Theoretical Computer Science 717 (2018) 26–36

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Parameterized complexity of team formation in social networks ☆

Robert Bredereck [a,b], Jiehua Chen [a,c,*], Falk Hüffner [a], Stefan Kratsch [d]

[a] *Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Berlin, Germany*
[b] *Department of Computer Science, University of Oxford, Oxford, United Kingdom*
[c] *Dept. Industrial Engineering and Management, Ben-Gurion University of the Negev, Beer Sheva, Israel*
[d] *Institut für Informatik I, Universität Bonn, Germany*

ARTICLE INFO

Article history:
Received 15 November 2016
Received in revised form 15 May 2017
Accepted 21 May 2017
Available online 7 June 2017

Keywords:
Computational complexity analysis
Parameterized complexity analysis
Team formation on social networks

ABSTRACT

Given a task that requires some skills and a social network of individuals with different skills, the Team Formation problem asks to find a team of individuals that together can perform the task, while minimizing communication costs. Since the problem is NP-hard, we identify the source of intractability by analyzing its parameterized complexity with respect to parameters such as the total number of skills k, the team size l, the communication cost budget b, and the maximum vertex degree Δ. We show that the computational complexity strongly depends on the communication cost measure: when using the weight of a minimum spanning tree of the subgraph formed by the selected team, we obtain fixed-parameter tractability for example with respect to the parameter k. In contrast, when using the diameter as measure, the problem is intractable with respect to any single parameter; however, combining Δ with either b or l yields fixed-parameter tractability.

1. Introduction

Assembling teams based on required skills is a classic management task. Recently, it has been suggested to take into account not only the covering of the required skills, but also the expected communication costs (see Lappas et al. [13] for a survey). This cost can be estimated based on a given edge-weighted social network, where a low weight value on an edge between two individuals indicates a low communication cost. For example, edge weights can reflect distance in an organizational chart or the number of joint projects completed.

Lappas et al. [12] formalized the setting as the optimization problem of minimizing the communication cost and studied two cost measures: the diameter (Diam) and the weight of a minimum spanning tree (Mst). For our complexity analysis, we formulate it as a decision problem by fixing the maximum team size.

Diam-Team Formation
Input: An undirected graph $G=(V,E)$ with edge-weight function $w\colon E\to\mathbb{N}$, a set T of k skills, a skill function $S\colon V\to 2^T$, a team size $l\in\mathbb{N}$, and a budget $b\in\mathbb{N}$.
Question: Is there a subset $V'\subseteq V$ with $|V'|\le l$ such that $\bigcup_{v\in V'} S(v)=T$ and the w-weighted diameter of the induced subgraph $G[V']$ is at most b?

☆ An extended abstract version appeared in the proceedings of the 11[th] International Conference on Algorithmic Aspects in Information and Management (AAIM), Bergamo, Italy, July of 2016.

* Corresponding author
E-mail addresses: robert.bredereck@tu-berlin.de (R. Bredereck), jiehua.chen2@gmail.com (J. Chen), kratsch@cs.uni-bonn.de (S. Kratsch).

http://dx.doi.org/10.1016/j.tcs.2017.05.024
0304-3975/

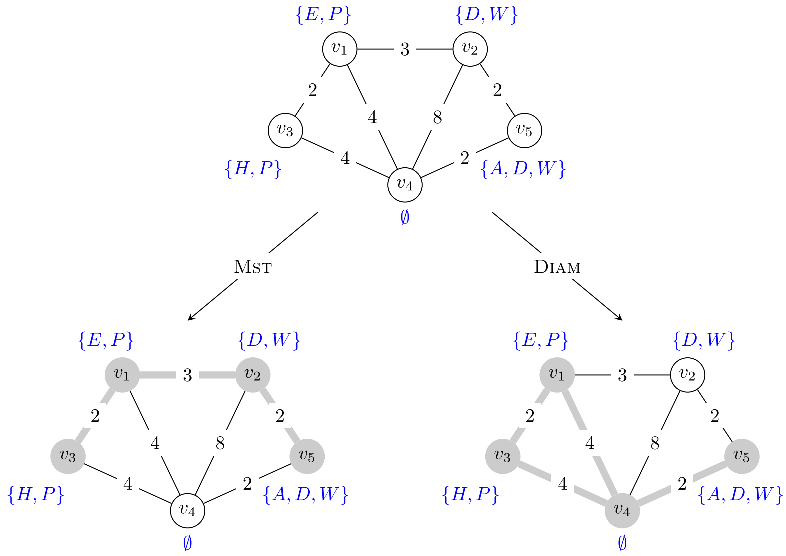

Fig. 1. A TEAM FORMATION example: The upper figure shows a social network of five potential team members and six skills, "algorithms" (A), "data bases" (D), "software engineering" (E), "hardware support" (H), "programming" (P), and "web programming" (W). When minimizing the weight of a minimum spanning tree of the subgraph induced by a team (MST), the team with members v_1, v_2, v_3, v_5 has the lowest cost, 7 (see the lower left figure). However, when minimizing the diameter of the subgraph induced by a team (DIAM), it is worthwhile to replace v_2 with v_4—who has no specific skill—to reduce the diameter of 7 in $G[\{v_1, v_2, v_3 v_5\}]$ to the diameter of 6 in $G[\{v_1, v_3, v_4, v_5\}]$ (see the lower right figure).

Here, the *diameter* of an edge-weighted graph G, denoted as DIAM(G), is the maximum distance between any two vertices in the input graph and the *distance* between two vertices is the minimum sum of the weights of the edges along any path between these two vertices. Our formulation of the team formation problem allows to choose individuals (vertices) that do not contribute any skills, but serve as intermediate vertices to lower overall communication costs. We further assume w.l.o.g. that no individual has a skill that is not in the request set T.

The *weight of a minimum spanning tree* of graph G, MST(G), is the smallest sum of the weights of the edges in any spanning tree of G. We define the corresponding MST-TEAM FORMATIONproblem by replacing "diameter" in the definition of DIAM-TEAM FORMATIONwith "weight of a minimum spanning tree".

