A Comparative Assessment of Memetic, Evolutionary, and Constructive Algorithms for the Multiobjective d-MST Problem

Joshua D. Knowles and David W. Corne
Department of Computer Science, University of Reading, UK
J.D.Knowles@reading.ac.uk, D.W.Corne@reading.ac.uk
FAX: +44(0) 118 975 1994, TEL: +44 (0) 118 931 8983
http://www.rdg.ac.uk/~ssr97jdk

Abstract

Finding a minimum-weight spanning tree (MST) in a graph is a classic problem in operational research with important applications in network design. In this paper, we consider the degree-constrained multiobjective MST problem, which is NP-hard. On fifteen benchmark instances, we compare the performance of three different algorithms: the Pareto archived evolution strategy (PAES); a new multiobjective evolutionary algorithm, AESSEA; and the memetic PAES algorithm, M-PAES, all employing the same initialization procedure, encoding and operators. We find M-PAES performs well on the whole range of problem types, generally outperforming the pure evolutionary and local search algorithms.

1 Introduction

For many years, minimum spanning tree (MST) problems have been of great interest to the operational research community. More recently, the multiobjective minimum spanning tree (mc-MST) problem, in which there are multiple weights defined on each edge, has become subject to considerable attention. Several papers on this subject [1, 3, 12] have proposed both approximate polynomial algorithms, and worst-case exponential time exact methods, for tackling the problem.

Zhou and Gen [13] have also proposed a multiobjective GA approach to the mc-MST. However, the instances tackled were very simple and could, in fact, be much better solved using good exact methods [12] (smaller instances), or heuristic approximation methods [1, 3] (larger instances). In a recently published paper by us [8], we extended on the work of Zhou and Gen by developing a new multiobjective EA for the mc-MST problem, called AESSEA, and compared a Prüfer encoding [10] (as used by Zhou and Gen) with a direct encoding and specialized operators that we adapted from those of Raidl [11] for the degree-constrained MST (d-MST). Our results demonstrated clear superiority of the direct encoding method on some simple problem instances without constraints. However, the simple instances could also be tackled more efficiently using a polynomial-time, constructive heuristic approach.

In this paper, we tackle a set of more challenging benchmark instances developed by us, with constraints and other difficult problem features. Using the same initialization, encoding and operators developed previously, we compare the performance of PAES [5] - a local search strategy, AESSEA [8], and our multiobjective MA, M-PAES [6]. All of the problem instances, generators, and results sets are available for others to use by e-mail contact with the first author.

2 Multiobjective Degree-Constrained Spanning Tree Problem

A spanning tree of an undirected, connected graph, $G = (V, E)$, is a subgraph $T = (V, E_T), E_T \subset E$ that contains all vertices in V and connects them with exactly $|V| - 1$ edges, so that there are no cycles. If G is complete, then the set S of spanning trees T of G has $|S| = |V|^{|V|-2}$ members. If each edge $(i, j) \in E$ has $K > 1$ associated non-negative real numbers, representing K attributes defined on it and denoted with $\mathbf{w}_{i,j} = (w^1_{i,j}, w^2_{i,j}, \ldots, w^K_{i,j})$, then the mc-MST problem may be defined as:

$$\begin{aligned} \text{“minimize”}\ \mathbf{W} &= (W^1, W^2, \ldots, W^K) \\ \text{with } W^k &= \sum_{(i,j)\in E_T} w^k_{i,j},\ k \in 1..K \end{aligned} \tag{1}$$

where the term 'minimize' is in quotation marks to indicate that it may not be possible to find a single solution that is minimal on all the components of $\mathbf{W}$. Instead, one is required to find a set of spanning trees $S^* \subset S$, called the Pareto optimal set, with the property that:

$$\forall T^* \in S^* \bullet \ \nexists T \in S \bullet T \prec T^* \tag{2}$$

where $T \prec T^* \iff \forall k \in 1..K \bullet W^k \leq W^{k^*} \wedge \exists\, k \in 1..K \bullet W^k < W^{k^*}$. The expression $T \prec T^*$ is read as T *dominates* T^*, and solutions in the Pareto optimal set are also known as efficient or admissible solutions.

