 40th annual AAAI conference on artificial intelligence.

The rest of the document is structured as follows. The first section introduces the multi-objective combinatorial optimisation problems considered in this paper. The second and third sections give the experimental settings and additional experimental results for larger problem sizes. The fourth section introduces the procedure of investigating the distribution of the number of good neighbours. Finally, the fifth section gives the additional proofs regarding the two toy examples in the main paper.

Multi-Objective Combinatorial Problems

We consider four commonly used MOCOPs, the multi-objective 0/1 knapsack (KP) (Teghem 1994), travelling salesman problem (TSP) (Ribeiro et al. 2002), quadratic assignment problem (QAP) (Knowles and Corne 2003) and NK-landscapes (Aguirre and Tanaka 2004). These problems exhibit diverse characteristics, including pseudo-Boolean (knapsack and NK-landscape) and permutation (TSP and QAP) formulations; smooth (knapsack and TSP) to rugged (QAP and NK-landscape with a large K) fitness landscape; order-based (TSP) versus position-based (QAP) permutations; and constrained (knapsack) versus unconstrained (TSP, QAP and NK-landscape) settings. Each problem was instantiated in three sizes (100, 200 and 500 decision variables).

Multi-Objective 0-1 Knapsack Problem (Knapsack). The multi-objective 0-1 knapsack problem (Teghem 1994) is a widely studied MOCOP. Given a set of D items $\mathbf{x} = (x_1, x_2, \ldots, x_D) \in \{0,1\}^D$, the m-objective problem is defined as the following.

$$\begin{aligned} \max\ f_j(x) = \sum_{i=1}^{D} v_{ji} x_i, \quad j = 1, \ldots, m \\ \text{s.t.} \sum_{i=1}^{D} w_i x_i \le c \end{aligned} \tag{1}$$

Here, $v_{ji} \ge 0$ is the value of the item i in objective j, w_i is the item's weight, and $c = \frac{1}{2}\sum w_i$ is the capacity. Following (Li et al. 2024), both v_{ji} and w_i are sampled uniformly from $\{10, 11, \ldots, 100\}$.

Multi-Objective Travelling Salesman Problem (TSP). The multi-objective TSP extends the classical TSP, with multiple costs between each pair of cities (Ribeiro et al. 2002), and aims to find the route minimising multiple travelling costs for visiting all the cities exactly once, returning to the start. Formally, given a network $L = (V, C)$, where $V = \{v_1, \ldots, v_D\}$ is a set of D nodes and $C = \{C_j : j \in \{1, \ldots, m\}\}$ is a set of m cost matrices between nodes $(C_j : V \times V)$, the problem is to find the Pareto optimal set of Hamiltonian cycles that minimise each of the m cost objectives.