Fig. 1 illustrates an example for the DIAM-TEAM FORMATION and MST-TEAM FORMATION problems. Lappas et al. [12] showed that both problems are NP-complete. Experiments on DIAM-TEAM FORMATION, MST-TEAM FORMATION and similar team formation problems so far use heuristic algorithms [1,5,12,9,14]. However, it might be that instances encountered in practice are actually easier than a one-dimensional complexity analysis suggests, and can be solved optimally. For example, it might be reasonable to assume that only a small number of skills is required. Thus, we try to identify the sources of intractability through a parameterized complexity analysis.

1.1. Optimization variant

There are two natural ways to define approximate solutions of our team formation problem. First, to allow solutions with larger communication costs. This leads to the MINCOST-ζ-TEAM FORMATION problem, ζ being either DIAM or MST, which asks for a vertex subset $V' \subseteq V$ with $|V'| \le l$ such that $\bigcup_{v \in V'} S(v) = T$ and the communication cost $\zeta(G[V'])$ is minimized. Second, to allow solutions with larger teams. This leads to the MINTEAMSIZE-ζ-TEAM FORMATION problem, which asks for a minimum vertex subset $V' \subseteq V$ such that $\bigcup_{v \in V'} S(v) = T$ and $\zeta(G[V']) \le b$.

Cost measure "diameter" Arkin and Hassin [2] studied MINCOST-DIAM-TEAM FORMATION with unlimited team size l under the name MULTIPLE-CHOICE COVER. They showed that even when no skill is allowed to be covered by more than three team members, the problem still cannot be approximated with a constant-factor error guarantee, unless P = NP. However, when the weights satisfy the triangle inequality, a 2-approximation is possible; this bound is sharp [2].

Cost measure "minimum spanning tree" As already mentioned by Lappas et al. [12], the MINCOST-MST-TEAM FORMATION problem with an unlimited team size l is equivalent to the GROUP STEINER TREE problem: given an undirected edge-weighted graph $G = (V, E)$ and vertex subsets (*groups*) $g_i \subseteq V$, $1 \le i \le k$, find a subtree $T = (V_T, E_T)$ of G such that $V_T \cap g_i \neq \emptyset$ for all $1 \le i \le k$ and the cost $\sum_{e \in E_T} w(e)$ is minimized. Clearly, each group of GROUP STEINER TREE corresponds to a subset of vertices in MST-TEAM FORMATION that have a particular skill. From this relation to GROUP STEINER TREE and an inapproximability result by Halperin and Krauthgamer [11], we obtain that it is unlikely that MINCOST-MST-TEAM FORMATION can be approximated to a factor of $O(\log^{2-\epsilon} k)$ for any $\epsilon > 0$, where k is the number of skills to be covered.

Despite the polylogarithmic inapproximability result, we can obtain fixed-parameter tractability for the parameter "number k of skills to be covered". First, we reduce the Mst-Team Formation problem with limited team size l to the Mst-Team Formation problem with an unlimited team size by adding a large weight W (for example the sum over all edge weights) to each edge weight and adding $l \cdot W$ to the budget. Then, by the relation between Mst-Team Formation and Group Steiner Tree, we can think of the resulting instance as a Group Steiner Tree instance, which can be solved by reducing it to Steiner Tree: introduce a new vertex for each group and connect it to each vertex contained in this group by an edge with very high weight. The resulting Steiner Tree instance can be solved using inclusion–exclusion in $O^*(2^k)$ time and polynomial space when the edge weights are integers [15] (the O^* notation omits factors polynomial in the input size); for arbitrary weights, it can be solved in $O^*(3^k)$ time and exponential space by dynamic programming [7].

1.2. Further related work

Our team formation problem can be generalized in different ways. First, we can require each skill to be covered by a given number of team members instead of once. Li and Shan [14] proposed three heuristics for this problem; Gajewar and Sarma [9] studied it with the objective of maximizing the *collaborative compatibility*, an alternative to Diam and Mst. Here, the collaborative compatibility is the sum of the weights of all edges in the subgraph induced by the team divided by the team size (the number of vertices in the subgraph). They showed that this version is also NP-hard and provided a 1/3-approximation algorithm. Second, we can additionally require the workload to be balanced within the team [1,5]. Recently, Gutiérez et al. [10] give a comprehensive summary of related work. They also study a generalized variant of the problem, multiple team formation, where each individual may have fractional skill ability and multiple projects are to be covered with a quadratic optimization goal. They propose three heuristics for their problem and analyze them experimentally.

A number of experimental studies examine the validity of these models, using data for example from bibliography databases [12,1,9,14] or the GitHub programming collaboration platform [5].

Diam-Team Formation and Mst-Team Formation have also applications in keyword search in relational databases [17]: the vertices in the graph correspond to tables, edges represent foreign key relationships, and skills model keywords that match the table. A subgraph covering all keywords with small communication costs helps to create efficient SQL queries.

1.3. Parameterized complexity

Parameterized algorithmics analyzes problem difficulty not only in terms of the input size, but also for an additional parameter, typically an integer p. Formally, an instance of a parameterized problem is a tuple of the unparameterized instance I and the parameter p. A parameterized problem with parameter p is *fixed-parameter tractable* (FPT) if there is an algorithm that decides each instance (I, p) in $f(p) \cdot |I|^{O(1)}$ time, where f is a computable function depending only on p; we call this algorithm a *fixed-parameter algorithm*. In such case, we say that our problem can be solved in FPT-time for the parameter p. Clearly, if the problem is NP-hard, we must expect f to grow superpolynomially.

There are parameterized problems for which there is good evidence that no fixed-parameter algorithms exist. Analogously to the concept of NP-hardness, the concept of W[1]-*hardness* was developed. It is widely assumed that a W[1]-hard problem cannot have a fixed-parameter algorithm (hardness for the classes W[t], $t \geq 2$ has the same implication). To show that a problem is W[t]-hard, a *parameterized reduction* from a known W[t]-hard problem can be used. This is a reduction that runs in FPT-time and maps the parameter p to a new parameter p' that is upper-bounded by some function $g(p)$. We refer to recent text books [4,6,8,16] for details on parameterized complexity theory and W[t]-complete problems.

1.4. Contributions

We focus on the parameterized complexity of Diam-Team Formation, which has, to the best of our knowledge, not been considered before. We consider parameters that are related to the communication cost and to the input graph: the number k of skills to be covered, the cost budget b, the maximum vertex degree Δ, and the team size l.