If there is, in addition, a constraint d on the maximum vertex degree in the spanning tree, then the problem is called the multiobjective degree-constrained minimum spanning tree (mc$d$-MST) problem.

3 Algorithms

Details of the PAES algorithm can be found in [5], details of the M-PAES algorithm can be found in [6], and AESSEA is described in [8]. A baseline polynomial algorithm mc$d$-Prim, used for benchmarking the EA results, is described briefly below, as are the initialization procedure, direct encoding and operators used in the EAs.

Algorithm mc$d$-Prim

Prim's algorithm [9] is a well-known polynomial time constructive algorithm for solving the (single-objective, unconstrained) MST problem. It can be adapted to the d-MST by changing it so that at each step, in its construction of a tree, it checks for a degree-constraint violation before adding in the next edge. Of course, since d-MST is NP-hard, the resulting algorithm does not guarantee optimality. For tackling the mc$d$-MST, a multiobjective version of this constructive algorithm is also easily devised. By simply replacing the vector of edge weights in the graph by a weighted sum scalarization of them, optimization can be carried out in one 'direction' of the objective space. In mc$d$-Prim, this procedure is iterated for many different weightings of the objectives, giving a whole range of solutions approximating the Pareto front. We use mc$d$-Prim (using 1001 different weightings) here as a baseline algorithm. In the experiments reported below, we always run mc$d$-Prim five times, each time with a different start vertex.

Randomized primal method for initialization

The randomized primal method (RPM) was put forward in [4] (where it is described in detail) as an encoding for solving the d-MST problem using any meta-heuristic search method. It is a decoder type of representation, that is, the chromosome encodes for choices that are made when a constructive algorithm builds a tree. The problem with this type of encoding is that it does not exhibit good locality, and it has super-linear growth in complexity for linear increase in the graph size $|V|$. However, it is good for initialization where it is used only a small number of times. The RPM for multiobjective problems is, like mc$d$-Prim, based on a weighted scalarization of the objectives. Note also that RPM reduces to mc$d$-Prim if the chromosome controlling it is set to all zeros. We use a slightly more random initialization than this, though with a strong bias towards lower allele values, for initialization to provide slightly more initial diversity. We note that on easy problems, using the mc$d$-Prim (all-zeros) initialization would make the EAs converge slightly faster.

Direct encoding and operators

The direct encoding and operators used in AESSEA and M-PAES are multiobjective versions of those put forward by Raidl in [11] for the d-MST problem. We describe how these operators were adapted for the mc$d$-MST problem in detail in [8]. In summary, the operators are adapted so that they bias the choice of edges towards those that are the minima on some weighted-sum single objective evaluation of the multiobjective tree weight. The weights in the weighted-sum are also encoded for by the chromosome and are subject to mutation and crossover. The method ensures that good solutions are found across the whole Pareto front.

4 Experimental method

Generated instances

Four types of problems were considered: Correlated, Anti-Correlated, M-Correlated, and Concave. Each presents a particular difficulty to an optimizer. These problems are described in detail in [7].

Three graphs for each of the problem types were generated: one each at sizes of 10, 25, and 50 vertices, giving 12 graphs in all, from which 15 instances are created by setting degree constraints of 3 on all of the instances, and an additional, lighter degree constraint of 5 on the three M-correlated instances.

The correlations for the Anti-Correlated graphs, 10vAC, 25vAC, and 50vAC were set at -0.7. For the Correlated graphs, 10vC, 25vC, and 50vC, the correlation was set at 0.7. For the M-Correlated graphs the

Parameter	PAES	AESSEA	M-PAES
population size, $\|P\|$	1	200	50
nondominated solutions archive size, *arcsize*	200	200	$G = 200, H = 200$
initialization method	RPM	RPM	RPM
mutation type	mutate_RPM	edge-mutation	edge-mutation
crossover type	—	edge-crossover	edge-crossover
total number of function evaluations, *num_evals*	20k/50k/50k	20k/50k/50k	20k/50k/50k
# of grid squares used for 'crowding' strategy [5]	1024	1024	1024

Table 1: Parameter settings for the three algorithms; in addition $l\_opt = 50$, $l\_fails = 20$, $cr\_trials = 10$ for M-PAES (see [6] for the meaning of these parameters). The three figures for number of evaluations relate to the three different problem sizes, 10, 25, and 50 vertices, respectively

correlation was also set at 0.7, and the other parameters were $f = 1, 2, 5$, $ld = 6, 6, 7$, $ud = 8, 10, 9$, for the 10vM-C, 25vM-C, and 50vM-C, respectively. There is no correlation between the edge weight components in the concave graphs, and the other parameters used for generating these graphs were $\zeta = 0.1, 0.05, 0.03$ and $\eta = 0.25, 0.2, 0.125$ for the 10vConc, 25vConc and 50vConc graphs respectively.

