Runtime Analysis of Evolutionary Multi-Objective Algorithms Optimizing the Degree and Diameter of Spanning Trees

Wanru Gao, Mojgan Pourhassan, Vahid Roostapour, Frank Neumann

Optimisation and Logistics, School of Computer Science, The University of Adelaide, Adelaide, Australia

Abstract. Motivated by the telecommunication network design, we study the problem of finding diverse set of minimum spanning trees of a certain complete graph based on the two features which are maximum degree and diameter. In this study, we examine a simple multi-objective EA, GSEMO, in solving the two problems where we maximise or minimise the two features at the same time. With a rigorous runtime analysis, we provide understanding of how GSEMO optimize the set of minimum spanning trees in these two different feature spaces.

1 Introduction

Evolutionary algorithms (EAs) have wide application in solving complex problems in various areas such as combinatorial optimization, bioinformatics and engineering. In EA research, the algorithm works with a set of solutions which is called the population and is evolved during the optimization process to cover a so-called Pareto front. Most evolutionary algorithms incorporate certain diversity mechanisms which ensure that the population consists of a diverse set of individuals [3, 16]. By presenting a set of different solutions with acceptable quality to the decision maker, EAs with diversity maximisation provide a better exploration and understanding of the search space. In recent years, EAs with diversity optimisation mechanism have been proposed and examined in both theoretical and practical aspects [9, 8, 14].

There have been many EAs that are applied in solving multi-objective optimisation problems and have gained significant success. Evolutionary multi-objective optimization (EMO) aims at achieving a set of solutions which is used to approximate the so-called Pareto front. The solutions are evaluated based on two or more conflicting objective functions and EAs are suitable in computing several trade-off during a single process. There have been many well-known multi-objective evolutionary algorithms (MOEAs) which include MOEA/D [17], IBEA [18] and NSGA-II/III [6, 5].

In this paper, we consider a simple MOEA which finds a diverse set of Minimum spanning trees (MSTs) with different features for an undirected unweighted complete graph and analyse the algorithm theoretically. Minimum spanning tree problem is a fundamental problem with diverse applications including network design and approximation algorithms design of NP-hard problems [4, 12, 1]. A spanning tree of a graph refers to a subgraph that contains all the vertices in the graph and is a tree. A graph may have many spanning trees. When all edges are assigned weights or lengths, the minimum spanning tree of a graph is the one with the minimum sum of weights. For an unweighted complete graph, all spanning trees are MSTs which have different structures. Although they have the same total weights, they have various features which make them different to the decision makers. There exist many different features other than the total weight that researchers used to evaluate a MST, which include the maximum degree, diameter and depth. The features examined in this paper are the maximum degree and diameter, which evaluate different structural characteristics.

Finding MSTs with different maximum degree and diameter is important for real-world applications such as telecommunication network design with certain connection requirement. When designing a telecommunication network, there are a lot of factors that affect the choice of the decision makers. The degree of each node indicates the number of descendants which is proportional to the workload of that certain node. It is essential to control the maximum degree of all nodes in the tree, which ensure the amount of work that each node has to do is under control [15]. In order to guarantee the communication speed, a MST with low diameter is preferred [11]. The diameter is also important in forcing the reliability constraints which should be taken into consideration of the designer.

Although finding a minimum spanning tree in a given graph is solvable in polynomial time, achieving a MST with certain maximum degree requirement is NP-hard [2]. There have been studies into the problem of approximating the search space of diversifying MSTs based on feature values [13, 7].

In this research, we focus on optimizing these two features in MSTs which are the maximum degree and diameter at the same time. Since maximising or minimising maximum degree leads to a MST with minimum or maximum diameter, it is suitable to consider the problem in a multi-objective space.

The paper is organized as follows. First, we introduce the background of the problem in Section 2. Then in Sections 3 and 4, we examine the MOEA on two multi-objective problems about MSTs. Finally, we finish the paper with some conclusions in Section 5.