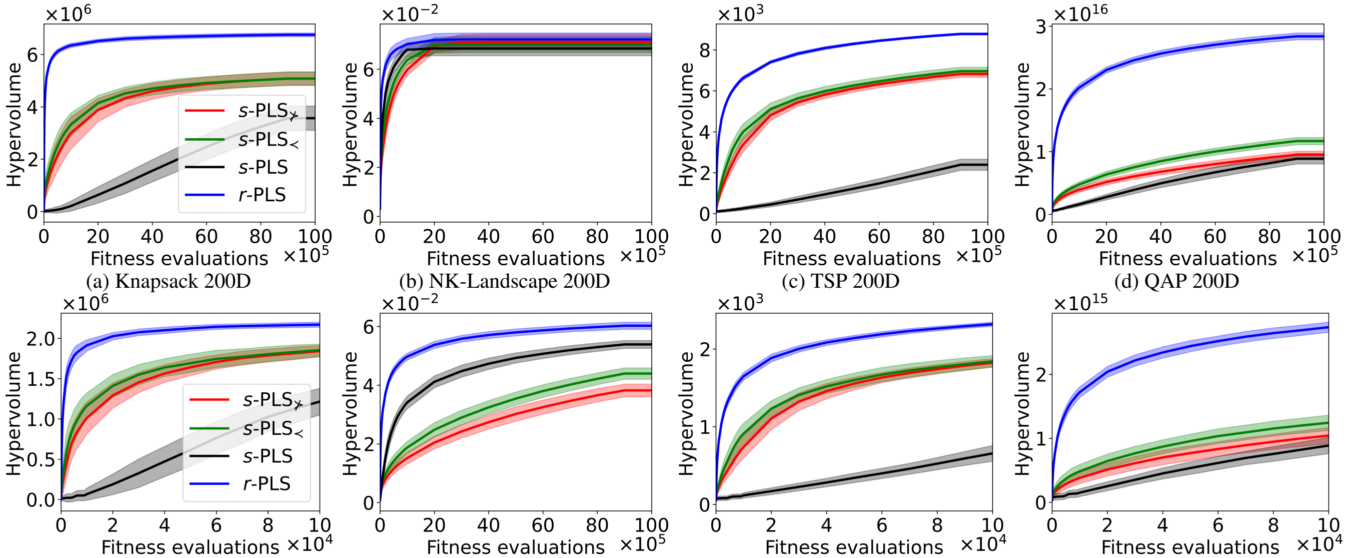
Multi-Objective Quadratic Assignment Problem (QAP). The multi-objective QAP (Knowles and Corne 2003) models facility-location assignments with multiple flow types. Given m cost matrices $[C_{1ij}], \ldots, [C_{mij}]$ and a distance matrix $[L_{uv}]$, a solution is a permutation $x = (x_1, ..., x_D)$ where x_i denotes the location of facility i. The problem is defined as the following.

$$\min\ f_k(x) = \sum_{i=1}^{D} \sum_{j=1}^{D} C_{kij} L_{x_i x_j}, \quad k = 1, \ldots, m \tag{2}$$

Multi-Objective NK-Landscape. NK-landscapes (Aguirre and Tanaka 2004) are widely used due to their tuneable ruggedness (Verel et al. 2013). Given D bits and epistasis degree K, each objective f_j is defined as:

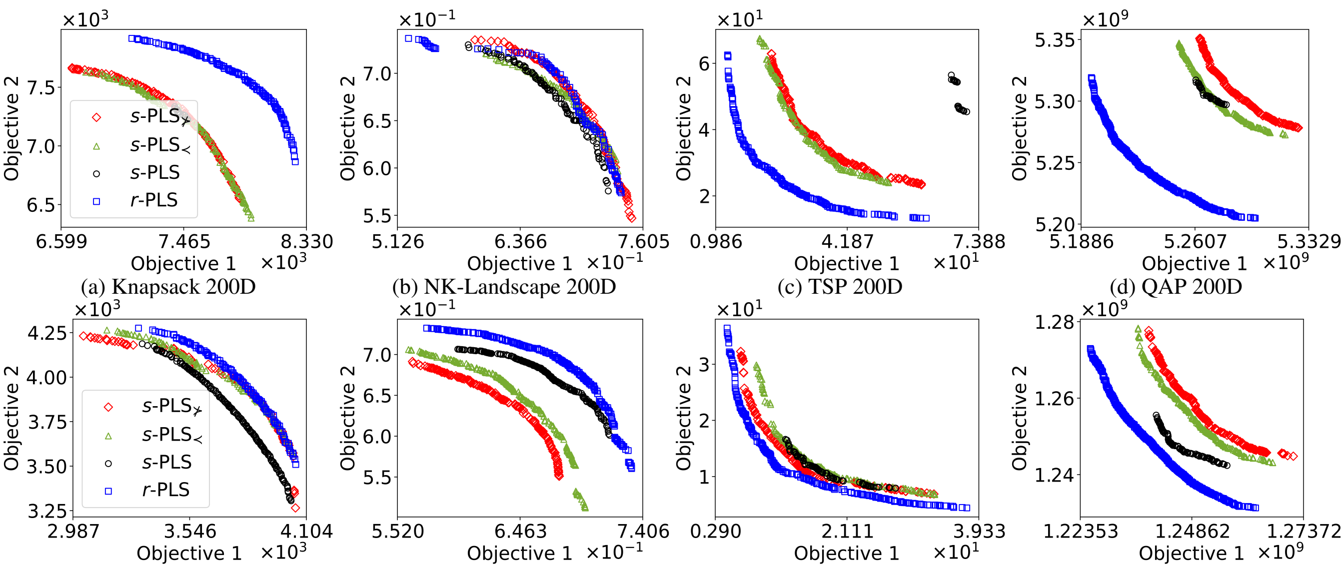
$$\max f_j(x) = \frac{1}{D} \sum_{i=1}^{D} c_{ij}(x_i, x_{k_{ij1}}, ..., x_{k_{ijK}}), \quad j = 1, \ldots, m. \tag{3}$$

