SIAM J. Discrete Math.
Vol. 29, No. 2, pp. 888–914

NETWORK-BASED VERTEX DISSOLUTION*

RENÉ VAN BEVERN†, ROBERT BREDERECK†, JIEHUA CHEN†, VINCENT FROESE†, ROLF NIEDERMEIER†, AND GERHARD J. WOEGINGER‡

Abstract. We introduce a graph-theoretic vertex dissolution model that applies to a number of redistribution scenarios, such as gerrymandering in political districting or work balancing in an online situation. The central aspect of our model is the deletion of certain vertices and the redistribution of their load to neighboring vertices in a completely balanced way. We investigate how the underlying graph structure, the knowledge of which vertices should be deleted, and the relation between old and new vertex loads influence the computational complexity of the underlying graph problems. Our results establish a clear borderline between tractable and intractable cases.

Key words. computational complexity analysis, combinatorial algorithms, economization, election control, flow networks, matching, NP-hard problems, political districting, redistribution scenarios

AMS subject classifications. 68Q25, 68R10

DOI. 10.1137/140978880

1. Introduction. Motivated by applications in areas like political redistricting, economization, and distributed systems, we introduce a class of graph modification problems that we call *network-based vertex dissolution*. We are given an undirected graph where each vertex carries a load consisting of discrete entities (e. g., voters, tasks, data). These loads are *balanced*: all vertices carry the same load. Now a certain number of vertices has to be *dissolved*; that is, they are to be deleted from the graph, and their loads are to be redistributed among their neighbors so that afterwards all loads are balanced again.

In fact, our vertex dissolution problem comes in two flavors: Dissolution and Biased Dissolution. Dissolution is the basic version described in the preceding paragraph. Biased Dissolution is a variant that is motivated by gerrymandering in the context of political districting. It is centered around a two-party scenario with two types, A and B, of discrete entities. The goal is to find a redistribution that maximizes the number of vertices in which the A-entities form a majority. See Section 2 for a formal definition of these models.

*Received by the editors July 23, 2014; accepted for publication (in revised form) March 9, 2015; published electronically May 21, 2015. An extended abstract of this paper appeared under the title "Network-Based Dissolution" in *Proceedings of the 39th International Symposium on Mathematical Foundations of Computer Science* (MFCS '14), Lecture Notes in Comput. Sci. 8635, Springer, Berlin, 2014, pp. 69–80.
http://www.siam.org/journals/sidma/29-2/97888.html
†Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, 10587 Berlin, Germany (rene.vanbevern@tu-berlin.de, robert.bredereck@tu-berlin.de, jiehua.chen@tu-berlin.de, vincent.froese@tu-berlin.de, rolf.niedermeier@tu-berlin.de). The first author's research was supported by the DFG, project DAPA (NI 369/12). The second author's research was supported by the DFG, project PAWS (NI 369/10). The third author's research was supported by the Studienstiftung des Deutschen Volkes. The fourth author's research was supported by the DFG, project DAMM (NI 369/13).
‡Department of Mathematics and Computer Science, TU Eindhoven, 5600 MB Eindhoven, The Netherlands (gwoegi@win.tue.nl). This author's research, while visiting TU Berlin, was supported by a Humboldt Research Award of the Alexander von Humboldt Foundation, Bonn, Germany.

Our focus lies on analyzing the computational complexity of network-based vertex dissolution problems and on getting a good understanding of polynomial-time solvable and NP-hard cases.

1.1. Three application scenarios. We discuss three example scenarios in detail. The first two examples relate more to Biased Dissolution, while the third example is closer to Dissolution.

Our first example comes from *political districting*, the process of setting electoral districts. Let us consider a situation with two political parties (A and B) and an electorate of voters that each support either A or B. The electorate is currently divided into n districts, each consisting of precisely s individual voters. A district is won by the party that receives the majority of votes in the district (for simplification, assume that ties are resolved in favor of B). The local government performs an electoral reform that reduces the number of districts, and the local governor (from party A) is in charge of the redistricting. His goal is, of course, to let party A win as many districts as possible while dissolving some districts and moving their voters to adjacent districts. All resulting new districts should have equal size s_{new} (where $s_{\text{new}} > s$). In the network-based vertex dissolution model, the districts and their neighborhoods are represented by an undirected graph, where each vertex represents a district and each edge indicates that two districts are adjacent.

Our second example concerns storage updates in parallel or distributed systems. Consider a distributed storage array consisting of n storage nodes, each having a capacity of s storage units, of which some units are empty. As the prices on cheap hard disk space are rapidly decreasing, the operators want to upgrade the storage capacity of some nodes and to deactivate other nodes for saving energy and cost. As their distributed storage concept takes full advantage only if all nodes have equal capacity, they want to upgrade all (nondeactivated) nodes to the same capacity s_{new} and move capacities from deactivated nodes to nondeactivated neighboring nodes. In the resulting system, every nondeactivated node should only use half of its storage units. In the network-based vertex dissolution model, storage nodes and their neighborhoods are represented by an undirected graph, where each vertex represents a storage node and each edge indicates that two nodes are neighbored in the array. The storage units are modeled by our two-party variant, where empty units are represented by party A and used units are represented by party B. Finally, one asks for redistribution such that A-entities form a majority for every vertex.

Our third and last example concerns *economization* in a fairly general form. Consider a company with n employees, each producing s units of a desirable good during a month; for concreteness, let us say that each employee proves s theorems per month. Now, due to the increasing support of automatic theorem provers, each employee is able to prove s_{new} theorems per month ($s_{\text{new}} > s$). Hence, without lowering the total number of proved theorems per month, some employees may be moved to a special task force for improving automatic theorem provers: this will secure the company's future competitiveness in proving theorems, without decreasing the overall theorem output. By company regulations, all theorem-proving employees have to be treated equally and should have identical workloads. In the network-based vertex dissolution model, employees correspond to vertices, and edges indicate that the corresponding employees are comparable in qualification and research interest. Employees in the special task force are dissolved and disappear from the scene of action; their workload is to be taken over by employees who are comparable in qualification and research interests.

1.2. Related work. We are not aware of any previous work on our network-based vertex dissolution problem. Our main inspiration came from the area of political districting, in particular from gerrymandering [19, 26, 27], and from supervised regionalization methods [11]. Of course, graph-theoretic models have been employed earlier for (political) districting; for instance, Mehrota, Johnson, and Nemhauser [22] draw a connection to graph partitioning, and Duque [10] and Maravalle and Simeone [21] use graphs to model geographic information in the regionalization problem. These models are tailored towards very specific applications and are mainly used for the purpose of developing efficient heuristic algorithms, often relying on mathematical programming techniques. The computational hardness of districting problems has been known for quite some time [2].

Also related to our problem is constructive (or destructive) control by partitioning voters, which was introduced by Bartholdi III, Tovey, and Trick [4]. In this scenario, a chair wants to make some candidate become a winner (or a loser) by partitioning the set of voters and applying some multistage voting protocol. The crucial difference from our model is that there are no restrictions on the possible voter set partitions. The computational complexity of control by voter partitioning has been investigated for many voting rules (Faliszewski and Rothe [13] give an overview).

1.3. Remark on nomenclature. For ease of presentation, throughout the paper we will adopt a political districting point of view on network-based vertex dissolution: the words districts and vertices are used interchangeably, and the entities in districts are referred to as voters or supporters.

1.4. Contributions and organization of this paper. We propose two simple computational problems, DISSOLUTION and BIASED DISSOLUTION, to make the model for network-based vertex dissolution (Section 2) concrete. In the main body of our work, we provide a variety of computational tractability and intractability results for both problems. We investigate relations of our new modeling to established models like matchings and flow networks. Furthermore, we analyze how the structure of the underlying graphs or how an in-advance fixing of which vertices should be dissolved influences the computational complexity (mainly in terms of polynomial-time solvability versus NP-hard cases).

In Section 3, using flow networks, we show that BIASED DISSOLUTION is polynomial-time solvable if the set of districts to be dissolved and the set of districts to be won are both specified as part of the input. Furthermore, we show how our new model generalizes established models such as partitioning graphs into stars and perfect matchings.