2 Preliminaries

In our research, we focus on the multi-objective optimization problem of finding a population containing MSTs with various feature values of a complete graph. Let $G = (V, E)$ be an undirected graph, where V and E denote the set of nodes and set of edges respectively. Define $|V| = n$ and $|E| = m$. A spanning tree of G is defined as a connected subgraph containing all vertices in V without cycles. In this study, we represent a spanning tree as a set of edges and use a bitstring of size m where each bit shows the existence of a certain edge in the subgraph to denote the spanning tree.

We characterize MSTs by two feature values which are the maximum degree and the diameter of an MST. The maximum degree $d(s)$ of an MST s is defined as the maximum value of the degrees of all nodes in V. The diameter $l(s)$ of an MST s is defined as the length of the longest path in s. We also define the number of longest paths in an MST s as $p(s)$.

Considering these two features as objectives, we examine the Global Simple Evolutionary Multi-objective Optimiser(GSEMO) [10] which is presented in Algorithm 1 in optimizing the problem. For the concept of dominance, we use the following definition.

Definition 1 (Dominance). *In multi-objective optimization, there exists a fitness function that maps each solution in the search space X to a vector of real values, i.e. $f : X \to \mathbb{R}^k$. Assume all k objectives should be minimised. For two solution $s, s' \in X$, s is said to weakly dominate s' iff $f_i(s) \leqslant f_i(s')$, where $1 \leqslant i \leqslant k$. S is said to (strictly) dominate s' iff s weakly dominates s' and $f(s) \neq f(s')$.*

The definition of dominance can be adapted to problems where one or more objectives should be maximised.

Definition 2 (Pareto optimality). *A solution s is Pareto optimal if it is not dominated by any other solutions in the search space. The set of all Pareto-optimal solution is called the Pareto set. All optimal objective vectors form the Pareto front in the objective space.*

Algorithm 1 GSEMO

```
1: Choose an initial MST x ∈ {0,1}^m uniformly at random for a certain complete
   graph G with n vertices and m edges.
2: Let P := x
3: while stopping criteria not met do
4:    Pick s from P uniformly at random.
5:    Create an offspring s' by flipping each bit in s with probability 1/m.
6:    if s' is not dominated by any individual in P then
7:       Add s' to P, and remove all individuals weakly dominated by s' from P.
8:    end if
9: end while
```

We focus our analysis on the simple multi-objective EA which is GSEMO proposed by Giel [10] because of its simplicity and suitability for the theoretical analysis. The algorithm starts with an MST which is selected uniformly at random for the complete graph G. Before the stopping criteria is reached, the algorithm selects a solution s uniformly at random from population P and an offspring s' is generated by flipping each bit of s with probability $1/m$. In the case where s' is not dominated by any solution in P, it is added to P. The new population contains only non-dominated solutions.

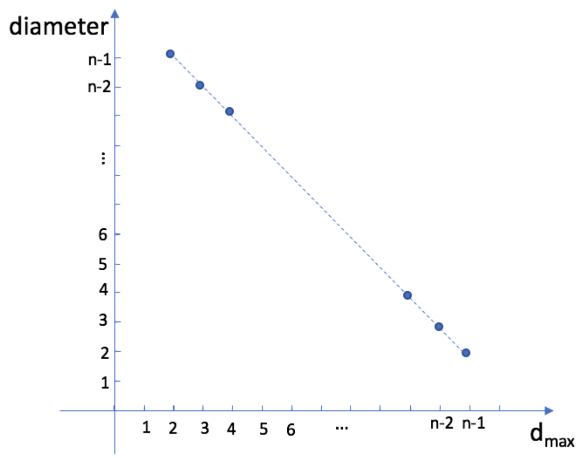

Fig. 1: The Pareto front for the multi-objective problem of maximising the diameter and maximising the max degree of a MST.

The algorithm is examined in terms of the number of generations until it has achieved a population that covers the whole Pareto front. The expected optimisation time refers to the expected number of iterations to reach this goal.

3 The Max-Max problem

We look into two multi-objective problems considering these two features. In the first problem we aim at maximising both the diameter and the maximum degree at the same time, which is referred to as the Max-Max problem in this paper. The dominance definition for the Max-Max problem is defined as follows.