For the parameter l, Diam-Team Formation is W[2]-hard even with either constant budget b or constant maximum degree Δ (Proposition 1). For the parameter k, while Mst-Team Formation is fixed-parameter tractable, Diam-Team Formation is W[1]-hard even on graphs of maximum vertex degree three and with unrestricted team size l (Theorem 1). For the combined parameter $l + k$, Diam-Team Formation is W[1]-hard even if the cost budget b is one (Proposition 3). Concerning the parameter maximum degree Δ, we find that the problem is NP-hard even if the graph is a caterpillar with maximum degree $\Delta = 3$ (Proposition 2), where a caterpillar is a tree in which all the vertices are within distance one of a central path. Our results rule out fixed-parameter tractability for all considered single parameters and several parameter combinations.

By our parameterized hardness reductions, we obtain that MinCost-Diam-Team Formation is inapproximable even when we allow for a superpolynomial running time factor in the team size l, even on complete graphs or on stars (Corollary 1). MinTeamsize-Diam-Team Formation is inapproximable even when we allow for a superpolynomial running time factor in the number k of skills, even on graphs with maximum degree $\Delta = 3$ (Corollary 2).

Geared towards robustness, we also consider the situation where the subgraph induced by the team is two-connected (that is, between each two team members, there are at least two edge-disjoint paths). We find that unless FPT = W[1], it is

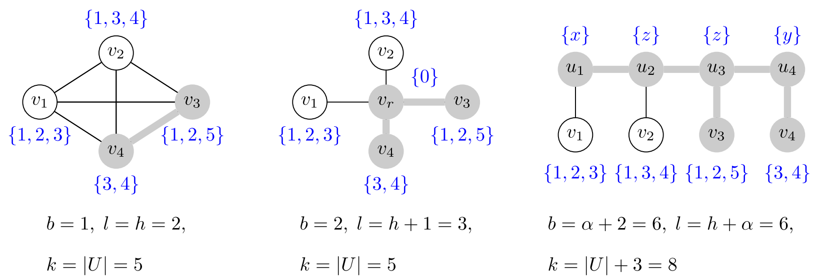

Fig. 2. Examples for the reductions given in the proof of Observation 1. A Set Cover instance is given by $I = (\mathcal{F}, U, h = 2)$. It is a yes-instance: S_3 and S_4 cover all the five elements. The left figure denotes a Diam-Team Formation instance for complete graphs and diameter bound one. The middle figure denotes a Diam-Team Formation instance for stars and diameter bound two. The middle figure denotes a Diam-Team Formation instance for caterpillars and diameter bound six. The vertices marked in gray form a desired team.

unlikely that there is an algorithm that forms a team of size at most l, covering all k skills and inducing a two-connected subgraph, in $f(k+l) \cdot |I|^{O(1)}$ time, where $|I|$ denotes the size of our input instance (Theorem 2).

On the positive side, we provide some tractability results: Diam-Team Formation can be solved in $O^*(\Delta^{\Delta^b} \cdot \text{dcheck})$ time and in $O^*(\Delta^l \cdot \text{dcheck})$ time (Theorem 3), where dcheck denotes the running time of checking whether a subgraph has diameter most b, which can for example be solved in $O(\Delta \cdot n^2 \cdot \log(n))$ time by Dijkstra's algorithm. Finally, if the input graph is a tree, then we obtain that Diam-Team Formation is fixed-parameter tractable for parameter k (Theorem 4).

2. Hardness results

Throughout this section, we assume each edge in the input graph to have weight one; thus, we omit the introduction of the edge weight function w. We will see that our Diam-Team Formation problem is already hard in this setting. First, to get a feeling for the computational hardness of our Team Formation model we start with a simple observation which basically says that Diam-Team Formation with an unbounded number k of skills is at least as hard as the Set Cover problem, even on simple graph classes.

Set Cover
Input: A set family $\mathcal{F} = \{F_1, \dots, F_\alpha\}$ over a universe $U = \{1, 2, \dots, \beta\}$ and a non-negative integer h.
Question: Is there a *set cover* of size at most h, that is, a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ with $|\mathcal{F}'| \le h$ such that $\bigcup_{F \in \mathcal{F}'} F = U$?

Observation 1. *For edge weight one,* Diam-Team Formation *parameterized by the team size l generalizes* Set Cover *parameterized by the set cover size h, even on simple graph classes such as*

(1) *complete graphs,*
(2) *stars, and*
(3) *caterpillars with maximum vertex degree three.*

Proof. Given a Set Cover instance $(\mathcal{F}, U, h)$, we construct a Diam-Team Formation instance $(G = (V, E), T, S, l, b)$ for each of the settings as follows. We provide an example for each the constructions in Fig. 2.

(1) Define the skill set $T := U$, and for each set $F_i \in \mathcal{F}$, create one vertex v_i and define $S(v_i) := F_i$. Add an edge between each pair of vertices to obtain a complete graph. Finally, define the team size $l := h$ and let the cost budget b be an arbitrary integer at least one.
(2) Define the skill set $T := U$, and for each set $F_i \in \mathcal{F}$, create one vertex v_i and define $S(v_i) := F_i$. Add a special skill 0 to T, add a center vertex v_r to V, and define $S(v_r) := \{0\}$. Construct a star graph with center v_r by adding an edge between each vertex v_i and v_r, $1 \le i \le \alpha$. Finally, define the team size $l := h$, and let the cost budget b be an arbitrary integer at least two.
(3) Define the skill set $T := U \uplus \{x, y, z\}$, and for each set $F_i \in \mathcal{F}$, create two vertices u_i and v_i. Define $S(v_i) := F_i$ for all $1 \le i \le \alpha$, $S(u_1) := \{x\}$ and $S(u_\alpha) := \{y\}$, and $S(u_i) := \{z\}$ for all $1 < i < \alpha$. Add an edge between u_i and u_{i+1} for all $1 \le i < \alpha$ and edge between u_i and v_i for all $1 \le i \le \alpha$. Finally, define the team size l to be at least $h + \alpha$, and the cost budget $b := \alpha + 2$.