Section 4 presents a complexity dichotomy for both DISSOLUTION and BIASED DISSOLUTION with respect to the old district size s and the increase Δ_s in district size (that is, the difference between the new and the old district size). DISSOLUTION is polynomial-time solvable for $s = \Delta_s$, and BIASED DISSOLUTION is polynomial-time solvable for $s = \Delta_s = 1$; all other cases are NP-complete.

Section 5 analyzes the complexity of DISSOLUTION and BIASED DISSOLUTION for various specially structured graphs, including planar graphs (NP-complete), cliques (polynomial-time solvable), and graphs of bounded treewidth (linear-time solvable if s and Δ_s are constant).

2. Formal setting. We start by introducing notation and formal definitions of the technical terms that we use throughout the paper.

2.1. Graphs. Unless stated otherwise, we consider simple, undirected graphs $G = (V, E)$, where V is a set of n vertices and $E \subseteq \binom{V}{2}$ is a set of m edges. We use $\binom{V}{2}$ to denote the family of all size-two subsets of V. For a given graph G, we denote by $V(G)$ the set of vertices and by $E(G)$ the set of edges of G. For a subset $V' \subseteq V(G)$ of vertices and a subset $E' \subseteq (E(G) \cap \binom{V'}{2})$ of edges, the graph $G' = (V', E')$ is called a *subgraph* of G. We also say G contains G'. For a vertex subset $V' \subseteq V$, the *induced subgraph* $G[V']$ of G is defined as $G[V'] := (V', E \cap \binom{V'}{2})$.

A *path* is a graph $P = (V, E)$ with vertex set $V = \{v_1, v_2, \ldots, v_n\}$ and edge set $E = \{\{v_1, v_2\}, \{v_2, v_3\}, \ldots, \{v_{n-1}, v_n\}\}$. The vertices v_1 and v_n are the *endpoints* of the path P. We say two vertices v and v' in a graph G are *connected* if G contains a path with the endpoints v and v'. A graph is *connected* if every two vertices are connected. The *connected components* of a graph are its maximal connected subgraphs. For a vertex $v \in V$, we denote by $N(v) := \{u \in V \mid \{u, v\} \in E\}$ the *(open) neighborhood* of v, that is, all vertices that are connected to v by an edge.

A *t-star* is a graph $K_{1,t} = (V, E)$ with vertex set $V = \{v_1, v_2, \ldots, v_{t+1}\}$ and edge set $E = \{\{v_1, v_i\} \mid 2 \leq i \leq t + 1\}$. The vertex v_1 is called the *center* of the star. A *t-star partition* of G is a partition $\{V_1, \ldots, V_{n/(t+1)}\}$ of the vertex set V into subsets of size $t + 1$ such that each $G[V_i]$ contains a t-star as a subgraph. Note that a 1-star partition is a *perfect matching*.

2.2. Networks and flows. A flow network I^* consists of a directed graph $G^* = (V^*, E^*)$, where V^* is the set of nodes and E^* is a set of arcs, an arc capacity function $c^* \colon E^* \to \mathbb{R}^+$, and two distinguished nodes $\sigma, \tau \in V^*$ called the *source* and the *target* of the network. An arc is an ordered pair of nodes from V^*, and $\mathbb{R}^+$ is the set of nonnegative real numbers.

A (σ, τ)-flow $f \colon E^* \to \mathbb{R}^+$ is an arc value function with $f(u, v) \geq 0$ for all $(u, v) \in E^*$ such that

1. the *capacity constraint* is fulfilled, i.e.,

$$\forall (u, v) \in E^* : f(u, v) \leq c(u, v), \text{ and}$$

2. the *conservation property* is satisfied, i.e.,

$$\forall u \in V^* \setminus \{\sigma, \tau\} : \sum_{(u,v) \in E^*} f(u, v) = \sum_{(v,u) \in E^*} f(v, u).$$

We call f *integer* if all its values are integers. The *value* of f is $\sum_{(\sigma,u) \in E^*} f(\sigma, u)$. Note that we distinguish between vertices in graphs and nodes in flow networks.

2.3. Dissolutions. Let G be an undirected graph representing n districts. Let $s, \Delta_s \in \mathbb{N}^+$ be the *district size* and *district size increase*, respectively, where $\mathbb{N}^+$ is the set of nonnegative integer numbers. For a subset $V' \subseteq V(G)$ of districts, let

$$Z(V', G) := \{(x, y) \mid x \in V' \wedge y \in V(G) \setminus V' \wedge \{x, y\} \in E(G)\}$$

be the set of district pairs consisting of a district from V' and a neighbor that is not from V'. The central notion for our studies is that of a *dissolution*, which basically describes a valid movement of voters from dissolved districts into nondissolved districts. The formal definition is as follows.

DEFINITION 2.1 (dissolution). *Let G be an undirected graph, let $D \subset V(G)$ be a subset of* districts to dissolve*, and let $z \colon Z(D, G) \to \{0, \ldots, s\}$ be a function that describes how many voters shall be* moved *from one district to its nondissolved neighbors. Then, (D, z) is called an* (s, Δ_s)-dissolution*for G if*

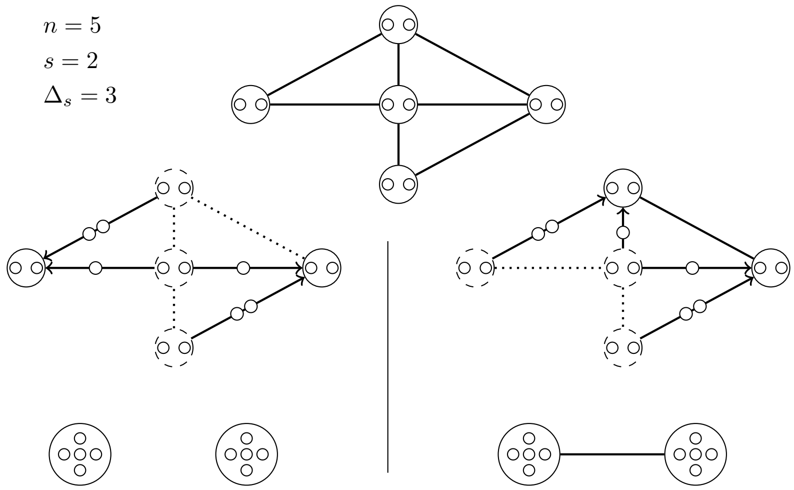

FIG. 1. *An illustration of two* $(2,3)$*-dissolutions. Small circles represent the voters. The graph on the top shows a neighborhood graph of five districts, each district consisting of two voters. The task is to dissolve three districts such that each remaining district contains five voters. The graphs in the middle show two possible realizations of dissolutions. The graphs on the bottom show the two corresponding outcomes. The arrows in the "middle graphs" point from the districts to be dissolved to the "target districts," and the white circle labels on the arrows represent the voters moved along the arrows.*

(a) *no voter remains in any dissolved district:*

$$\forall v' \in D : \sum_{(v',v) \in Z(D,G)} z(v',v) = s, \text{ and}$$

(b) *the size of all remaining (nondissolved) districts increases by* Δ_s*:*

$$\forall v \in V \setminus D : \sum_{(v',v) \in Z(D,G)} z(v',v) = \Delta_s.$$

Throughout this work, we use

- $s_{\text{new}} := s + \Delta_s$ *to denote the new district size,*
- $d := |D| = |V(G)| \cdot \Delta_s / s_{\text{new}}$ *to denote the number of dissolved districts, and*
- $r := |V(G)| - d$ *to denote the number of remaining, nondissolved districts.*

We write *dissolution* instead of (s, Δ_s)-dissolution when s and Δ_s are clear from the context. By definition, a dissolution ensures that the numbers of voters moving between districts fulfill the given constraints on the district sizes, that is, the size of each remaining district increases by Δ_s. Figure 1 gives an example illustrating two possible $(2,3)$-dissolutions for a 5-vertex graph.

Motivated from social choice application scenarios, we additionally assume that each voter supports one of two parties, A and B. We then search for a dissolution such that the number of remaining districts won by party A is maximized. Here, a district is won by the party that is supported by a strict majority of the voters inside the district. This yields the notion of a *biased dissolution*, which is defined as follows.