Definition 3 (Domination for Max-Max Problem). *For two MSTs s and s' of an unweighted complete graph G, in the* Max-Max *problem, s dominates s' iff $d(s) \geq d(s')$ and $l(s) \geq l(s')$.*

Lemma 1. *Let s be a Pareto optimal solution of the* Max-Max *problem, then $d(s) + l(s) = n + 1$, where n denotes the number of nodes in the graph.*

Proof. Assume the minimum spanning tree with the maximum degree is s and its diameter and maximum degree are represented as $l(s)$ and $d(s)$. In MST s, the longest path has length $l(s)$ which has $l(s) + 1$ nodes on it. Then there are another $n - (l(s) + 1)$ nodes which are not on the path. In order to maximise $d(s)$, these nodes should be connected to one of the nodes on the path except the tailing ones. Hence,

$$d(s) = 2 + n - (l(s) + 1) = n - l(s) + 1.$$

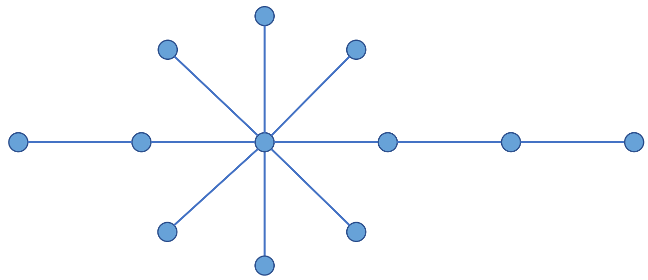

Fig. 2: A Pareto solution with degree 8 and diameter 5

The sum of the diameter and the maximum degree equals to $l(s)+n-l(s)+1 = n+1$. □

According to Lemma 1, the Pareto front of the MAX-MAX problem is as shown in Figure 1. It is easy to see that each Pareto solution consists of a star node with degree d and a longest path of length l as shown in Figure 2. Note that for a specific degree and diameter, the Pareto solution is not unique. However, a solution is Pareto optimal if and only if

1. It has at most one node with degree more than 2.
2. All the nodes with degree 2 and more lie on the longest path.

Moreover, for each diameter value, Algorithm 1 keeps only one solution because of the dominance definition. Hence, the size of the population produced by the algorithm is at most $n-2$. The next theorem considers the expected time to find the Pareto front using Algorithm 1.

Theorem 1. *Algorithm 1 finds all the Pareto optimal solutions of the* MAX-MAX *problem in expected time $O(n^2m^2)$.*

Proof. Let $s_P \in P$ denote a Pareto optimal solution in the population with diameter $l(s_P)$. We consider the proof in the following two phases. The first phase is to maintain P such that it contains at least one Pareto optimal solution. The second phase is to find other Pareto optmial solutions starting from s_P. We prove that each phase needs expected time $O(n^2m^2)$ to be completed.

Now let $s' \in P$ be a solution with the highest diameter $l(s') < n-1$. The mutation step on s' that detaches a leaf from a node with degree more than 2 and attaches it to one side of the longest path increases $l(s')$ by one. Since the probability of any single bit flip is $\frac{1}{m}$, the probability of such a mutation would be $\frac{1}{em^2}$. On the other hand, the size of P is upper bounded by n and the probability of selecting s' for mutation is at least $\frac{1}{n}$. Hence, after expected time $O(nm^2)$, the diameter of s' is increased by one. Furthermore, the diameter of s' is at least 2. Therefore, we need at most $n-2$ such mutations to obtain the Pareto optimal solution with maximum degree 2 and diameter $n-1$. It implies that Algorithm 1 needs expected time $O(n^2m^2)$ to complete the first phase.