It is easy to verify that the constructed instances are yes-instances if and only if $(\mathcal{F}, U, h)$ is a yes-instance. □

2.1. Parameterized by the team size l

Given the close relation between Diam-Team Formation and Set Cover (Observation 1). We know that our problem remains intractable even if we require the team size to be small. Thus, we obtain the following hardness results.

Proposition 1. *Even when each edge has weight one, the following holds. (1)* Diam-Team Formation *parameterized by the team size l is* W[2]*-hard even if the budget b is one and the graph is complete. (2)* Diam-Team Formation *parameterized by the team size l is* W[2]*-hard even if the budget b is two and the graph is a star.*

We note that the budget in the proof of Statements (1)–(2) in Observation 1 as well as the team size in the proof of Statement (3) may have extremely large values that effectively do not upper-bound the communication costs or the team size. Since Set Cover is NP-complete and W[2]-complete when parameterized by h, in terms of minimizing the communication cost or team size, we have the following inapproximability result.

Corollary 1. *Unless all problems in* W[2] *are fixed-parameter tractable,* MinCost-Diam-Team Formation *is inapproximable in* FPT*-time for the parameter team size l, even on complete graphs or on stars.*

2.2. Parameterized by the maximum degree Δ

The reduction in Observation 1(3) shows that assuming the vertex degree to be a small constant alone does not lower the computational complexity of Diam-Team Formation.

Proposition 2. Diam-Team Formation *is* NP*-hard even on caterpillar graphs with maximum degree three.*

2.3. Parameterized by the number k of skills

Consequently, to identify tractable cases one should start with cases where Set Cover is tractable. A very well-motivated restriction for Diam-Team Formation is to assume that there are not too many skills to cover, that is, the number k of skills is (part of) the parameter. Our next result, however, shows that this assumption alone does not make the problem fixed-parameter tractable.

Theorem 1. Diam-Team Formation *parameterized by the number k of skills is* W[1]*-hard, even on graphs with maximum degree three and with each edge weight one, and when the team size is unrestricted.*

Proof. We give a parameterized reduction from the W[1]-complete problem Multicolored Clique parameterized by the clique size h to Team Formation parameterized by k on graphs of maximum degree three.

> Multicolored Clique
> **Input:** An undirected graph $G = (V, E)$, a non-negative integer $h \in \mathbb{N}$, and a vertex coloring $\phi\colon V \to \{1, 2, \ldots, h\}$.
> **Question:** Does G admit a colorful h-clique, that is, a size-h vertex subset $Q \subseteq V$ such that the vertices in Q are pairwise adjacent and have pairwise distinct colors?

Let (G, ϕ, h) be a Multicolored Clique instance. We construct in FPT-time an equivalent Diam-Team Formation instance $(G' = (V', E'), T, S, l, b)$. Without loss of generality, we assume that in G all edges are between vertices of different colors (according to ϕ) since they could be deleted without changing presence of a colorful h-clique. Let $n := |V|$, let $V = \{v_1, \ldots, v_n\}$ be an arbitrary ordering of the vertices, and let y be the smallest integer with $n \le 2^y$.

Construction. (Illustrated by an example in Fig. 3.) We construct the graph G' starting with an empty graph. First, we add to V' all vertices in V (as an independent set). We connect the vertices in V by the following three steps:

1. Attach to each v_i a path of length s (to be determined later) whose other endpoint is denoted w_i; i.e., the distance between v_i and w_i shall be s. We call this path, including v_i and w_i the *path of v_i*.
2. Make each w_i the root of a newly added complete binary tree of height y, i.e., with 2^y leaves. Arbitrarily pick any n of its leaves and assign them names $x_{i,1}, \ldots, x_{i,n}$. Thus, $x_{i,j}$ will be the jth leaf in the binary tree attached by a path of length s (via w_i) to vertex v_i. We call the binary tree with root w_i and leaves $x_{i,1}, \ldots, x_{i,n}$ the *binary tree of v_i*.
3. Finally, we encode the adjacency from G as follows: If v_i and v_j are vertices that are adjacent in G, which implies by our assumption that they have different colors, i.e., $\phi(v_i) \ne \phi(v_j)$, then add an edge between $x_{i,j}$ and $x_{j,i}$. Thus, a leaf in the binary tree of v_i (namely $x_{i,j}$) is now adjacent to a leaf in the binary tree of v_j (namely $x_{j,i}$). Note that the naming convention of leaves prevents using each leaf for more than one adjacency.

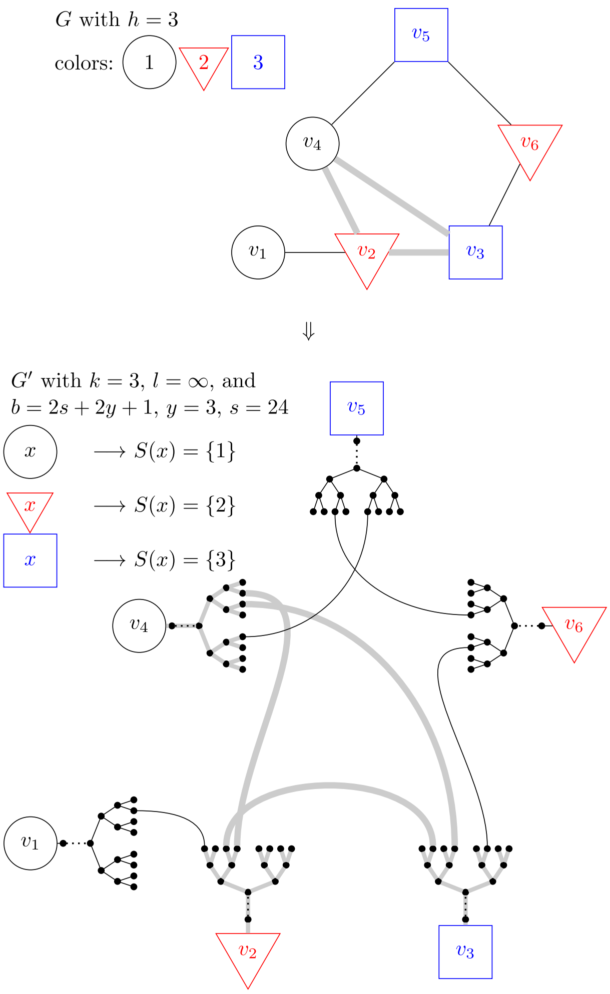

Fig. 3. An example for the reduction from Diam-Team Formation to Multicolored Clique as used in Theorem 1. Dummy vertices illustrated by small black circles represent people without any skill. Dotted lines represent paths consisting of s dummy vertices. The subgraph induced by the gray edges corresponds to the respective solutions.