Each c_{ij} is randomly sampled over all 2^{K+1} possible combinations of its loci. Following (Aguirre and Tanaka 2007; Daolio et al. 2015), loci are drawn independently for each i and j, resembling a random epistasis pattern.



(a) Knapsack 200D (b) NK-Landscape 200D (c) TSP 200D (d) QAP 200D
(e) Knapsack 500D (f) NK-Landscape 500D (g) TSP 500D (h) QAP 500D

Figure 1: The hypervolume trajectory (the higher the better) of the considered s-PLS, s-PLS$_{\nsucc}$, s-PLS$_{\prec}$ and r-PLS across 30 runs on the four MOCOPs with 200 variables (the upper panel) and with 500 variables (the lower panel). The bolded line and shaded area represent the mean and standard deviation of the hypervolume, respectively.



(a) Knapsack 200D (b) NK-Landscape 200D (c) TSP 200D (d) QAP 200D
(e) Knapsack 500D (f) NK-Landscape 500D (g) TSP 500D (h) QAP 500D

Figure 2: Non-dominated solutions obtained by the considered s-PLS, s-PLS$_{\nsucc}$, s-PLS$_{\prec}$ and r-PLS in a typical run on the four MOCOPs with 200 variables (the upper panel) and with 500 variables (the lower panel), where the Knapsack and NK-Landscape are maximisation problems, and the TSP and QAP are minimisation problems.

Experimental Settings

In our experiments, for the four MOCOPs, we test the problem sizes of 100, 200 and 500 respectively, with a budget of 1,000,000 fitness evaluations. For illustration purpose, the results of problem size 100 presented in the main paper are 100,000 fitness evaluations only, as the s-PLS stops early with this problem size.

It is worth noting that SEMO and PLS use different stopping conditions: SEMO exhausts the budget, while PLS may terminate early upon archive saturation. To ensure fairness, all algorithms start from the same initial solution, and SEMO is forcibly stopped when PLS terminates. Each instance is run independently 30 times.

Neighbourhood operators are matched to the encoding scheme of the problem: NK-landscape uses 1-bit (binary), TSP uses 2-opt (order-based), and QAP uses 2-swap (position-based). Knapsack requires a 2-bit flip neighbourhood, as 1-bit flips often fail to yield feasible or improved solutions near the constraint boundary.

We evaluate performance using the Hypervolume (HV) indicator (Zitzler and Thiele 1999), using 100,000 random samples of the problem to estimate the reference point (Li et al. 2024). This is necessary because using the reference point determined by the non-dominated combined set of all generated solutions may easily cause the HV to be zero (Li, Chen, and Yao 2022; Li et al. 2024).

Additional Results

Here, we present the additional experimental results on the four MOCOPs with the problem sizes of 200 and 500. Figure 1 shows the average hypervolume (bolded line) and standard deviation (shaded area) of the two *s*-PLS algorithms, as well as the basic *s*-PLS and *r*-PLS across 30 independent runs on the four problems with 100 decision variables. As can be seen, *r*-PLS obtains a better HV value than all *s*-PLS variants throughout the search on all the problems, and obtained a better result in most of the problems (except for NK-landscape with 200 variables, where all algorithms perform similarly). Figure 2 gives the non-dominated solutions obtained by the four algorithms in a typical run on the four problems when the search ends.