Now we analyse the second phase and assume that there is at least one Pareto optimal solution s_P in the population. Assume that the Pareto optimal solution with diameter $l(s_P) - 1$ is not included in P yet. In this case, the mutation step on s_P that removes a leaf from one side of the longest path and connects it to the node with the highest degree will produce a new Pareto optimal solution with diameter $l(s_P) - 1$ and maximum degree $d(s_P) + 1$. Similar to the argument in the first part of the proof, the algorithm needs expected time $O(nm^2)$ to complete this mutation. Furthermore, from the first phase, it is known that the solution s_P with diameter $n - 1$ exists in P. Hence, the algorithm is able to produce all the Pareto optimal solutions gradually, starting from s_P. Since the size of the Pareto set is $n - 2$, Algorithm 1 finds all the Pareto optimal solution in expected time $O(n^2m^2)$. □

4 The Min-Min problems

In this section, we investigate the second problem, in which both feature values are minimised at the same time. The minimum diameter happens when the MST has a star structure where the diameter is 2 and the node in the centre has the maximum degree $n - 1$. The minimum maximum degree happens when the graph is a single path. In this case, the maximum degree is 2 and the diameter is $n - 1$. The general dominance definition is adapted for the Min-Min problem as follows.

Definition 4 (Dominance for the Min-Min Problem). *For two MSTs s and s' of an unweighted complete graph G, in the* Min-Min *problem, a solution s is said to dominate solution s' iff $d(s) \leq d(s')$ and $l(s) \leq l(s')$.*

Based on the fact that the diameter is either even or odd, the Pareto optimal MST with diameter l have different structures. Figure 3 shows the structure of an optimal MST with odd diameter. The optimal MST with even diameter only contains multiple subtrees with the same depth $l/2$.

the Pareto front for this problem is not as simple as the Pareto front for the Max-Max problem. Having a solution s in the Pareto front, the adjacent solution with a smaller diameter, named s', can have $d(s') = d(s) + i$ and $l(s') = l(s) - j$ for some $i \geq 1$ and $j \geq 1$. Therefore, it is not always possible to find the solution s' by means of a 2-bit flip on s.

In order to overcome this problem we use a different definition of dominance in analysing the Min-Min problem, which still leads to a population of linear size. The new definition of dominance is presented in Definition 5, where $p(s)$ is the number of longest paths. Furthermore, it should be noted that we consider Algorithm 1 with the new definition of dominance (instead of weak dominance in lines 6 and 7).

Definition 5 (Extended dominance for the Min-Min Problem). *In the* Min-Min *problem, for two MSTs s and s' of a complete graph, s dominates s' iff $l(s') = l(s) \wedge d(s') = d(s) \wedge p(s') \leq p(s)$ or $l(s') < l(s) \wedge d(s') < d(s)$.*

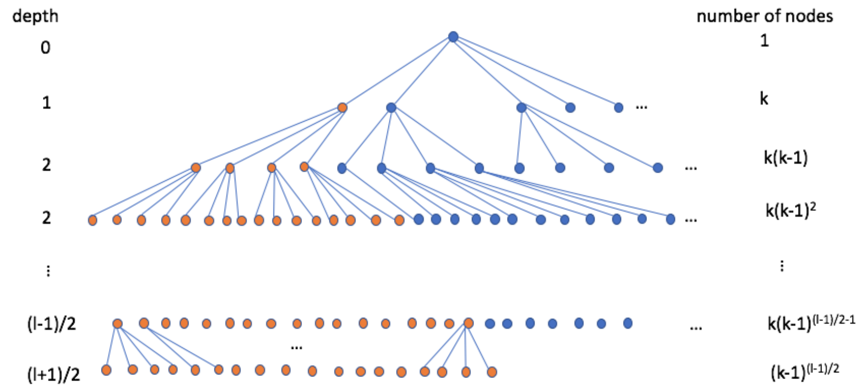

Fig. 3: The MST with odd diameter and maximum degree minimised, where l denotes the diameter and k denotes the maximum degree in the MST. The nodes coloured in orange are from a single subtree of the root.

We define an *Extended Pareto optimal solution* and the *Extended Pareto front* to be a Pareto optimal solution and the Pareto front with the new definition of dominance in Definition 5. Then in Lemma 2 we prove that the Extended Pareto front set is a superset of the original Pareto front, which is defined by the original definition of dominance.