This completes the construction of the graph G'. The graph can be constructed in FPT-time; indeed, it can be computed in polynomial time. Its maximum degree is three. Observe that if v_i and v_j are adjacent in G, then they have distance at most $2s + 2y + 1$ in G'. We set $s := 4n$ (with the intention of having vertices in two binary trees with adjacent leaves be at distance at most $4n$).

To complete the construction define the skill set $T := \{1, 2, \ldots, h\}$ and the skill function $S : V \to 2^T$ such that $S(v) := \{\phi(v)\}$ for all vertex $v \in V \subseteq V'$, i.e., for all vertices of the input graph G the skill equals the color according to ϕ, and $S(v) = \emptyset$ for all further vertices of $V' \setminus V$. The budget b (the diameter) is set to $b := 2s + 2y + 1$. Finally, the team size l is set to $|V'|$, i.e., the team size is effectively unbounded. We return instance $(G' = (V', E'), T, S, l, b)$.

Clearly this construction can be performed in polynomial time, and note that the parameter value number k of skills equals the parameter clique size h of the given Multicolored Clique instance. Let us check that (G, ϕ, h) and (G', T, S, l, b) are indeed equivalent.

Correctness. Assume that the constructed instance is a yes-instance and contains a team X of diameter at most c. Clearly, X must contain at least one vertex v with $S_v \supseteq \{p\}$ for each $p \in T = \{1, \ldots, k\}$. Since only vertices in V have nonempty skill sets and each vertex has at most one skill, there must be k different vertices, say $p_1, \ldots, p_k \in V$, with $S(p_i) = \{i\}$. We claim that these k vertices form a clique in G. Consider any two vertices p_i and $p_{i'}$. These vertices must be at distance at most $b = 2s + 2y + 1$ in $G'[X]$, i.e., in the graph induced by the team. Notice that since $p_i \in V$ the vertices reachable within distance at most $s + y$ from p_i are exactly those in the path and the binary tree of p_i; the same holds for $p_{i'}$. In particular, the leaves of the respective binary trees are at distance exactly $s + y$. Thus, the only way for having p_i and $p_{i'}$ at distance at most $2s + 2y + 1 = 2(s + y) + 1$ is a direct edge between leaves of their binary trees. This, however, directly implies adjacency of p_i and $p_{i'}$ in G, by construction. Thus, $p_1, \ldots, p_k$ indeed form a k-clique in G. Since $S(p_i) = \{i\}$ for

$i \in \{1, \ldots, k\}$ we directly get that the clique is also colorful and, hence, that (G, ϕ, h) is a yes-instance for MULTICOLORED CLIQUE.

For the converse, assume that the given instance (G, ϕ, h) of MULTICOLORED CLIQUE is a yes-instance and let $p_1, \ldots, p_h$ be the vertices of a colorful h-clique in G. We show how to find a team X of diameter at most b and covering all h skills in G'. Concretely, we claim that a feasible team can be obtained by taking the vertices $p_1, \ldots, p_h$ as well as their attached paths and their binary trees. Clearly, the vertices $p_1, \ldots, p_h$ hold the necessary skills and the main thing to check is the diameter. We prove that *all vertices in* X are at pairwise distance at most b in $G'[X]$ by proving the following: (1) Each two vertices on the paths of p_i and p_j, respectively, are at distance at most b. (2) Each two vertices in binary trees belonging to p_i and p_j are at a *smaller distance of at most* s. (3) Each vertex in a binary tree and each vertex on an attached path are at distance at most b.

1. For each two vertices u and v on the paths of two vertices p_i and p_j we can observe that both u and v are on all paths connecting p_i and p_j. Thus, the distance of u and v is upper-bounded by the distance of p_i and p_j. Since p_i and p_j are adjacent in G, the paths to the corresponding adjacent leaves in their binary tress in G' are both of length $s + y$. This implies that p_i and p_j are at distance at most $2 \cdot s + 2 \cdot (s + y) = b$.
2. Consider now any two vertices u and v in binary trees belonging to some p_i and p_j. We know that the two binary trees are connected by an edge. Thus, the graph induced just by the two trees is connected and has at most $4n$ vertices. Therefore, the distance of u and v is at most $4n = s$.
3. Finally, let u be a vertex in the binary tree of p_i and v a vertex on the path of p_j. We know from the previous point that the distance from u to the root of the binary tree of p_j is at most s. Clearly, from that root to v the distance is upper-bounded by the path length s, bounding the total distance by $2s \leq b$.

We have checked that all pairs of vertices in X are at distance at most b in $G'[X]$. Thus, the output instance (G', T, S, l, b) is indeed a yes-instance for DIAM-TEAM FORMATION, as claimed. This completes the proof. □

2.4. Parameterized by the team size l and the size of skills k

We know from Proposition 1 that DIAM-TEAM FORMATION is W[2]-hard for the parameter team size l and from Theorem 1 that DIAM-TEAM FORMATION is W[1]-hard for the parameter number k of skills (in the latter case even if the maximum degree Δ is a small constant). This invokes the question whether our problem becomes tractable for the combined parameter $l + k$, that is, for cases where both the team size and the number of skills are small. It turns out that DIAM-TEAM FORMATION remains intractable for the parameter $l + k$, even for a constant budget. We will see later (Theorem 3 in Section 3) that our problem becomes tractable when both values, the maximum vertex degree Δ and the cost budget b, are small.

Proposition 3. DIAM-TEAM FORMATION *parameterized by the combined parameter* $l + k$ *is* W[1]*-hard, even if the cost budget is one.*

Proof. We provide a parameterized reduction from MULTICOLORED CLIQUE parameterized by the clique size h (the formal definition of MULTICOLORED CLIQUE is given in the proof of Theorem 1). Let (G, ϕ, h) be an instance of MULTICOLORED CLIQUE; w.l.o.g. there are no edges $\{u, v\}$ with $\phi(u) = \phi(v)$. For $i \in \{1, \ldots, h\}$ define $V_i := \phi^{-1}(i)$, i.e., the set of all vertices of color i in G.