DEFINITION 2.2 (biased dissolution). *Let* G *be an undirected graph, and let* $\alpha \colon V(G) \to \{0, \ldots, s\}$ *be an* A-supporter distribution, *where* $\alpha(v)$ *denotes the number of A-supporters in district* $v \in V$. *Let* (D, z) *be an* (s, Δ_s)*-dissolution for* G*; that is,*

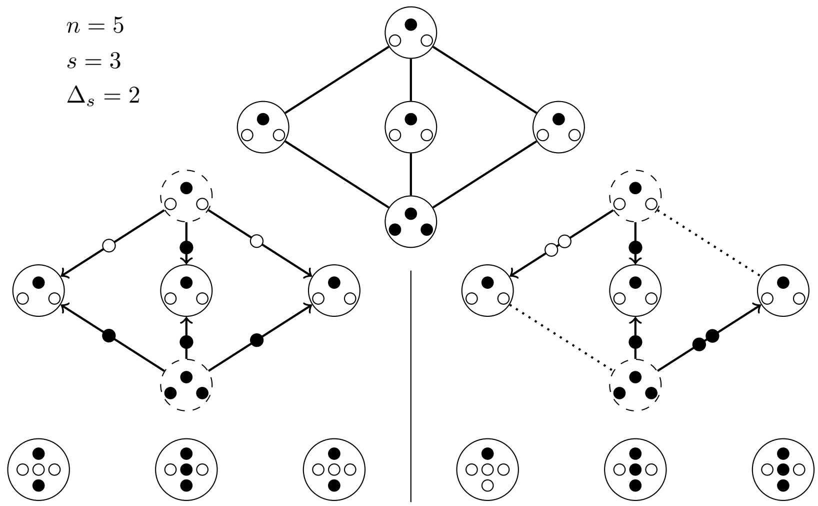

FIG. 2. *An illustration of a* 1*-biased* (3, 2)*-dissolution (left) and a* 2*-biased* (3, 2)*-dissolution (right). Black circles represent A-supporters, while white circles represent B-supporters. The graph on the top shows a neighborhood graph of five districts, each district consisting of three voters. The task is to dissolve two districts such that each remaining district contains five voters. The graphs in the middle show two possible realizations of dissolutions. The graphs on the bottom show the two corresponding outcomes. The arrows point from the districts to be dissolved to the "target districts" and the black/white circle labels on the arrows indicate which kind of voters are moved along the arrows.*

properties (a) *and* (b) *of Definition* 2.1 *are fulfilled. Let $r_\alpha \in \mathbb{N}$ be the minimum number of districts that party A shall win after the dissolution and $z_\alpha \colon Z(D,G) \to \{0,\dots,s\}$ be an* A-supporter movement, *where $z_\alpha(v',v)$ denotes the number of A-supporters moving from district v' to district v. Finally, let $R_\alpha \subseteq V(G) \setminus D$ be a size-r_α subset of districts. Then, $(D, z, z_\alpha, R_\alpha)$ is called an r_α*-biased (s, Δ_s)-dissolution *for (G, α) if*

(a) *a district does not receive more A-supporters from a dissolved district than the total number of voters it receives from that district:*

$$\forall (v',v) \in Z(D,G) : z_\alpha(v',v) \leq z(v',v),$$

(b) *no A-supporters remain in any dissolved district:*

$$\forall v' \in D : \sum_{(v',v) \in Z(D,G)} z_\alpha(v',v) = \alpha(v'), \text{ and}$$

(c) *each district in R_α has a strict majority of A-supporters:*

$$\forall v \in R_\alpha : \alpha(v) + \sum_{(v',v) \in Z(D,G)} z_\alpha(v',v) > \frac{s + \Delta_s}{2}.$$

We also say that a district wins *if it has a strict majority of A-supporters, and* loses *otherwise.*

Figure 2 shows two biased dissolutions: one with $r_\alpha = 1$ and the other one with $r_\alpha = 2$. We are now ready to formally state the definitions of the two computational dissolution problems (in their decision versions) that we discuss in this work.

DISSOLUTION
Input: An undirected graph $G = (V, E)$ and positive integers s and Δ_s.
Question: Is there an (s, Δ_s)-dissolution for G?

BIASED DISSOLUTION
Input: An undirected graph $G = (V, E)$, positive integers s, Δ_s, r_α, and an A-supporter distribution $\alpha : V \to \{0, \ldots, s\}$.
Question: Is there an r_α-biased (s, Δ_s)-dissolution for (G, α)?

Note that DISSOLUTION is equivalent to BIASED DISSOLUTION with $r_\alpha = 0$. As we will see later, DISSOLUTION and BIASED DISSOLUTION are NP-complete in general. In this work, we additionally look into special cases and investigate what the causes of intractability may be.

3. Relations to established models. In this section, we identify relations of our model to established graph concepts like matchings, flow networks, or star partitions. This will also be useful for proofs in later sections. In Section 3.1, we show that DISSOLUTION and BIASED DISSOLUTION instances where the roles of the districts are already known can be translated into flow networks. In Section 3.2 we show that dissolutions generalize star partitions and perfect matchings.

3.1. Flow networks. Sometimes the districts that are to be dissolved and the districts that are to be won are not arbitrary but already determined beforehand. For this case we show that BIASED DISSOLUTION can be modeled as a network flow problem, which can be solved in polynomial time.

THEOREM 3.1. *Let $(G, s, \Delta_s, r_\alpha, \alpha)$ be a* BIASED DISSOLUTION *instance, and let $D, R_\alpha \subset V(G)$ be two disjoint fixed subsets of districts. The problem of deciding whether (G, α) admits an r_α-biased (s, Δ_s)-dissolution $(D, z, z_\alpha, R_\alpha)$ can be reduced in linear time to a maximum flow problem with $2|V(G)|+2$ nodes, $2|V(G)|+3|E(G)|$ arcs, and maximum arc capacity $\max(s, \Delta_s)$.*

Proof. Denote the set of remaining districts by R, that is, $R := V(G) \setminus D$. With $R_\alpha \subseteq R$ given beforehand, we can compute how many A-supporters a district $v \in R_\alpha$ needs from its neighboring dissolved districts in order to win after the dissolution. With D also given beforehand, we can use a flow network with two nodes corresponding to each district to compute an r_α-biased (s, Δ_s)-dissolution.

To this end, we first remove all edges between two vertices from D or between two vertices from R since only edges between D and R may be taken into account for the dissolution. Doing this, we obtain a bipartite neighborhood graph with the two disjoint vertex sets $D = \{d_1, \ldots, d_k\}$ and $R = \{r_1, \ldots, r_{n-k}\}$. Second, we observe that, in order to let a district $r \in R$ win after the dissolution, r needs at least $\max\{0, \lceil (s_{\text{new}} + 1)/2 \rceil - \alpha(r)\}$ additional A-supporters. Hence, we compute a "demand" function $\kappa : R \to \{0, \ldots, \lceil (s_{\text{new}} + 1)/2 \rceil\}$ for each nondissolved district r by $\kappa(r) := \max\{0, \lceil (s_{\text{new}} + 1)/2 \rceil - \alpha(r)\}$ if $r \in R_\alpha$ and $\kappa(r) := 0$ otherwise.