Lemma 2. *The Extended Pareto front is a super set of the Pareto front for the* Min-Min *problem.*

Proof. According to Definition 4, a Pareto optimal solution s of the Min-Min problem should fulfil the requirement that $\nexists s'$ dominates s where s' is any other MST of the same graph. This indicates that $\nexists s'$, where $d(s') \leq d(s)$ and $l(s') \leq l(s)$.

Assume there exists a MST s'' that dominates s according to Definition 5. Then either $l(s) = l(s'') \wedge d(s) = d(s'') \wedge p(s) \leq (s'')$ or $l(s) < l(s'') \wedge d(s) < d(s'')$ is true. For maximum degree and diameter, it should fulfil that $l(s) \leq l(s'') \wedge d(s) \leq d(s'')$, which is contradict to the fact that $\nexists s'$ in the search space, where $d(s') \leq d(s)$ and $l(s') \leq l(s)$.

Therefore, the MST s is not dominated by any other solutions in the extended Pareto front which means it should be included in the Pareto set of the extended problem.

□

In the following, we analyse the performance of the algorithm in finding the the whole Extended Pareto front. Since this set is a super set for the original Pareto front, we are also analysing the performance of the algorithm in finding the original Pareto front. Lemma 3 proves an upper bound on the size of the population during the optimisation process.

Lemma 3. *The population size is upper bounded by* $2n$*.*

Proof. Here we prove that the maximum size of the population is $2n - 5 < 2n$. According to Definition 5, solution s does not dominate solution s' if $d(s) = d(s')$ and $l(s) \leq l(s')$. Similarly, s' is not dominated by s when $d(s) \leq d(s')$ and $l(s) = l(s')$. Moreover, for each specific combination of diameter and maximum degree, the algorithm keeps only one solution.

Let P be the population of an arbitrary iteration during the process. We partition P to at most $n-2$ subsets $P^i = \{s_1^i, \cdots, s_{k_i}^i\}$,$2 \leq i \leq n-1$, such that for any $s \in P^i$, $d(s) = i$. Moreover, for any $j_1 < j_2 \leq k_i$ we have $l(s_{j_1}^i) > l(s_{j_2}^i)$. For each subset, we have $|P^i| = k_i \leq l(s_1^i) - l(s_{k_i}^i) + 1$. Without loss of generality, let all the subsets have at least one solution. Since P is the set of non-dominated solutions, for any $2 \leq i \leq n-2$ we have $l(s_{k_i}^i) \geq l(s_1^{i+1})$. Otherwise, $s_{k_i}^i$ dominates s_1^{i+1}. Hence, for any subsets P^i and P^{i+1}, we have

$$|P^i \cup P^{i+1}| \leq l(s_1^i) - l(s_{k_i}^i) + 1 + l(s_1^{i+1}) - l(s_{k_{i+1}}^{i+1}) + 1 \leq l(s_1^i) - l(s_{k_{i+1}}^{i+1}) + 2.$$

With the same argument we have

$$\begin{aligned} |P| = |\bigcup_{j=2}^{n-1} P^j| &\leq l(s_1^2) - l(s_{k_{n-1}}^{n-1}) + (n-2) \\ &\leq (n-1) - 2 + (n-2) \\ &\leq 2n - 5 \end{aligned}$$

□

In any tree with n nodes, there is only one path between any two nodes. Hence the total number of paths in a tree, which is an upper bound for the number of paths with length d, is

$$\binom{n}{2} \leq n^2.$$

Therefore, in a solution with diameter d, the number of paths with length d is upper bounded by n^2.

In Lemma 4 and Lemma 5 we give some properties about Extended Pareto optimal solutions and show how they are produced in $O(n^3 m^2)$ for each diameter size. In order to simplify the presentation, we first define Important-Objective-Positions (Definition 6), which refers to the positions with diameter l and maximum degree d, at which an extended Pareto optimal solution can be formed.