Distribution Fit At Each Collection Time

To find a model that can describe the number of good neighbours, at each collection time, we consider several common discrete distributions as candidates, namely uniform, Poisson, geometric, binomial and Zipf. We also consider the Categorical distribution as a last resort, which can fit any data. Parameters were estimated via maximum likelihood: Poisson's λ, geometric's p and binomial's p by sample mean; Zipf's exponent s via numerical minimisation of negative log-likelihood over $[1.01, 10]$; and categorical frequencies from empirical counts.

We assess absolute fit quality by a χ^2 goodness-of-fit test at a $\alpha = 5\%$ significance level. To compare the relative quality of fit, we rank all six fits by Akaike's Information Criterion (AIC) (Akaike 1974), computing degrees of freedom as $n_{\text{bin}} - 1 - k$ where n_{bin} is the number of solutions that have at least one good neighbour and k is the number of free parameters of the distribution. Here, if the categorical distribution has the best fit, it indicates that there may not be a pattern within the data.

Figure 3 plots the goodness-of-fit of the five discrete distributions with respect to the numbers of good neighbours of solutions in the archive during the search process of *s*-PLS and *r*-PLS on the four MOCOPs. Each horizontal band corresponds to a candidate distribution, and at each sampled evaluation, two coloured ticks, black and blue, respectively indicate a good fit for *s*-PLS and *r*-PLS at $\alpha = 0.05$, namely, the χ^2 test does not reject the distribution. As can be seen in the figure, across nearly all problem instances and time-scenarios, the geometric distribution achieved the lowest or the second-lowest AIC by margins often exceeding ten to the next(i.e. "strong" over second-best). Moreover, the geometric distribution more frequently passed the χ^2 test significantly than others, indicating that the number of good neighbours is more consistent with a geometric distribution. In contrast, the Poisson distribution only fits in the later stages of the search, where the geometric distribution also fits, as well. The binomial usually has the worst AIC on permutation problems (e.g., TSP), and a second worst on binary problems (e.g., knapsack). The uniform distribution is only fitted and more fitted than others for PLS at the first to four $|\mathcal{N}|$ searches, as it is basically telling the quality of neighbouring solutions or the initial solution, but is not fitted in other cases. Zipf's distribution cannot capture the large spike at zero (indicating no good neighbours at the later stage of the search); if multiple distributions fit at the same time, Zipf's AIC is often the worse.

Other Proof Details

Proposition 1 (Half good and half bad solutions)**.** *Let an archive of n solutions contains exactly $n/2$ solutions have no good neighbours ($E = 0$), and another $n/2$ solutions with all neighbours are promising ($E = |\mathcal{N}|$). Then, given $|\mathcal{N}| \geq 2$,* r*-PLS has a smaller expected runtime than* s*-PLS in finding the next new promising solution.*

Proof. For *r*-PLS, with probability $\frac{1}{2}$, the selected solution is full of good neighbours, and a uniformly drawn neighbour is automatically promising. Thus,

$$p_{\text{succ}} = \tfrac{1}{2}, \qquad \mathbb{E}[T_{r\text{-PLS}}] = \frac{1}{p_{\text{succ}}} = \frac{1}{1/2} = 2. \tag{4}$$

As for *s*-PLS, it wastes $|\mathcal{N}|$ evaluations for every "bad" solution encountered at the beginning until it reaches a "good" solution. Hence, we need to first estimate the timing that *s*-PLS finds a "good" solution. *s*-PLS typically scan through each archive solution one-by-one as it remember the explored solution. Therefore, the above estimation resembles the process of finding the first index of a good neighbour during scan. By lemma 1, the first solution with all good neighbours appears at index

$$\mathbb{E}[J] = \frac{n+1}{(n/2)+1} = \frac{2(n+1)}{n+2}.$$

Since each preceding "bad" solution consumes $|\mathcal{N}|$ evaluations and the first solution with all good neighbours consumes exactly 1, the expected runtime of *s*-PLS is the following:

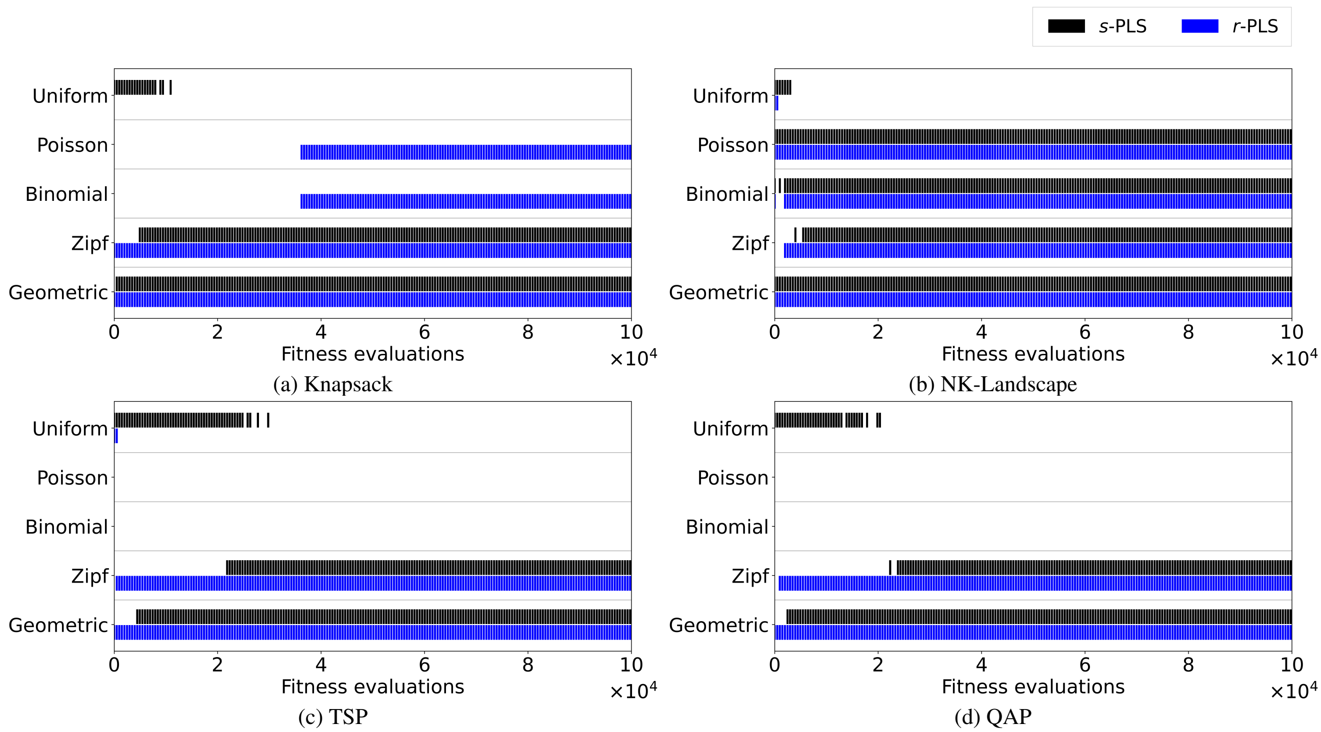
$$\begin{aligned} \mathbb{E}[T_{s\text{-PLS}}] &= \big(\mathbb{E}[J] - 1\big)|\mathcal{N}| + 1 \\ &= \left(\frac{2(n+1)}{n+2} - 1\right)|\mathcal{N}| + 1 = \frac{n|\mathcal{N}|}{n+2} + 1. \end{aligned} \tag{5}$$

Since $|\mathcal{N}| \geq 2$, it follows that $\mathbb{E}[T_{s\text{-PLS}}] > \mathbb{E}[T_{r\text{-PLS}}]$. $\square$