The idea now is to construct a flow network which models the movement of A-supporters that are necessary for a district in R_α to win and models the movement of the remaining voters necessary to end up with district size s_{new} separately. More precisely, we split each $d \in D$ into a node d^A, modeling the supply of A-supporters from d, and into a node d^B, modeling the supply of the B-supporters from d. Similarly, we split each $r \in R$ into a node r^A, modeling the demand for A-supporters for r, and into a node r^{AB}, modeling the remaining demand for voters, that is, voters to finally end up with district size s_{new}. Now, following the constraints given by the neighborhood graph, A-supporters may move in order to satisfy some demand for A-

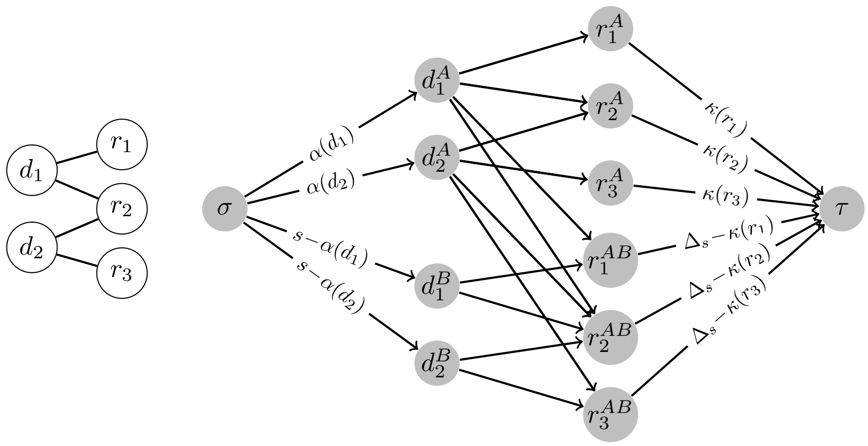

FIG. 3. *An illustration of the network flow construction. Left: the graph G of an instance of* BIASED DISSOLUTION *with $D = \{d_1, d_2\}$. Right: the corresponding network flow. The capacities of the arcs from dissolved nodes to nondissolved nodes are omitted for the sake of brevity.*

supporters or in order to satisfy the general demand on voters. Clearly, B-supporters may only move in order to satisfy the general demand on voters.

Formally, we construct a flow network $I^* = (G^* = (V^*, E^*), c^*, \sigma, \tau)$ for our input instance $(G, s, \Delta_s, r_\alpha, \alpha)$ as follows (see Figure 3 for an illustration). The node set V^* in G^* consists of a source node σ, a target node τ, two nodes d_i^A and d_i^B for each district $d_i \in D$, and two nodes r_i^A and r_i^{AB} for each district $r_i \in R$. In total, V^* has $2|V| + 2$ nodes.

The arcs in E^* are divided into three layers:

1. Arcs from the source node to all nodes corresponding to dissolved districts: For each dissolved district $d_i \in D$, add to E^* two arcs (σ, d_i^A) and (σ, d_i^B) with capacities $c^*(\sigma, d_i^A) = \alpha(d_i)$ and $c^*(\sigma, d_i^B) = s - \alpha(d_i)$.
2. Arcs from the nodes corresponding to dissolved districts to nodes corresponding to nondissolved neighbored districts: For each dissolved district $d_i \in D$ and for each $r_j \in N(d_i)$ of its nondissolved neighbors, add to E^* three arcs (d_i^A, r_j^A), (d_i, r_j^{AB}), and (d_i^B, r_j^{AB}) with capacities $c^*(d_i^A, r_j^A) = c^*(d_i^A, r_j^{AB}) = \alpha(d_i)$ and $c^*(d_i^B, r_j^{AB}) = s - \alpha(d_i)$.
3. Arcs from all nondissolved nodes to the target node: For each nondissolved district $r_j \in R$, add to E^* two arcs (r_j^A, τ) and (r_j^{AB}, τ) with capacities $c^*(r_j^A, \tau) = \kappa(r_j)$ and $c^*(r_j^{AB}, \tau) = \Delta_s - \kappa(r_j)$.

This completes the description of the flow network construction.

We show that there is an r_α-biased (s, Δ_s)-dissolution $(D, z, z_\alpha, R_\alpha)$ for (G, α) if and only if the constructed flow network I^* has a (σ, τ)-flow of value $s \cdot |D|$.

For the "only if" part, suppose that there is a dissolution $(D, z, z_\alpha, R_\alpha)$ for (G, α). Construct a (σ, τ)-flow $f : E^* \to \mathbb{R}$ by defining $f(\sigma, d_i^A) := c^*(\sigma, d_i^A)$ and $f(\sigma, d_i^B) := c^*(\sigma, d_i^B)$, where d_i is a dissolved district. Then, define $f(r_j^A, \tau) := c^*(r_j^A, \tau)$ and $f(r_j^{AB}, \tau) := c^*(r_j^{AB}, \tau)$, where r_j is a nondissolved district. Note that by definition of the network, this means that $f(\sigma, d_i^A) = \alpha(d_i)$ and $f(\sigma, d_i^B) = s - \alpha(d_i)$, where d_i is a dissolved district, as well as that $f(r_j^A, \tau) = \kappa(r_j)$ and $f(r_j^{AB}, \tau) = \Delta_s - \kappa(r_j)$, where r_j is a nondissolved district. It remains to define the values of f for the arcs in layer 2. For each dissolved district $d_i \in D$ and for each $r_j \in N(d_i)$ of its nondissolved

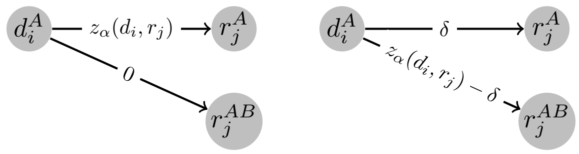

FIG. 4. *Two cases of setting the flow values for arcs outgoing from d_i^A nodes. Left: The sum of the flow through arcs to r_j^A is at most $\kappa(r_j) - z_\alpha(d_i, r_j)$. Right: The sum of the flow through arcs to r_j^A is $\kappa(r_j) - \delta$, where $0 \le \delta < z_\alpha(d_i, r_j)$.*

neighbors, define $f(d_i^B, r_j^{AB}) := z(d_i, r_j) - z_\alpha(d_i, r_j)$. To also define the flow values for arcs outgoing from a node $d_i^A, 1 \le i \le k$, we use the following procedure, where we remember in each step the total amount $u(r_j^A)$ of flow going into r_j^A. We initialize u by setting $u(r_j^A) := 0$ for each $r_j \in R$. Now, process all pairs (d_i, r_j) with $d_i \in D$ and $r_j \in N(d_i)$ in an arbitrary ordering, where the following two cases may occur (illustrated in Figure 4).

Case 1. If $u(r_j^A) + z_\alpha(d_i, r_j) \le \kappa(r_j)$, then increase $u(r_j^A)$ by $z_\alpha(d_i, r_j)$ and set $f(d_i^A, r_j^A) := z_\alpha(d_i, r_j)$ and $f(d_i^A, r_j^{AB}) := 0$.

Case 2. If $u(r_j^A) + \delta = \kappa(r_j)$ for some nonnegative integer $\delta < z_\alpha(d_i, r_j)$, then increase $u(r_j^A)$ by δ and set $f(d_i^A, r_j^A) := \delta$ and $f(d_i^A, r_j^{AB}) := z_\alpha(d_i, r_j) - \delta$.

Now, observe that by our definition of f the flow value is $\sum_{(s,x)\in E^*} f(s, x) = s\cdot|D|$. It remains to show that f is valid. By our definition of f, the flow value of each arc does not exceed its capacity. For each $d_i \in D$, the conservation property for the nodes d_i^A and d_i^B is fulfilled by property (a) of Definition 2.1 and property (b) of Definition 2.2 of the biased dissolution. For each $r_j \in R$, the conservation property for the node r_j^A is fulfilled by our definition of f (which ensures that the ingoing flow is at most $\kappa(r_j)$) and by property (c) of Definition 2.2 of the biased dissolution (which ensures that the ingoing flow is at least $\kappa(r_j)$). The conservation property for the node r_j^{AB} is fulfilled by properties (a) and (c) of Definition 2.2 of the biased dissolution (and the way we defined f).

For the "if" part, suppose that f is a (σ, τ)-flow for I^* with value $s \cdot |D|$. Let $z_\alpha : Z(D, G) \to \{0, \dots, s\}$ and $z : Z(D, G) \to \{0, \dots, s\}$ be two functions with values $z_\alpha(d_i, r_j) := f(d_i^A, r_j^A) + f(d_i^A, r_j^{AB})$ and $z(d_i, r_j) := z_\alpha(d_i, r_j) + f(d_i^B, r_j^{AB})$. One can verify that $(D, z, z_\alpha, R_\alpha)$ is an r_α-biased (s, Δ_s)-dissolution for (G, α) as follows: Property (a) of Definition 2.1 is fulfilled since the total flow going over d_i^A and d_i^B has value exactly s. Property (b) of Definition 2.1 is fulfilled since the total flow going over r_j^A and r_j^{AB} has value exactly Δ_s. Property (a) of Definition 2.2 is fulfilled by our definition of z and z_α. Property (b) of Definition 2.2 is fulfilled since the total flow going over d_i^A is $\alpha(d_i)$. Property (c) of Definition 2.2 is fulfilled since the total flow going over r_j^A is $\kappa(r_j)$. ∎

The following corollary shows that plain dissolutions can be modeled using a much simpler flow network in comparison to biased dissolutions. In particular, all capacity values are either s or Δ_s—a property which will be important in later proofs.