We use the same graph G as the graph G' of our DIAM-TEAM FORMATION problem and define the skill set $T := \{1, 2, \ldots, h\}$ to be the set of all the colors from ϕ. We set the skill of each vertex $v \in V(G')$ to $S(v) := \phi(v)$. Finally, we set the diameter budget $b := 1$ and the team size bound l to an arbitrary value at least h.

This completes the construction. It is easy to see that the whole construction can be done in linear time and thus, in FPT-time. One can also verify that (G, h) is a yes instance for MULTICOLORED CLIQUE if and only if there is a team that cover all the skills from T and has diameter $b = 1$, that is, (G, T, S, l, b) is a yes instance for DIAM-TEAM FORMATION. □

Observe that in the proofs of Theorem 1 and Proposition 3 we have made no use of the upper bound on the team size. Any team with all $k = h$ skills and diameter at most one was proven to lead directly to a h-clique. Thus, the minimum team size is strongly inapproximable in the sense that even finding any feasible team respecting the cost budget is W[1]-hard with respect to the size k of the skills.

Corollary 2. *Unless all problems in* W[1] *are fixed-parameter tractable,* MINTEAMSIZE-DIAM-TEAM FORMATION *is inapproximable in* FPT*-time for the parameter number k of skills, even on graphs either with maximum degree three or with cost budget one.*

Finally, we show that the W[1]-hardness for the combined parameter $l + k$ still holds when we require that the graph induced by the team is only two-vertex-connected (resp. two-edge-connected) instead of requiring a small diameter, that is, requiring robustness instead of low communication costs.

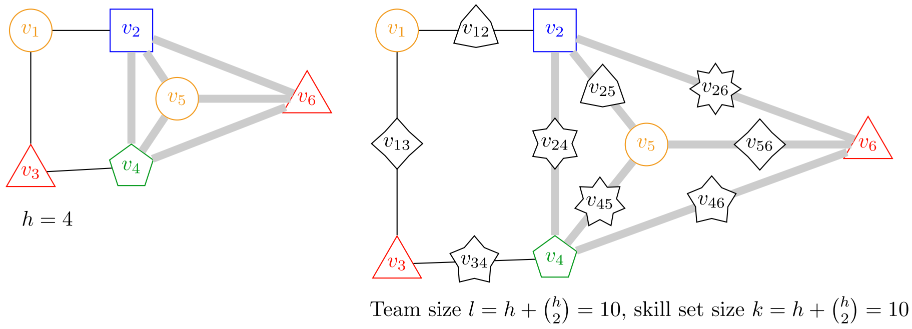

Fig. 4. An example showing a reduction from MULTICOLORED CLIQUE to TEAM FORMATION where the goal is to form a team such that the induced subgraph is two-connected. The left figure shows an instance of MULTICOLORED CLIQUE with six vertices and four colors. The graph induced by the gray edges corresponds to a colorful 4-clique. The right figure shows an instance constructed for TEAM FORMATION, where the skill set consists of exactly the four colors and all possible pairs of two colors. The skill of each vertex is directly encoded by the color.

Theorem 2. *Finding a team of size at most l, covering all k skills, such that the subgraph induced by the team is two-connected is* W[1]*-hard with respect to the combined parameter parameters $l + k$.*

Proof. We provide a parameterized reduction from MULTICOLORED CLIQUE parameterized by the clique size h to our problem parameterized by $l + k$ where l denotes the team size and k the number of skills. An example of how the construction works is shown in Fig. 4. Given a MULTICOLORED CLIQUE instance (G, ϕ, h), construct a graph G' by subdividing all edges. Use vertex sets $V_1, \ldots, V_h$ and let $V_{(i,j)}$ contain the subdividing vertices of (former) edges between V_i and V_j, $i < j$. Define the skill set $T := \{1, \ldots, h\} \cup \{(i, j) \mid 1 \le i < j \le h\}$. Assign the skills according to membership in sets V_i and $V_{(i,j)}$, i.e., a vertex v has skill $x \in T$ if and only if it is contained in set V_x. Finally, set the team size l to $h + \binom{h}{2}$. Note that the parameter value $l + k$ is upper-bounded by $O(h^2)$. This completes the construction which can clearly be done in linear time and in FPT-time.

Taking a colorful h-clique in G along with the subdividing vertices yields the required team in G' such that the subgraph induced by the team is two-connected. For the converse, the key point is that a team of size at most l must contain exactly one vertex for each skill from T. Then, however, since for all $(i, j) \in T$, vertices in $V_{(i,j)}$ have degree two, we must also include their two neighbors in V_i and V_j in the team; otherwise the team would no be two-connected. Thus, the vertices selected in $V_1 \cup \cdots \cup V_h$ must form a colorful h-clique in G. $\square$

3. Tractability results

In contrast to our hardness results, which all hold even for unit weights, we identify a number of tractable cases with arbitrary positive integer weights.

3.1. Parameterized by the maximum vertex degree Δ, combined with the team size l or the budget b

The first case models the situation where each potential team member is connected only to few others in the social network (that is, the maximum vertex degree Δ) and the budget (that is, the diameter b) or the desired team size l are small.

Theorem 3. DIAM-TEAM FORMATION *can be solved in $O^*(\Delta^l \cdot dcheck)$ time and in $O^*(\Delta^{\Delta^b} \cdot dcheck)$ time where Δ is the maximum vertex degree of the input graph, b is the communication cost budget (the diameter), l is the team size, and checking whether the diameter of a subgraph is at most b takes dcheck time.*

Proof. Let $(G = (V, E), w, T, S, l, b)$ be our DIAM-TEAM FORMATION instance. Without loss of generality, we assume that the team $V' \subseteq V$ which we search for induces a connected subgraph. Given that each vertex has at most Δ neighbors, we build a search tree algorithm that branches into selecting one of the Δ neighbors of a potential team member (vertex) adding it to our partial solution (team). Since the team can have at most l members, the depth of our search tree is upper-bounded by l. In each node of the search tree we need to check whether the subgraph induced by the partial solution has diameter at most b (regarding the edge weight function w); this check runs in polynomial time.