Definition 6 (Important-Objective-Positions (IOP)). *We define a point* (l, d) *in the diameter-degree space to be an Important-Objective-Position (IOP) if the extended Pareto set includes a solution with diameter* l *and degree* d*.*

Lemma 4. *If a point* (l, d) *in the diamater-degree space is an IOP and the point* $(l, d+1)$ *is not an IOP, then the point* $(l(s) - 1, d)$ *is an IOP.*

Proof. Since the point (l, d) is an IOP, by definition, an extended Pareto optimal solution s exists such that $l(s) = l$ and $d(s) = d$. A solution with diameter $l(s)-1$ and maximum degree $d(s)$ exists because reducing the number of longest paths in solution s either results in a solution with a larger maximum degree (which we have assumed that does not belong to the extended Pareto set), or a solution with smaller diameter. Moreover, a solution with diameter $l(s) - 1$ and maximum degree $d(s)$ can only be dominated by a solution with the same maximum degree and diameter, or a solution s' with maximum degree $d(s') < d(s)$ and diameter $l(s') < l(s)-1$ (Definition 5). If solution s' exists, then it would have dominated solution s as well, which contradicts with the assumption that s is an extended Pareto optimal solution. Therefore, a solution with diameter $l(s)-1$ and maximum degree $d(s)$ can only be dominated with a solution with the same maximum degree and diameter, which implies that the point $(l(s) - 1, d)$ is an IOP. □

Lemma 5. *Assume that points $(l, d + i)$, for $0 \leq i < k$ and $k > 1$, are IOPs and, a solution s with $l(s) = l$ and $d(s) = d$, is in the population. In expected time $O(n^3m^2)$, all Pareto optimal solutions with diameter l and also a solution s' with $l(s') = l - 1$ and $d(s') = d + k - 1$ are added to the population.*

Proof. Since the position $(l(s), d(s))$ is an IOP, the solution s can only be removed from the population if a solution with the same diameter and maximum degree and a smaller number of longest paths is found (Definition 5).

In a solution s, there always exists at least one pair of 2-bit flips that reduces the number of longest paths, $p(s)$. This can be done by disconnecting a leaf of one of these paths and connecting it to an inner node. At each step, with probability $\frac{1}{|P|}$ the assumed solution is selected for mutation, where $|P|$ is the size of the population. Moreover, while there exist inner nodes with degree less than $d(s)$, with probability $\frac{1}{e \cdot m^2}$ a proper 2-bit flip happens, which reduces $p(s)$ without increasing the maximum degree of the solution. Due to Definition 5, solution s is dominated and replaced by the new solution. The same process with reducing $p(s)$ continues until the algorithm reaches a solution s^0 that belongs to the extended Pareto front and stays in the population. Denoting the total reduction on the number of longest paths by $\Delta_{s^0} = p(s) - p(s^0)$, we can observe that the expected time until reaching the solution s^0 is $O(|P|m^2\Delta_{s^0})$.

We define solutions s^i, $i < k$, to be extended Pareto optimal solutions of diameter $l(s)$ and degree $d(s) + i$. We also define $\Delta_{s^i} = p(s^{i-1}) - p(s^i)$, $1 \leq i < k - 1$ as the total difference on the number of longest paths between solutions s^{i-1} and s^i. With similar analysis we can show that after reaching the solution s^i, $i < k-1$, at each step with probability $\frac{1}{|P|m^2}$ a solution with degree $d(s^i)+1$, diameter $l(s^i)$ and number of longest paths $p(s^i) - 1$ is produced, which is, due to Definition 5, either accepted by the algorithm, or dominated by a solution with the same degree and diameter, but a smaller number of longest paths. This process continues until reaching a solution with minimum number of longest paths, which implies that a solution s^{i+1} is reached by the algorithm in expected time $O(|P|m^2\Delta_{s^1})$. This means that all extended Pareto optimal solutions with diameter $l(s)$ can be found in expected time