COROLLARY 3.2. *Let G be a graph, and let $D \subset V(G)$ be a subset of vertices. If there exists an (s, Δ_s)-dissolution (D, z) for G, then it can be found by computing the maximum flow in a network with $|V(G)| + 2$ nodes and $|E(G)| + 2|V(G)|$ arcs where all capacities are either s or Δ_s.*

Proof. If the districts to dissolve are known and we search only for a dissolution (in other words, $r_\alpha = 0$), then the flow network used to compute a dissolution from

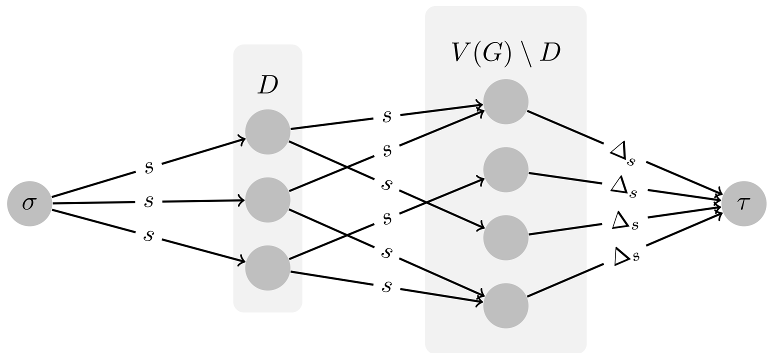

FIG. 5. *Flow network for* DISSOLUTION *when the set D of districts to dissolve is known.*

the proof of Theorem 3.1 basically reduces to a much simpler flow network. For this case, we can assume that $R_\alpha = \emptyset$ and $\alpha(v) = 0$ for all $v \in V(G)$, remove all arcs with capacity zero, and finally also remove nodes without a directed path from the source or to the sink.

Doing this, we end up with the following: We have a source σ and a sink τ and two additional layers of nodes: the first layer contains one node for each vertex from D, and the second layer contains one node for each vertex from $V(G) \setminus D$. There is an arc from the source σ to each node in the first layer with capacity s and an arc from each node in the second layer to the sink τ with capacity Δ_s. Finally, there is an arc of capacity s from a node in the first layer to a node in the second layer if and only if the corresponding vertices in the neighborhood graph G are adjacent. See Figure 5 for an illustration. ∎

Contrasting the polynomial-time solvability when D and R_α are known, we obtain NP-completeness for BIASED DISSOLUTION once at least one of the two sets D and R_α is unknown. DISSOLUTION is the special case of BIASED DISSOLUTION with $R_\alpha = \emptyset$, and we will see in Section 4.1 that DISSOLUTION is NP-complete for the case of $s \neq \Delta_s$ (Theorem 4.3). This means that BIASED DISSOLUTION is NP-hard even if the set R_α is known to be empty. For the case that only the set D of dissolved districts is given beforehand, it remains to decide how many A-supporters are moved to a certain nondissolved district. We will see in Section 4.2, however, that in the hardness construction for Theorem 4.6 it is already fixed which districts are to be dissolved. This means that BIASED DISSOLUTION is NP-hard even if the set D of dissolved districts is given beforehand. Summarizing, BIASED DISSOLUTION is NP-hard even if either the set D of districts to dissolve or the set R_α of districts to win is known.

With the help of the above flow network construction from Theorem 3.1, we can design an exact algorithm for BIASED DISSOLUTION that runs in polynomial time when the number of districts is a constant. Since the degree of the polynomial does not depend on the number of districts, this means fixed-parameter tractability with respect to the number of districts (see [9, 14, 24] for details on fixed-parameter tractability).

COROLLARY 3.3. *Any instance (G, s, Δ_s, α) of* BIASED DISSOLUTION *can be solved in $O(3^{|V(G)|} \cdot (\max(s, \Delta_s) \cdot |V(G)| \cdot |E(G)| + |V(G)|^3))$ time.*

Proof. Since each district will either be dissolved, won, or lost, there are at most $3^{|V(G)|}$ different ways to fix the roles of all $|V(G)|$ districts. In each case, we can construct a flow network with $O(|V(G)|)$ nodes and maximum capacity $\max(s, \Delta_s)$ in $O(\max(s, \Delta_s) \cdot |V(G)| \cdot |E(G)|)$ time and compute the maximum flow (Theorem 3.1) to solve Biased Dissolution. Hence, by using an $O(|V(G)|^3)$-time maximum flow algorithm we solve Biased Dissolution in $O(3^{|V(G)|}(\max(s, \Delta_s) \cdot |V(G)| \cdot |E(G)| + |V(G)|^3))$ time. ◻

3.2. Relation to star partition and matching. In this subsection, we analyze how dissolutions relate to star partitioning and matching. If $\Delta_s = 1$, then each nondissolved district receives exactly one additional voter from one of its neighboring districts. Each dissolved district has to move exactly one voter to each of s neighboring districts. Hence, it is easy to see that a graph has an $(s, 1)$-dissolution if and only if it has an s-star partition.

Using the flow construction from Corollary 3.2, we can even show that this equivalence to star partition generalizes to the case that s is an integer multiple of any Δ_s.

Proposition 3.4. *There exists a $(t \cdot \Delta_s, \Delta_s)$-dissolution for an undirected graph G if and only if G has a t-star partition.*

Proof. If $G = (V, E)$ can be partitioned into t-stars, then it is easy to see that there is a $(t \cdot \Delta_s, \Delta_s)$-dissolution for G: Let $C = \{c_1, \ldots, c_d\} \subset V$ be the set of t-star centers, and let $L_i \subset V$, $1 \leq i \leq d$, be the set of leaves of the ith star. Define function $z : Z(C, G) \to \{0, \ldots, t \cdot \Delta_s\}$ so that, for all $(c_i, l) \in Z(C, G)$, $z(c_i, l) := \Delta_s$ if $l \in L_i$ and $z(c_i, l) := 0$ otherwise. Obviously, (C, z) is a $(t \cdot \Delta_s, \Delta_s)$-dissolution for G.

Now, let (D, z) be a $(t \cdot \Delta_s, \Delta_s)$-dissolution for G. We show that G can be partitioned into t-stars with D being the t-star centers. To this end, consider the network flow constructed in Corollary 3.2 and modify the network as follows. For each arc, divide its capacity by Δ_s. Clearly, if there is a flow with value $|D| \cdot t \cdot \Delta_s = |V \setminus D| \cdot \Delta_s$, then the modified network has a flow with value $|D| \cdot t = |V \setminus D|$. As all capacities are integers, there exists a maximum flow f such that for each arc a it holds that $f(a)$ is integer [1]. Hence, a partition of G into t-stars consists of one star for each $v_i \in D$ such that v_i is the star center connected to its leaves $L_i = \{u \mid f(v_i, u) = 1\}$. ◻

Since a t-star partition with $t = 1$ is a perfect matching, we obtain the following corollary.

Corollary 3.5. *There exists an (s,s)-dissolution for an undirected graph G if and only if G has a perfect matching.*

4. Complexity dichotomy with respect to district sizes. In this section, we study the computational complexity of Dissolution and Biased Dissolution with respect to the relation of the district size s to the district size increase Δ_s. We show that Dissolution is polynomial-time solvable if $s = \Delta_s$, and NP-complete otherwise (Theorem 4.3). Biased Dissolution is polynomial-time solvable if $s = \Delta_s = 1$, and NP-complete otherwise (Theorem 4.6).