Proposition 2 (Half neighbours are good for all solutions)**.** *Let an archive of n solutions and each solution has exactly half good neighbours ($E = |\mathcal{N}|/2$). Then, given $|\mathcal{N}| \geq 2$,* s*-PLS has a smaller expected runtime than* r*-PLS in finding the next new promising solution.*

Proof. For *r*-PLS, with probability $\frac{1}{2}$, it picks a good neighbour from a randomly selected solution. Thus,

$$p_{\text{succ}} = \tfrac{1}{2}, \qquad \mathbb{E}[T_{r\text{-PLS}}] = \frac{1}{p_{\text{succ}}} = \frac{1}{1/2} = 2. \tag{6}$$



(a) Knapsack (b) NK-Landscape (c) TSP (d) QAP

Figure 3: Goodness-of-fit of the distributions with respect to the number of good neighbours of solutions in the archive during the search process of *s*-PLS (black) and *r*-PLS (blue) on the (a) Knapsack (100 items), (b) NK-Landscape (N=100, K=10), (c) TSP (100 cities) and (d) QAP (100 factories). A coloured tick in a row indicates that the corresponding algorithm's data at that point was not rejected under the model.

As for *s*-PLS, we only need to consider the neighbourhood of one solution since all solutions have half of their neighbours being promising. By lemma 1, the first good neighbour appears at the index of the neighbourhood at $E[J] = \frac{|\mathcal{N}|+1}{(|\mathcal{N}|/2)+1}$, resembles the runtime of *s*-PLS:

$$\mathbb{E}[T_{s\text{-PLS}}] = E[J] = \frac{|\mathcal{N}|+1}{(|\mathcal{N}|/2)+1} < 2 = \mathbb{E}[T_{r\text{-PLS}}].$$

$\square$

References

Aguirre, H.; and Tanaka, K. 2007. Working principles, behavior, and performance of MOEAs on MNK-landscapes. *European Journal of Operational Research*, 181(3): 1670–1690.

Aguirre, H. E.; and Tanaka, K. 2004. Effects of elitism and population climbing on multiobjective MNK-landscapes. In *IEEE Congress on Evolutionary Computation*, volume 1, 449–456.

Akaike, H. 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6): 716–723.

Daolio, F.; Liefooghe, A.; Verel, S.; Aguirre, H.; and Tanaka, K. 2015. Global vs local search on multi-objective NK-landscapes: contrasting the impact of problem features. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, 369–376.

Knowles, J.; and Corne, D. 2003. Instance generators and test suites for the multiobjective quadratic assignment problem. In *International Conference on Evolutionary Multi-Criterion Optimization*, 295–310.

Li, M.; Chen, T.; and Yao, X. 2022. How to evaluate solutions in Pareto-based search-based software engineering? A critical review and methodological guidance. *IEEE Transactions on Software Engineering*, 48(5): 1771–1799.

Li, M.; Han, X.; Chu, X.; and Liang, Z. 2024. Empirical Comparison between MOEAs and Local Search on Multi-Objective Combinatorial Optimisation Problems. In *Genetic and Evolutionary Computation Conference*, 547–556. Melbourne VIC Australia: ACM. ISBN 979-8-4007-0494-9.

Ribeiro, C. C.; Hansen, P.; Borges, P. C.; and Hansen, M. P. 2002. A study of global convexity for a multiple objective travelling salesman problem. *Essays and Surveys in Metaheuristics*, 129–150.

Teghem, J. 1994. Multi-objective combinatorial optimization problems: A survey. *Journal of Multi-Criteria Decision Analysis*.

Verel, S.; Liefooghe, A.; Jourdan, L.; and Dhaenens, C. 2013. On the structure of multiobjective combinatorial search space: MNK-landscapes with correlated objectives. *European Journal of Operational Research*, 227(2): 331–342.

Zitzler, E.; and Thiele, L. 1999. Multiobjective evolutionary algorithms: A comparative case study and the strength

Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4): 257–271.