Algorithm parameters and experiments

The parameters used for each of PAES, AESSEA, and M-PAES are given in Table 1. Each of the algorithms was given 30 independent runs on each of the 15 instances and the nondominated archive returned by each algorithm from each run was stored for statistical analysis, and comparison with the other non-EA approaches. For each problem instance, mc$d$-Prim was also run 5 times with a different start vertex, and the combined nondominated solution set was stored. On the ten-vertex instances we also use an enumerative procedure to give us the entire true Pareto front for comparison.

Statistical Analysis of Data

We use two different methods to analyse the statistical performance of the three algorithms over multiple runs on the 15 different problems. Both methods rely on a technique developed by Fonseca and Fleming [2], and later implemented and extended by us [5]. The technique relies on the notion that the nondominated points from any approximation to a Pareto set define a surface (called the attainment surface), that divides up the objective space into a region that is dominated by the discovered nondominated points, and a region that is not dominated by them. Over multiple runs, an approximate algorithm will generate multiple different attainment surfaces. By sampling the distance of these surfaces from an origin at many different angles, one can obtain statistical information about the expected position of the attainment surface along each different angled direction (or sampling line). For example, one can calculate the median attainment surface, or the quartile attainment surfaces of an algorithm. One can also compare directly the *whole* distribution of positions of the attainment surfaces obtained from multiple runs of two or more different algorithms. The first method of analysis that we use here does just this. The 30 attainment surfaces for each of the three algorithms are sampled using 500 different equally distributed angled sampling lines, and on each line we obtain the positions of the 90 intersections of the attainment surfaces. Using a Mann-Witney rank sum statistical test at the 95% confidence level we can determine if one distribution of attainment surfaces is significantly superior to another's. By doing this on each of the 500 sample lines we can calculate the percentage of the sample lines (and therefore the percentage of the Pareto front) on which one distribution of attainment surfaces is better than another, and the percentage where there is no statistical difference. For three or more optimizers we present these results in the form of two statistics: the 'unbeaten' statistic of a distribution of attainment surfaces describes the percentage of the sampling lines on which the distribution was unbeaten by any of the two or more other distributions of attainment surfaces, against which it is being compared; and the 'beats all' statistic gives the percentage of the sampling lines on which the distribution beats all of the other distributions, each at a statistically significant level[1]. The disadvantages of this approach are that the results are purely comparative (that is they are not absolute results that others can compare against), and also it is not clear by how much one algorithm's attainment surfaces dominate

[1] Although the statistical significance holds for each pair of distributions in turn, the level of significance is reduced over multiple tests

another. Because of these objections we use a second method that handles both of these problems.

The second method is not comparative - it gives a statistic which relates to just a single algorithm's distribution of attainment surfaces. This statistic can then be compared between algorithms. So for a single algorithm's sets of nondominated points, the method first calculates the attainment surfaces as above, and then calculates the median and quartile surfaces. Now, to convert these surfaces into a simple figure of merit, the size of the dominated region of the surfaces can be calculated. For a two-objective minimization problem (as we have here) this is simply the area above and to the right of the attainment surface up to some bounding rectangle. How the bounding rectangle is best set is open to debate. Here we calculate the weight of the worst feasible solution (heaviest spanning tree) for each of the objectives in turn, giving us a point (z^1, z^2) that is used as the upper right corner of our bounding rectangle. (The points were calculated using Prim's algorithm set to maximize). In our results we report the absolute (unnormalized) size of the dominated region of the median surface, and (to get an idea of the variation over different independent runs), the interquartile dominated region size. The disadvantage of this approach is that concave regions of the Pareto front are under-represented in the statistics. This is seen very clearly in the results section where there is a clear difference in the results reported by method 1 and by method 2 for the concave graph instances.