We start by showing a useful structural observation for dissolutions. More precisely, we observe a symmetry concerning the district size s and the district size increase Δ_s in the sense that exchanging their values yields an equivalent instance of Dissolution. Intuitively, the idea behind the following lemma is that the roles of dissolved and nondissolved districts in a given (s, Δ_s)-dissolution can in fact be exchanged by "reversing" the movement of voters to obtain a (Δ_s, s)-dissolution.

Lemma 4.1. *There exists an (s, Δ_s)-dissolution for an undirected graph G if and only if there exists a (Δ_s, s)-dissolution for G.*

Proof. Let (D, z) be an (s, Δ_s)-dissolution for G. We show that $(V(G) \setminus D, z')$, where z' is defined by $z'(x, y) := z(y, x)$ is a (Δ_s, s)-dissolution for G: First, observe that the domain of z' is correct:

$$\begin{aligned} Z(V(G) \setminus D, G) &= \{(x,y) \mid x \in V(G) \setminus D \wedge y \in V(G) \setminus (V(G) \setminus D) \\ &\qquad \wedge \{x,y\} \in E(G))\} \\ &= \{(x,y) \mid x \in V(G) \setminus D \wedge y \in D \wedge \{x,y\} \in E(G))\}. \end{aligned}$$

Second, observe that $(V(G) \setminus D, z')$ fulfills all properties of Definition 2.1: Property (a) is fulfilled for $(V(G) \setminus D, z')$ if and only if property (b) is fulfilled for (D, z), and property (b) is fulfilled for $(V(G) \setminus D, z')$ if and only if property (a) is fulfilled for (D, z). ∎

4.1. Dissolution. In this subsection, we show a P versus NP dichotomy of Dissolution with respect to the district size s and the size increase Δ_s. Observe that from Corollary 3.5 it directly follows that Dissolution is polynomial-time solvable if $s = \Delta_s$.

If $s \neq \Delta_s$, then Dissolution is NP-complete. We can use a result from number theory to encode instances of the NP-complete Exact Cover by t-Sets problem into instances of Dissolution.

Exact Cover by t-Sets
Input: A finite set X and a collection $\mathcal{C}$ of subsets of X of size t.
Question: Is there a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ that partitions X, that is, each element of X is contained in exactly one subset in $\mathcal{C}'$?

Now, let us briefly recall some elementary number theory.

Lemma 4.2 (Bézout's identity). *Let a and b be two positive integers, and let g be their greatest common divisor. Then, there exist two integers x and y with $ax + by = g$.*

Moreover, x and y in Lemma 4.2 can be computed in polynomial time using the extended Euclidean algorithm [7, Section 31.2]. Indeed, we can infer from Lemma 4.2 that any two integers x' and y' with $x' = ix + jb/g$ and $y' = iy - ja/g$ for some $i, j \in \mathbb{Z}$ satisfy $ax' + by' = ig$. We will make use of this fact several times in the NP-hardness proof of the following theorem.

Theorem 4.3. *If $s = \Delta_s$, then* Dissolution *is solvable in $O(n^\omega)$ time (where ω is the matrix multiplication exponent); otherwise the problem is NP-complete.*

Proof. First, Corollary 3.5 says that there is an (s, s)-dissolution if and only if there is a perfect matching in G, which can be computed in $O(n^\omega)$ time with ω being the smallest exponent such that matrix multiplication can be computed in $O(n^\omega)$ time. Currently, the smallest known upper bound of ω is 2.3727 [28].

For the case $s \neq \Delta_s$, we show that Dissolution is NP-complete if $s > \Delta_s$. Due to Lemma 4.1, this also transfers to the cases where $s < \Delta_s$. First, given a Dissolution instance (G, s, Δ_s) and a function $z : Z(D, G) \to \{0, \ldots, s\}$ where $D \subset V(G)$, one can check in polynomial time whether (D, z) is an (s, Δ_s)-dissolution. Thus, Dissolution is in NP.

To show the NP-hardness result, we give a reduction from the NP-complete Exact Cover by t-Sets [16] for $t := (s + \Delta_s)/g > 2$, where $g := \gcd(s, \Delta_s) \leq \Delta_s$ is the greatest common divisor of s and Δ_s.

Given an Exact Cover by t-Sets instance $(X, \mathcal{C})$, we construct a Dissolution instance (G, s, Δ_s) with a neighborhood graph $G = (V, E)$ defined as follows: For each element $u \in X$, add a clique C_u of properly chosen size q to G, and let v_u denote an arbitrary fixed vertex in C_u. For each subset $S \in \mathcal{C}$, add a clique C_S of properly

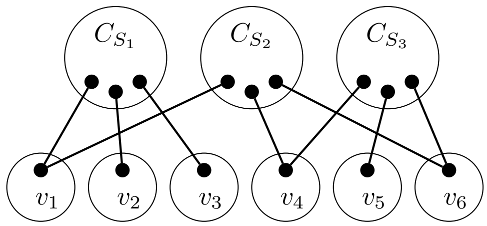

FIG. 6. *The constructed instance for $t = 3$.*

chosen size $r \geq t$ to G and connect each v_u for $u \in S$ to a unique vertex in C_S. Figure 6 shows an example of the constructed neighborhood graph for $t = 3$.

Next, we explain how to choose the values of q and r. We set $q = x_q + y_q$, where $x_q \geq 0$ and $y_q \geq 0$ are integers satisfying $x_q s - y_q \Delta_s = g$. Such integers exist by Lemma 4.2. The intuition behind this is as follows: Dissolving x_q districts in C_u and moving the voters to y_q districts in C_u creates an overflow of exactly g voters, who have to move out of C_u. Note that the only way to move voters into or out of C_u is via district v_u. Moreover, if the constructed instance (G, s, Δ_s) admits a dissolution, then exactly x_q districts in C_u are dissolved because dissolving more districts leads to an overflow of at least $g+s+\Delta_s > s$ voters, which is more than v_u can move, whereas dissolving fewer districts yields a demand of at least $s + \Delta_s - g > \Delta_s$ voters, which is more than v_u can receive. Thus, the district v_u must be dissolved since there is an overflow of g voters to move out of C_u, and this can only be done via district v_u.

The value of $r \geq t$ is chosen in such a way that, for each subset $S \in \mathcal{C}$ and each element $u \in S$, it is possible to move g voters from v_u to C_S (recall that v_u must be dissolved). In other words, we require C_S to be able to receive in total $t \cdot g = s + \Delta_s$ voters in at least t nondissolved districts. Thus, we set $r := x_r + y_r$, where $x_r \geq 0$ and $y_r \geq t$ are integers satisfying $x_r s - y_r \Delta_s = -(s + \Delta_s)$. Again, since $-(s + \Delta_s)$ is divisible by g, such integers exist by Lemma 4.2. It is thus possible to dissolve x_r districts in C_S, moving the voters to the remaining y_r districts in C_S such that we end up with a demand of $s + \Delta_s$ voters in C_S. Note that the only other possibility is to dissolve $x_r + 1$ districts in C_S in order to end up with a demand of zero voters. In this case, no voters of any other districts connected to C_S can move to C_S. By the construction of C_u, it is clear that it is also not possible to move any voters out of C_S because no v_u can receive voters in any dissolution. Thus, if the constructed instance (G, s, Δ_s) admits a dissolution, then either all or none of the districts v_u connected to some C_S move g voters to C_S.

We are now ready to show that G has a (s, Δ_s)-dissolution if and only if $(X, \mathcal{C})$ is a yes-instance of EXACT COVER BY t-SETS.

For the "only if" part, suppose that $(X, \mathcal{C})$ is a yes-instance, that is, there exists a partition $\mathcal{C}' \subseteq \mathcal{C}$ of X. We can thus dissolve x_q districts in each C_u (including v_u) and move the voters such that all y_q nondissolved districts receive exactly Δ_s voters. This is always possible since C_u is a clique. If we do so, then, by construction, g voters have to move out of each v_u. Since $\mathcal{C}'$ partitions X, each $u \in X$ is contained in exactly one subset $S \in \mathcal{C}'$. We can thus move the g voters from each v_u to C_S. Now, for each $S \in \mathcal{C}'$, we dissolve any x_r districts that are not adjacent to any v_u, and for the subsets in $\mathcal{C} \setminus \mathcal{C}'$, we simply dissolve $x_r + 1$ arbitrary districts in the corresponding cliques. As already discussed, each C_S with x_r dissolved districts receives $t \cdot g$ voters, and each C_S with x_r+1 dissolved districts receives no voter. Thus, this in fact yields an (s,Δ_s)-dissolution.