$$\sum_{i=0}^{k-1} O(|P|m^2 \Delta_{s^i}).$$

Moreover, since the solution s^{k-1} is the extended Pareto optimal solution with diameter $l(s)$ that maximises $d(s)$, it only contains one longest path. Therefore, moving an edge from it results in obtaining the solution s' with $l(s') = l(s) - 1$ and $d(s') = d(s^{k-1})$. This would also happen in expected time $O(|P|m^2 p(s^{k-1}))$. Together with the expected time of finding extended Pareto optimal solutions with diameter $l(s)$, the total expected time of finding all k extended Pareto optimal solutions with diameter $l(s)$ and also a solution s' with diameter $l(s') = l(s) - 1$ and maximum degree $d(s) + k - 1$ would be

$$\sum_{i=0}^{k-1} O(|P|m^2 \Delta_{s^i}) + O(|P|m^2 p(s^{k-1})) = O(|P|m^2 p(s)).$$

The equality holds because the total number of longest paths that have been reduced in the process is $\sum_{i=0}^{k-1} \Delta_{s^i} + p(s^{k-1}) = p(s)$. Since the number of longest paths in solution s is upper bounded by n^2 and the population size is upper bounded by $2n$ (Lemma 3), the obtained expected time is upper bounded by $O(n^3 m^2)$. □

Now we present the main theorem of this section, in which, starting from a solution with maximum degree of 2 (a path), the expected time until finding all Pareto front set is analysed.

Theorem 2. *Starting with a population that contains a solution s with $d(s) = 2$, Algorithm 1 finds the Pareto set of the* Min-Min *problem in expected time $O(n^4 m^2)$.*

Proof. Firstly, we prove that Algorithm 1 finds the extended Pareto set in expected time $O(n^4 m^2)$.

Since the maximum degree of a minimum spanning tree on a graph of at least three nodes cannot be less than 2, solution s belongs to the extended Pareto front. This solution is a path of length $n - 1$, which implies that $l(s) = n - 1$ and the corresponding IOP is $(n - 1, 2)$.

Having a solution s at IOP position $(l(s), d(s))$, from Lemma 5, we know that in expected time $O(n^3 m^2)$, all k extended Pareto optimal solutions with diameter $l(s)$ are added to the population in addition to a solution s' with diameter $l(s) - 1$ and degree $d(s) + k - 1$. The largest maximum degree among solutions with diameter $l(s)$ would be $d(s) + k - 1$, which implies that a diameter-degree position $(l(s), d(s) + k)$ is not an IOP. Therefore, by lemma 4 we know that the position $(l(s) - 1, d(s) + k - 1)$ is an IOP. Since solution s' is placed at this position, it can be used for Lemma 5 and diameter size $l(s) - 1$. We can use similar argument for smaller diameter sizes. Since we start with a diameter size of $n-1$, all extended Pareto optimal solutions for all diameter sizes are found in expected time $O(n^4m^2)$.

Since the extended Pareto front is a superset of the Pareto front, the dominanted solutions according to Definition 4 should be eliminated before the Pareto set of the MIN-MIN problem is achieved. As the population size is upper bounded by $2n$, getting rid of all dominated solutions takes expected $O(n^2)$ time. Hence, the statement of the theorem is proved.

□

5 Conclusions

The MOEAs used to optimise several objective functions always involve a set of solutions which approximates the so-called Pareto front. These algorithms are suitable in dealing with conflicting objective functions. In this paper, we examine a simple multi-objective optimisor on two bi-objective optimisation problems about MSTs of a complete graph. Inspired by the real-world application in telecommunication, we focus on the MAX-MAX and MIN-MIN problems which provide insights in dealing with the trade-off between optimising the features of maximum degree and diameter. With a rigorous runtime analysis, we provide a better understanding of the search space and the computational complexity of such problems.