5 Results

The results of our first statistical analysis method are given in Table 2. The best results are shown in bold. It is clear from the table that M-PAES and AESSEA both using a direct coding and employing crossover do favourably compared to PAES using only the RPM decoder encoding, and this superiority is emphasized further as instance size increases. M-PAES and AESSEA using the same encoding are well-matched on most instances but M-PAES is clearly best overall on the set of instances considered. It is particularly strong on the largest of the M-correlated graphs, and the large concave graph problem.

The results of the second statistical analysis method are given in Table 3. The results paint a similar picture overall, although there are some interesting points to notice as well. First, mc*d*-Prim performs very well compared to the evolutionary algorithms on the correlated and anti-correlated instances, and is considerably faster (but see Figure 1 for further help visualizing the Pareto fronts discovered). However, on the M-correlated graphs it clearly struggles. This is as expected because on these instances the degree constraint has a real effect on the difficulty of finding good solutions. We can see that the EAs are managing to cope with this; observe the size of the region discovered by them compared to the enumeration method on the 10 vertex M-correlated instance for both d=3 and d=5. This is further shown in a plot given in Figure 2. Finally, on the (larger) concave instances, it appears that mc*d*-Prim does better than the EAs but in fact is unable to find any solutions in the concave region of the Pareto front. Its larger dominated region is due to it finding the very edge of the Pareto Front, which the EAs do not achieve on every run. A plot show-

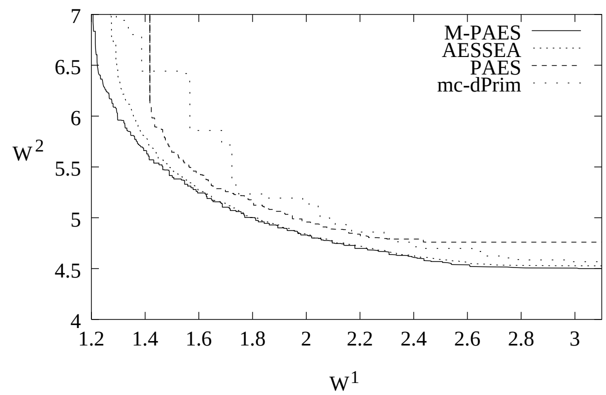

Figure 1: Median attainment surfaces on the 50 vertex correlated problem with a degree constraint of 3

Instance	PAES	AESSEA	M-PAES
10vAC	32.7 (0)	97.6 (0)	**100 (2.4)**
25vAC	0 (0)	13.3 (7.3)	**92.7 (86.7)**
50vAC	0 (0)	17.4 (15.7)	**84.3 (82.6)**
10vC	99.5 (0)	**100 (0)**	**100 (0)**
25vC	45.1 (0)	**99.6 (10.8)**	89.2 (0)
50vC	0 (0)	3.8 (0)	**100 (96.2)**
10vM-C-d3	95.6 (0)	**100 (0)**	**100 (0)**
25vM-C-d3	0 (0)	**99.3 (14.5)**	85.5 (0.7)
50vM-C-d3	0 (0)	17.3 (2.5)	**97.5 (82.7)**
10vM-C-d5	**100 (0)**	**100 (0)**	**100 (0)**
25vM-C-d5	0 (0)	**100 (3.8)**	96.2 (0)
50vM-C-d5	0 (0)	19.2 (0)	**100 (81.8)**
10vConc	52.1 (0)	**100 (0)**	**100 (0)**
25vConc	9.2 (0.6)	68.5 (22.7)	**74.6 (30.1)**
50vConc	0 (0)	36 (3.7)	**96.3 (64)**

Table 2: Unbeaten and (beats all) statistics for the three EAs on the full set of instances