For the "if" part, assume that there exists an (s,Δ_s)-dissolution for G. As already discussed, every (s,Δ_s)-dissolution generates an overflow of g voters in each C_u that has to be moved over v_u to some district in C_S. Furthermore, each C_S either receives g voters from all its adjacent v_u or no voters at all. Therefore, the subsets S corresponding to cliques C_S that receive $t \cdot g$ voters form a partition of X, showing that $(X,\mathcal{C})$ is a yes-instance. $\square$

4.2. Biased dissolution. Because DISSOLUTION is a special case of BIASED DISSOLUTION, the NP-hardness results for $s \neq \Delta_s$ transfer to BIASED DISSOLUTION. It remains to see whether BIASED DISSOLUTION remains polynomial-time solvable when $s=\Delta_s$. Interestingly, this is true for $s=\Delta_s=1$, but BIASED DISSOLUTION turns NP-hard when $s=\Delta_s \geq 2$.

To analyze the structure of dissolutions, we introduce the concept of the "edge set used by a dissolution," which we will use in several proofs. Let (D,z) be a dissolution of a graph G. Let $E_z \subseteq E(G)$ contain all edges $\{x,y\}$ with $(x,y) \in Z(D,G)$ and $z(x,y)>0$. Then, we call E_z the *edge set used by* the dissolution (D,z).

The following lemma shows that finding an r_α-biased $(1,1)$-dissolution essentially corresponds to finding a maximum-weight perfect matching.

LEMMA 4.4. *Let $(G=(V,E)$, $s=1$, $\Delta_s=1, r_\alpha, \alpha)$ be a* BIASED DISSOLUTION *instance. There is an r_α-biased $(1,1)$-dissolution for (G,α) if and only if there is a perfect matching of weight at least r_α in (G,w) with $w(\{x,y\}):=1$ if $\alpha(x)=\alpha(y)=1$, and $w(\{x,y\}):=0$ otherwise.*

Proof. For the "only if" part, let (D,z,z_α,R_α) be an r_α-biased $(1,1)$-dissolution for (G,α). Then, the edge set $E_z \subseteq E$ used by (D,z,z_α,R_α) partitions G into 1-stars, or in other words, E_z is a perfect matching for G (see Proposition 3.4). Note that a nondissolved district can only win if it already contains an A-supporter and receives one additional A-supporter. By the construction of w, this implies that the weight of each edge that connects a winning district is 1 (i.e., for each $e \in E_z$ it holds that $e \cap R_\alpha \neq \emptyset$ if and only if $w(e)=1$). Since $|R_\alpha| \geq r_\alpha$, the perfect matching E_z has weight at least r_α.

For the "if" part, let $E' \subseteq E$ be a perfect matching of weight at least r_α. By the construction of w, E' contains at least r_α edges, each of which has weight 1. Then, we construct an r_α-biased $(1,1)$-dissolution (D,z,z_α,R_α) as follows. For each edge $\{x,y\} \in E'$, arbitrarily add one of its endpoints, say x, to D and set $z(x,y):=1$. Furthermore, if $\alpha(x)=1$, then set $z_\alpha(x,y):=1$. If $w(\{x,y\})=1$, meaning that the districts corresponding to x and y have an A-supporter each, then add y to R_α since y wins after the dissolution. Finally, $|R_\alpha| \geq r_\alpha$ since $|E'| \geq r_\alpha$. $\square$

As we have already seen from Corollary 3.5, the edge set used by a $(1,1)$-dissolution is a perfect matching. This is useful for finding a polynomial-time algorithm solving BIASED DISSOLUTION, exploiting that maximum-weight perfect matchings can be computed in polynomial time. Can we find similar useful characterizations for r_α-biased (s,s)-dissolutions for $s>1$?

Already for $(2,2)$-dissolutions, a characterization by the edge set used is not as compact as for $(1,1)$-dissolutions: The edge set used by a $(2,2)$-dissolution for some graph G corresponds to a partition of the graph into disjoint cycles of even length and disjoint paths on two vertices. For the case of r_α-biased $(2,2)$-dissolution, one would at least need some weights, and it is not clear how to find such a partition efficiently. However, by appropriately setting α and r_α, we can enforce that the edge set used by

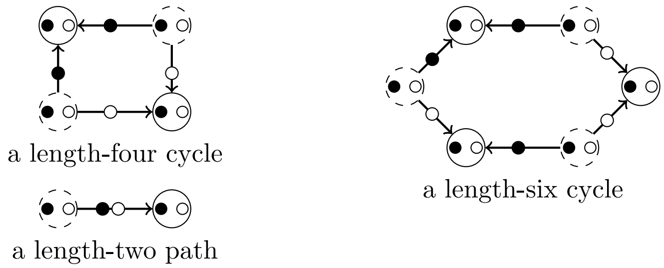

Fig. 7. *Graphs induced by edge sets used by an r_α-biased $(2,2)$-dissolution. In order to have a majority of A-supporters (black dots) in at least half of the new districts, each component must be a cycle of length divisible by four.*

any r_α-biased $(2,2)$-dissolution induce only cycles of lengths divisible by four: We let each district have one A-supporter and one B-supporter (i.e., $\alpha : V \to \{1\}$ for each district v) and let $r_\alpha := |V(G)|/4$. Doing this we end up with a restricted two-factor problem which was already studied in the literature [18].

L-Restricted Two Factor

Input: An undirected graph $G = (V, E)$.

Question: Is there a two-factor $E' \subseteq E$ such that the number of vertices in each connected component in (V, E') belongs to L?

A *two-factor* of a graph $G = (V, E)$ is a subset of edges $E' \subseteq E$ such that each vertex in the subgraph $G' := (V, E')$ has degree exactly two, that is, G' contains only disjoint cycles.

Lemma 4.5. *Let $G = (V, E)$ be an undirected graph with $4q$ vertices ($q \in \mathbb{N}$). Then, G has a two-factor E' whose cycle lengths are all multiples of four if and only if (G, α) admits a q-biased $(2,2)$-dissolution with $\alpha(v) = 1$ for all $v \in V$.*

Proof. For the "only if" part, let $E' \subseteq E$ be an edge subset such that each vertex in $G' := (V, E')$ has degree two and G' consists of disjoint cycles of lengths divisible by four. We now construct a q-biased $(2,2)$-dissolution $(D, z, z_\alpha, R_\alpha)$ for (G, α). To this end, we start with $D := \emptyset$, $R_\alpha := \emptyset$ and do the following for each cycle $c_1 c_2 \ldots c_{4l} c_1, l \geq 1$. For each number i with $1 \leq i \leq 2l$, add c_{2i} to D, and set $z(c_{2i}, c_{2i-1}) := z(c_{2i}, c_{(2i+1) \mod 4l}) := 1$. For each $1 \leq i \leq l$, we set

$$\begin{aligned} z_\alpha(c_{4i-2}, c_{4i-3}) &:= 1, \quad & z_\alpha(c_{4i-2}, c_{4i-1}) &:= 0, \\ z_\alpha(c_{4i}, c_{(4i+1) \mod 4l}) &:= 1, \quad & z_\alpha(c_{4i}, c_{4i-1}) &:= 0. \end{aligned}$$

Doing this, every fourth vertex in each cycle receives two additional A-supporters (see Figure 7 for an illustration of the corresponding dissolutions). It is easy to verify that $(D, z, z_\alpha, R_\alpha)$ is indeed a q-biased $(2,2)$-dissolution.