References

1. Abuali, F.N., Schoenefeld, D.A., Wainwright, R.L.: Designing telecommunications networks using genetic algorithms and probabilistic minimum spanning trees. In: Proceedings of the 1994 ACM Symposium on Applied Computing. pp. 242–246. SAC '94, ACM, New York, NY, USA (1994). https://doi.org/10.1145/326619.326733, http://doi.acm.org/10.1145/326619.326733
2. Bui, T.N., Zrncic, C.M.: An ant-based algorithm for finding degree-constrained minimum spanning tree. In: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation. pp. 11–18. GECCO '06, ACM, New York, NY, USA (2006). https://doi.org/10.1145/1143997.1144000, http://doi.acm.org/10.1145/1143997.1144000
3. Chaiyaratana, N., Piroonratana, T., Sangkawelert, N.: Effects of diversity control in single-objective and multi-objective genetic algorithms. J. Heuristics **13**(1), 1–34 (2007)
4. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, Third Edition. The MIT Press, 3rd edn. (2009)
5. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. IEEE Transactions on Evolutionary Computation **18**(4), 577–601 (Aug 2014). https://doi.org/10.1109/TEVC.2013.2281535
6. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Transactions on Evolutionary Computation **6**(2), 182–197 (April 2002). https://doi.org/10.1109/4235.996017

7. Dekker, A., Pérez-Rosés, H., Pineda-Villavicencio, G., Watters, P.: The maximum degree & diameter-bounded subgraph and its applications. Journal of Mathematical Modelling and Algorithms **11**(3), 249–268 (Sep 2012). https://doi.org/10.1007/s10852-012-9182-8, https://doi.org/10.1007/s10852-012-9182-8
8. Gao, W., Nallaperuma, S., Neumann, F.: Feature-based diversity optimization for problem instance classification. In: Parallel Problem Solving from Nature - PPSN XIV - 14th International Conference, Edinburgh, UK, September 17-21, 2016, Proceedings. pp. 869–879 (2016). https://doi.org/10.1007/978-3-319-45823-6_81, https://doi.org/10.1007/978-3-319-45823-6_81
9. Gao, W., Neumann, F.: Runtime analysis for maximizing population diversity in single-objective optimization. In: Genetic and Evolutionary Computation Conference, GECCO '14, Vancouver, BC, Canada, July 12-16, 2014. pp. 777–784 (2014). https://doi.org/10.1145/2576768.2598251, http://doi.acm.org/10.1145/2576768.2598251
10. Giel, O.: Expected runtimes of a simple multi-objective evolutionary algorithm. In: The 2003 Congress on Evolutionary Computation, 2003. CEC '03. vol. 3, pp. 1918–1925 Vol.3 (Dec 2003). https://doi.org/10.1109/CEC.2003.1299908
11. Gouveia, L., Magnanti, T.L.: Network flow models for designing diameter-constrained minimum-spanning and steiner trees. Networks **41**(3), 159–173. https://doi.org/10.1002/net.10069, https://onlinelibrary.wiley.com/doi/abs/10.1002/net.10069
12. Khan, M., Pandurangan, G., Kumar, V.S.A.: Distributed algorithms for constructing approximate minimum spanning trees in wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems **20**(1), 124–139 (Jan 2009). https://doi.org/10.1109/TPDS.2008.57
13. Könemann, J., Levin, A., Sinha, A.: Approximating the degree-bounded minimum diameter spanning tree problem. Algorithmica **41**(2), 117–129 (2005). https://doi.org/10.1007/s00453-004-1121-2, https://doi.org/10.1007/s00453-004-1121-2
14. Neumann, A., Gao, W., Doerr, C., Neumann, F., Wagner, M.: Discrepancy-based evolutionary diversity optimization. In: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2018, Kyoto, Japan, July 15-19, 2018. pp. 991–998 (2018). https://doi.org/10.1145/3205455.3205532, http://doi.acm.org/10.1145/3205455.3205532
15. Robins, G., Salowe, J.S.: Low-degree minimum spanning trees. Discrete Comput. Geom **14**, 151–165 (1999)
16. Ursem, R.K.: Diversity-guided evolutionary algorithms. In: PPSN. pp. 462–474 (2002)
17. Zhang, Q., Li, H.: Moea/d: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on Evolutionary Computation **11**(6), 712–731 (Dec 2007). https://doi.org/10.1109/TEVC.2007.892759
18. Zitzler, E., Laumanns, M., Bleuler, S.: A tutorial on evolutionary multiobjective optimization. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) Metaheuristics for Multiobjective Optimisation. pp. 3–37. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)