Instance	Total size		Median (Interquartile) size		
	Enum	mc$d$-Prim	PAES	AESSEA	M-PAES
10vAC	21.3837	20.4108	21.2963 (0.2014)	21.3565 (0.0032)	**21.357 (0.0044)**
25vAC		246.256	239.248 (32.121)	245.175 (1.428)	**245.5 (1.893)**
50vAC		1240.27	1040.02 (440.492)	1224.07 (7.6)	**1225.67 (13.85)**
10vC	36.0955	35.3919	35.7349 (0.0323)	**35.7672 (0)**	**35.7672 (0)**
25vC		363.233	362.409 (1.492)	363.329 (0.109)	**363.386 (0.149)**
50vC		1868.59	1853.01 (24)	1869.3 (3.52)	**1873.29 (4.23)**
10vM-C-$d3$	26.8362	23.6895	**26.8399 (0)**	26.8342 (0)	26.8342 (0)
25vM-C-$d3$		281.923	338.026 (7.135)	343.892 (0.55)	**343.988 (0.005)**
50vM-C-$d3$		1302.33	1454.86 (40.14)	**1493.89 (3.17)**	1493.48 (8.9)
10vM-C-$d5$	40.0389	35.3428	**40.0365** (0)	**40.0365 (0)**	**40.0365 (0)**
25vM-C-$d5$		345.62	383.684 (2.801)	**384.511 (0)**	384.439 (0)
50vM-C-$d5$		1496.8	1598.07 (26.44)	1615.49 (4.56)	**1616.33 (7.79)**
10vConc	37.3255	37.0848	37.2362 (0.0061)	**37.2367 (0)**	**37.2367 (0)**
25vConc		334.694	332.901 (3.81)	**334.644 (0.354)**	334.491 (0.852)
50vConc		2122	2109.76 (19.39)	**2118.39 (1.67)**	2114.7 (4.29)

Table 3: Size of the median and interquartile dominated regions for the different evolutionary algorithms, and the total combined size of the dominated region found using five runs of mc$d$-Prim. For the ten-vertex instances the true size of the dominated region is represented by the results of the Enumeration algorithm

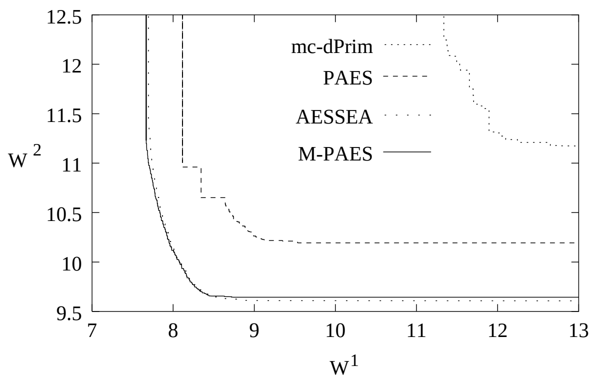

Figure 2: Median attainment surfaces on the 50 vertex M-correlated problem with a degree constraint of 3

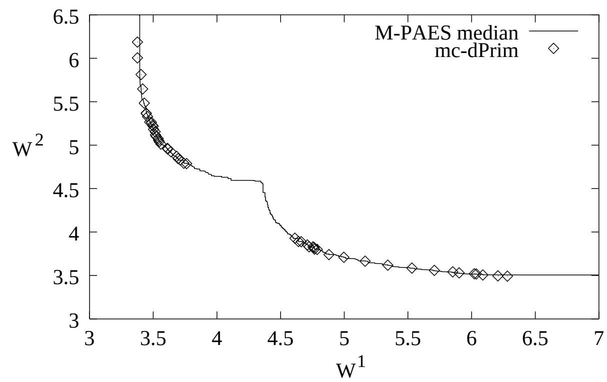

Figure 3: Nondominated points found from 5 runs of mc$d$-Prim, and the median attainment surface achieved by M-PAES. Note how M-PAES finds points in the concave region of the Pareto front

ing the median attainment surface for M-PAES, and the points found from 5 runs of mc$d$-Prim (Figure 3) clearly illustrates this.