Applying Dijkstra's algorithm (see Cormen et al. [3] for the description), we can solve the all-pairs shortest paths problem in $O(\Delta \cdot n^2 \log(n))$ time, where n denotes the number of vertices and Δ denotes the maximum vertex degree. Our approach is to guess by iterating through each vertex $v \in V$ which will be in the team V' and then, in each iteration, perform the following procedure. Initialize $V' \leftarrow \{v\}$ and $u \leftarrow v$. Note that u denotes the newly added vertex in each branching step. Repeat the subsequent procedure until either all skills are covered by V' or $|V'| = l$: Branch into $\Delta - 1$ possibilities of adding one of u's neighbors to the team V' such that the diameter cost of $G[V']$ does not exceed b.

The search tree has depth at most l since in each step the size of V' is increased by one and $|V'| \le l$. Also, the number of branching possibilities is bounded by Δ. Thus, the whole algorithm takes $O(n \cdot \Delta^l \cdot \text{dcheck})$ where dcheck is the running time of checking whether a graph has diameter at most b; this check can be done in polynomial time.

As for the budget cost b (the diameter), since the subgraph induced by a team with team size l have maximum vertex degree Δ and diameter b, this subgraph can have at most Δ^b vertices, that is, $l \le \Delta^b$. Thus, we can conclude that our problem can be solved in $O(\Delta^{\Delta^b} \cdot \text{dcheck})$ time. □

3.2. Parameterized by the number k of skills

Our second tractable case models situations where the team members are organized in a hierarchical tree structure.

Theorem 4. *If the input social network is a tree, then* Diam-Team Formation *can be solved in $O(2^k \cdot n \cdot b^2 \cdot B_k)$ time, where k denotes the number of skills, n denotes the number of individuals in the network, b denotes the target diameter, and B_k denotes the kth Bell number.*

Proof. We describe a dynamic programming algorithm to solve Diam-Team Formation on trees. Let $I = (G = (V, E), w, T, S, l, b)$ be an instance of Diam-Team Formation with the input graph G being a tree. The basic idea is to store for each vertex $v \in V$ of the tree and each subset $T' \subseteq T$ of skills whether T' can be covered within the subtree rooted at v. To this end, we assume that $G = (V, E)$ is an arbitrarily rooted tree and denote the subtree rooted at $v \in V$ by subtree(v). We denote the set of children of each vertex $v \in V$ by children(v).

We define the dynamic programming table A as follows. For each subset $T' \subseteq T$ of skills, each vertex $v \in V$, each cost budget b', $b' \in \{0, 1, \dots, b\}$, and each depth bound z, $z \in \{0, 1, \dots, b\}$, the entry $A(T', v, b', z)$ stores the size of a smallest team $V' \subseteq V$ which fulfills the following requirements:

(a) V' covers T'.
(b) V' consists of vertex v and vertices only from subtree(v).
(c) The subgraph induced by V' is a tree with diameter at most b' and depth at most z. (That is, the largest weight of a shortest path between two arbitrary vertices of the tree $G[V']$ is at most b' and the largest weight of a shortest path between v and an arbitrary vertex $v \in V'$ is at most z.)

It is easy to see that there is a yes-instance if and only if $\min_{v \in V} A(T, v, b, b) \le l$.

We fill the table entries following the tree from the leaves to the root. We initialize the entries concerning the set $V_L \subseteq V$ of leaves of the tree as follows.

$$\forall T' \subseteq T;\, v \in V_L;\, b' \in \{0, \dots, b\};\, z \in \{0, \dots, b\}:$$
$$A(T', v, b', z) = \begin{cases} 1 & \text{if } T' \subseteq S_v \\ \infty & \text{otherwise} \end{cases}$$

Now, we consider some non-leaf v of the tree. The key question is which subtree rooted at some child of v contributes to the team. Clearly, in a *smallest* team,, each of such subtrees must cover at least some skill uniquely. Observe that there are at most B_k partitions of T' where B_k is the kth Bell number. That is, there are at most B_k possibilities of having at most k subtrees, each of which is rooted at some child of v and contributes to some smallest team covering T'. The idea is to consider for each part of the partition only the "cheapest" subtree covering it while fulfilling the diameter and depth requirements. Another crucial observation is that the diameter of the subtree rooted at $v \in V$ is the maximum of

(1) the largest diameter of all subtree(v'), $v' \in$ children(v), and
(2) the length of a longest path in subtree(v) containing v.

To calculate the value in (2), we need to know the length z_1 of a longest path from v to a leaf of the subtree(v') rooted at a child v' of v, and the length z_2 of a longest path from v to a leaf of the subtree(v'') rooted at a child v'' of v with $v'' \neq v'$.

Using these ideas, updating the entries bottom up works as follows. To handle the diameter costs that come from two different subtrees in subtree(v), we fix a partition of T' and the part of skills to be covered by the child v' of v such that subtree(v') has the largest depth.

$$\forall T' \subseteq T;\, v \in V;\, b', z \in \{0, \dots, b\}:\; A(T', v, b', z) =$$
$$\min_{\substack{1 \le i' \le k' \le k \\ T' = S_v \uplus T'_1 \uplus T'_2 \uplus \cdots \uplus T'_{k'}}} \text{cheapestCover}(T'_1, \dots, T'_{k'}, i', v, b', z),$$

where "cheapestCover()" denotes the size of a smallest team covering T' in the following way.

(i) Each disjoint subset T'_i of skills is covered by the vertices of the subtree rooted at one child of v.
(ii) The team that covers $T'_{i'}$ induces a subtree with the largest depth.
(iii) The overall team (including v and covering T') has diameter at most b', and depth at most z.

It can be computed as follows.

$$\text{cheapestCover}(T'_1, \dots, T'_{k'}, i', v, b', z) = 1 + \min_{\substack{z_2 \le z_1 \le z \\ z_1 + z_2 \le b'}} \Bigg\{ \min_{\substack{v' \in \text{children}(v) \\ z_1 \ge w(\{v, v'\})}} A(T'_{i'}, v', b', z_1 - w(\{v, v'\})) + \sum_{\substack{1 \le i \le k' \\ i \ne i'}} \min_{\substack{v'' \in \text{children}(v) \\ z_2 \ge w(\{v, v''\})}} A(T'_i, v'', b', z_2 - w(v, v'')) \Bigg\}$$

For the correctness of our algorithm, if $\min_{v \in V} A(T, v, b, b) \le l$, then there is indeed a set $V' \subseteq V$ with at most l vertices such that $\bigcup_{v \in V'} S(v) = T$ and $\text{Diam}(G[V']) \le b$, which can be constructed by standard backtracking of our dynamic programming algorithm. However, it is not obvious that our algorithm considers all possible solutions, since it assumes that each part $T'_i, 1 \le i \le k'$ of the partition $T' = S_v \uplus T'_1 \uplus T'_2 \uplus \cdots \uplus T'_{k'}$ is covered by a distinct cheapest subtree. To see that this is no real restriction, consider some fixed partition with $T' = S_v \uplus T'_1 \uplus T'_2 \uplus \cdots \uplus T'_{i_1} \uplus T'_{i_2} \uplus \cdots \uplus T'_{k'}$. Of course, it may happen that the cheapest subtrees for T'_{i_1} and T'_{i_2} are identical. In this case, the value of "cheapestCover()" might be higher than the size of the corresponding team and one might think that a smaller team may not be identified. However, as this also means that $T'_{i''} := T'_{i_1} \uplus T'_{i_2}$ can be covered within the same subtree, the team will be found using some partition with $T' = S_v \uplus T'_1 \uplus T'_2 \uplus \cdots \uplus T'_{i''} \uplus \cdots \uplus T'_{k'}$, that is, replacing the two parts T'_{i_1} and T'_{i_2} with $T'_{i''}$. Summarizing, for every team there is always a partition of T' such that no two parts are covered within the same cheapest subtree.

Finally, the size of the two tables is upper-bounded by some function in $O(2^k \cdot n \cdot b^2)$. The initialization phase takes $O(2^k \cdot n \cdot b^2)$ time. The update phase takes $O(2^k \cdot n \cdot b^2 \cdot B_k)$ time. □

Finally we conjecture that the fixed-parameter tractability result from Theorem 4 can be extended to hold even for the combined parameter "number k of skill" and "treewidth t". However, showing this requires extensive technical details going beyond the scope of this work.

4. Conclusion and outlook

We performed a parameterized complexity analysis of the team formation problem on social networks, focusing on the case where the cost is measured by the diameter of the social network induced by the team. We have shown that the problem is extremely intractable and many intractability results lead to inapproximability, even allowing FPT-time. However, we could also identify tractable cases when the maximum vertex degree is small and either the team size or the diameter bound is small.

There are several possibilities for future work. First, besides the diameter model and the minimum spanning tree model, there are several other models studied in the literature (see the related work section). It would be interesting to also study the parameterized complexity of other popular models.

Second, we model the skill ability of each individual to be either one or zero. It would be interesting to also study the case where people only have graded skills.

Third, we have shown FPT result for the number k of skills when the given social network is a tree. It would be interesting to know whether this holds for the combined parameter k and the tree width.

Acknowledgements

This work was started at the research retreat of the TU Berlin Algorithms and Computational Complexity group held in April 2014. Robert Bredereck was partially supported by the DFG, project PAWS (NI 369/10), and from September 2016 to September 2017 on postdoctoral leave at the University of Oxford (GB), supported by the DFG fellowship BR 5207/2. We thank the anonymous reviewers of *AAIM 2016* and of *Theoretical Computer Science* for improving our work.

References

[1] A. Anagnostopoulos, L. Becchetti, C. Castillo, A. Gionis, S. Leonardi, Online team formation in social networks, in: Proc. 21st International Conference on World Wide Web, WWW '12, ACM, 2012, pp. 839–848.
[2] E.M. Arkin, R. Hassin, Minimum-diameter covering problems, Networks 36 (3) (2000) 147–155.
[3] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, 2nd edition, MIT Press, 2001.
[4] M. Cygan, F.V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, S. Saurabh, Parameterized Algorithms, Springer, 2015.
[5] S. Datta, A. Majumder, K. Naidu, Capacitated team formation problem on social networks, in: Proc. 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, ACM, 2012, pp. 1005–1013.

[6] R.G. Downey, M.R. Fellows, Fundamentals of Parameterized Complexity, Springer, 2013.

[7] S.E. Dreyfus, R.A. Wagner, The Steiner problem in graphs, Networks 1 (3) (1972) 195–207.

[8] J. Flum, M. Grohe, Parameterized Complexity Theory, Springer, 2006.

[9] A. Gajewar, A.D. Sarma, Multi-skill collaborative teams based on densest subgraphs, in: Proc. 12th SIAM International Conference on Data Mining, SDM '12, SIAM/Omnipress, 2012, pp. 165–176.

[10] J.H. Gutiérrez, C.A. Astudillo, P. Ballesteros-Pérez, D. Mora-Melià, A. Candia-Véjar, The multiple team formation problem using sociometry, Comput. Oper. Res. 75 (2016) 150–162.

[11] E. Halperin, R. Krauthgamer, Polylogarithmic inapproximability, in: Proc. 35th Annual ACM Symposium on Theory of Computing, STOC '03, ACM, 2003, pp. 585–594.

[12] T. Lappas, K. Liu, E. Terzi, Finding a team of experts in social networks, in: Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09, ACM, 2009, pp. 467–476.

[13] T. Lappas, K. Liu, E. Terzi, A survey of algorithms and systems for expert location in social networks, in: Social Network Data Analytics, Springer, 2011, pp. 215–241.

[14] C. Li, M. Shan, Team formation for generalized tasks in expertise social networks, in: Proc. 2nd IEEE International Conference on Social Computing, SocialCom '10, IEEE, 2010, pp. 9–16.

[15] J. Nederlof, Fast polynomial-space algorithms using Möbius inversion: Improving on Steiner tree and related problems, in: Proc. 36th International Colloquium on Automata, Languages and Programming, ICALP '09, in: LNCS, vol. 5555, Springer, 2009, pp. 713–725.

[16] R. Niedermeier, Invitation to Fixed-Parameter Algorithms, Oxford University Press, 2006.

[17] J. Park, S. Lee, Keyword search in relational databases, Knowl. Inform. Syst. 26 (2) (2010) 175–193.