6 Conclusion

Over the range of instances tested, M-PAES was clearly the best approach, being only slightly outperformed by AESSEA on a couple of instances. Its performance was also seen to be robust on these instances, as no parameter tuning was carried out after an initial setting was chosen. Although M-PAES has been shown to perform well on other problems, it seems from these results that in this application, a local-search element may be particularly useful. This may well be explained with reference to an observation recently made by Ehrgott and Klamroth [1]. They found that from a sample of 50 random instances of a random weight bi-objective MST problem, all of them had a connected efficient set under a single exchange operator (although they prove this will not always be true). That is, it was possible to visit the entire efficient set by finding first one (extremal) efficient solution, and then by making single edge exchanges, without encountering any non-efficient solutions. In light of this, we predict that further advances in tackling these difficult constrained mc-MST problems will come from techniques that incorporate a strong local search element, as used in M-PAES. It remains to be seen whether the results presented here scale up to larger problem instances, and whether the number of objectives influences the relative performance of M-PAES, AESSEA, and PAES. However, its seems plausible that the results will scale up as very large d-MST instances have been tackled previously using the operators developed by Raidl that are employed in M-PAES and AESSEA.

On some of the instances considered in this paper, it would not be necessary to use an evolutionary or memetic algorithm at all, because a simple, iterative approach — m*cd*-Prim — was shown to provide very good results in a fraction of the time. This indicates that in previous studies some researchers may have used problem instances that were too easy, for testing their EAs. However, we have also demonstrated that for certain problem instances with constraints that are difficult to meet, the evolutionary algorithms do obtain substantially better solutions than the simple constructive iterative approach. Furthermore, the evolutionary algorithms are able to find points in the non-supported regions of the Pareto front, as was clearly demonstrated using the concave graph generator. Although the more difficult instances considered in this paper may not be realistic-looking telecommunications problems at present, they do serve to demonstrate the robustness of the EA and MA approach to different problem difficulties. In real problems of interest to the telecommunications industry, a great number and variety of constraints must be satisfied. These real problems may well have different types of constraints and objectives than those considered here, but these will probably introduce greater difficulty for tailored constructive heuristics, and necessitate further the use of more general and robust techniques, like the MA used in this paper for the m*cd*-MST problem.

References

[1] M. Ehrgott and K. Klamroth. Connectedness of efficient solutions in multiple criteria combinatorial optimization. *European Journal of Operational Research*, 97:159–166, 1997.

[2] C. M. Fonseca and P. J. Fleming. On the Performance Assessment and Comparison of Stochastic Multiobjective Optimizers. In H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature—PPSN IV*, Lecture Notes in Computer Science, pages 584–593, Berlin, Germany, September 1996. Springer-Verlag.

[3] H. W. Hamacher and G. Ruhe. On spanning tree problems with multiple objectives. *Annals of Operations Research*, 52:209–230, 1994.

[4] J. Knowles and D. Corne. A new evolutionary approach to the degree-constrained minimum spanning tree problem. *IEEE Transactions on Evolutionary Computation*, 4(2):125–134, July 2000.

[5] J. D. Knowles and D. W. Corne. Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 8(2):149–172, 2000.

[6] J. D. Knowles and D. W. Corne. M-PAES: A Memetic Algorithm for Multiobjective Optimization. In *Proceedings of the 2000 Congress on Evolutionary Computation (CEC00)*, pages 325–332, Piscataway, NJ, 2000. IEEE Press.

[7] J. D. Knowles and D. W. Corne. Benchmark problem generators and results for the multiobjective degree-constrained minimum spanning tree problem. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, 2001. (To appear).

[8] J. D. Knowles and D. W. Corne. A comparison of encodings and algorithms for multiobjective minimum spanning tree problems. In *Proceedings of the 2001 Congress on Evolutionary Computation (CEC'01)*. IEEE Press, 2001. (To appear).

[9] R. Prim. Shortest connection networks and some generalizations. *Bell Systems Technical Journal*, 36:1389–1401, 1957.

[10] H. Prüfer. Neuer beweis eines satzes über permutationen. *Arch. Math. Phys.*, 27:742–744, 1918.

[11] G. R. Raidl. An efficient evolutionary algorithm for the degree-constrained minimum spanning tree problem. In *Proceedings of the 2000 Congress on Evolutionary Computation (CEC00)*, pages 104–111, Piscataway, NJ, 2000. IEEE Press.

[12] R. M. Ramos, S. Alonso, J. Sicilia, and C. González. The problem of the optimal biobjective spanning tree. *European Journal of Operational Research*, 111:617–628, 1998.

[13] G. Zhou and M. Gen. Genetic Algorithm Approach on Multi-Criteria Minimum Spanning Tree Problem. *European Journal of Operational Research*, 114(1), April 1999.