For the "if" part, let $(D, z, z_\alpha, R_\alpha)$ be a q-biased $(2,2)$-dissolution for (G, α). Furthermore, let E_z denote the edge set used by $(D, z, z_\alpha, R_\alpha)$. Each component C in $G[E_z]$ is either a path of length two or a cycle of even length and consists of exactly $|V(C)|/2$ dissolved and $|V(C)|/2$ nondissolved districts. Since each nondissolved district needs at least two A-supporters in order to win and only $|V(C)|/2$ A-supporters can be moved from the $|V(C)|/2$ dissolved districts, at most $|V(C)|/4$ districts can win. With $r_\alpha = q$, this implies that in total exactly q districts must win. This can only succeed if each component C is a cycle of length divisible by four (also see Figure 7 for an illustration). ∎

Now, we are ready to show that Biased Dissolution is NP-complete even for constant values of s and Δ_s, except if $s = \Delta_s = 1$, where it is solvable in polynomial time.

Theorem 4.6. Biased Dissolution *can be solved in* $O(n \cdot (m + n \log n))$ *time if* $s = \Delta_s = 1$*; otherwise it is NP-complete.*

Proof. For $s = \Delta_s = 1$, Biased Dissolution reduces to computing a maximum-weight perfect matching (see Lemma 4.4). This can be done in $O(n \cdot (m + n \log n))$ time [15].

It is easy to see that Biased Dissolution is in NP. Now, we show the NP-hardness for $s = \Delta_s \geq 2$. For $s = \Delta_s = 2$, observe that Lemma 4.5 implicitly provides a polynomial-time reduction from the graph problem L-Restricted Two Factor to Biased Dissolution with $L \subseteq \{3, \ldots, |V|\}$.

Two-factors of graphs are computable in polynomial time [12]. However, L-Restricted Two Factor is NP-hard if $(\{3, 4, \ldots, |V|\} \setminus L) \nsubseteq \{3, 4\}$ [18]. By Lemma 4.5, $(G = (V, E), L)$ with $|V| = 4q$ and $L = \{4, 8, \ldots, 4q\}$ is a yes-instance of L-Restricted Two Factor if and only if $(G, 2, 2, q, \alpha)$ with $\alpha(v) = 1$ for all $v \in V$ is a yes-instance of Biased Dissolution. Since $(\{3, 4, \ldots, |V|\} \setminus \{4, 8, \ldots, 4q\}) \nsubseteq \{3, 4\}$ for all $q > 1$, it follows that Biased Dissolution is NP-complete when $s = \Delta_s = 2$.

For $s = \Delta_s \geq 3$, we show NP-hardness by a polynomial-time reduction from the NP-complete Exact Cover by t-Sets for $t \geq 3$ (see the corresponding definition in Section 4.1). Given an Exact Cover by t-Sets instance $(X, \mathcal{C})$ with $|X| = t \cdot q$ elements and $r := |\mathcal{C}|$, we construct a Biased Dissolution instance $(G = (V, E), t, t, r_\alpha, \alpha)$.

To construct the graph G, we use the so-called *t-element gadget*. A t-element gadget consists of a t-star where each leaf has an additional degree-one neighbor. We call the degree-t vertex *center district*, the original star leaves *inner districts*, and the additional degree-one vertices *element districts*. A 3-element gadget is illustrated in Figure 8. Now, we add to the graph G the following:

- q t-element gadgets; we arbitrarily identify each element $x \in X$ with exactly one of the $(q \cdot t)$-*element districts* that is denoted as v_x in the following,
- for each subset $Y \in \mathcal{C}$ a *set district* v_Y, and
- $r - q$ *dummy districts*.

Then, we connect each set district v_Y with each element district $v_x, x \in Y$ and connect each dummy district with each set district. We set the number r_α of winning districts to $(t + 1) \cdot q$.

We now describe how many A-supporters each district contains (that is, the function α).

- The dummy district contains no A-supporters.
- Each set district contains exactly one A-supporter.
- For each t-element gadget, the center district contains no A-supporters, each inner district contains exactly two A-supporters, and each element district contains t A-supporters.

This concludes the construction which is illustrated for $t = 3$ in Figure 9.

Now, we show that $(X, \mathcal{C})$ is a yes-instance of Exact Cover by t-Sets if and only if the constructed Biased Dissolution instance $(G, t, t, (t + 1)q, \alpha)$ is a yes-instance.

For the "only if" part, let $\mathcal{C}' \subseteq \mathcal{C}$ be a subcollection such that each element of X is contained in exactly one subset of $\mathcal{C}'$. A $(t + 1)q$-biased (t, t)-dissolution can be constructed as follows. Dissolve each center district, and move one B-supporter to

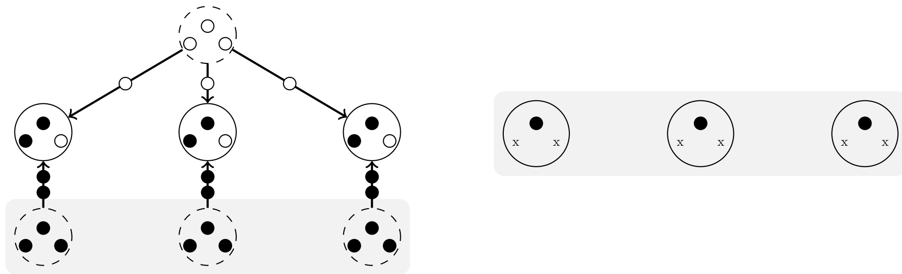

FIG. 8. *Left: A 3-element gadget. The only dissolution where A wins all districts requires dissolving the top district and moving exactly one B-supporter from the top district to each neighbor. Right: Gadget symbol in the construction.*

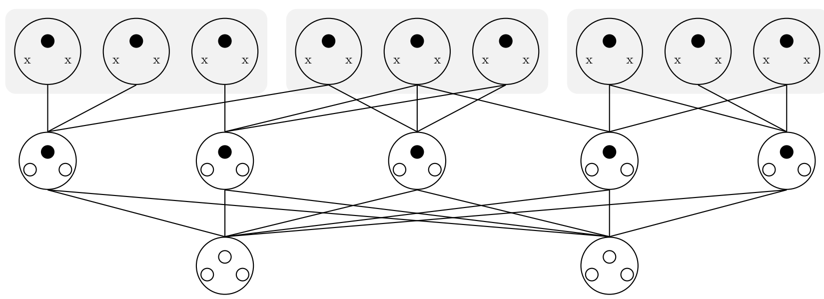

FIG. 9. *Illustration of the construction for $t = 3$, $r = 5$, and $q = 3$.*

each of its adjacent inner districts. Dissolve each element district, and move $(t-1)$ A-supporters to its uniquely determined adjacent inner district. For each element district $v_x, x \in X$, move the remaining A-supporter to the set district $v_Y, Y \in \mathcal{C}'$, with $x \in Y$. Since $\mathcal{C}'$ partitions X, v_Y is uniquely determined. The set R_α of winning districts consists of all inner districts and the set districts corresponding to the sets in $\mathcal{C}'$. For each dummy district v_{dummy}, uniquely choose one of the set districts $v_Y, Y \notin \mathcal{C}'$, and move all voters from v_{dummy} to v_Y. This is possible because there are $r - q$ dummy districts and $r - q$ set districts $v_Y, Y \notin \mathcal{C}'$, and each dummy district is adjacent to each set district.

To show that this indeed gives a $(t+1)q$-biased (t, t)-dissolution, observe that we move all t voters from each dissolved district to the adjacent nondissolved districts. Each inner district receives $\Delta_s = t$ voters: $t - 1$ A-supporters and one B-supporter. Since each inner district initially contained two A-supporters, party A wins a total of $t \cdot q$ inner districts. Each set district $v_Y, Y \in \mathcal{C}'$, receives t A-supporters and initially contains one A-supporter. Furthermore, $|\mathcal{C}'| = q$, and hence party A wins q set districts in total and loses the remaining $r - q$ set districts. Thus, we indeed constructed a $(t + 1)q$-biased (t, t)-dissolution.

For the "if" part, assume that there is some $(t + 1)q$-biased (t, t)-dissolution for the constructed instance. Since $s = \Delta_s$ and G has $2t \cdot q + 2m$ districts, after the dissolution a total number of $t \cdot q + r$ districts is dissolved, and party A wins at least $(t+1)q$ districts and loses at most $r - q$ districts. Observe that the only neighbors of the dummy districts are the set districts, and hence, by the construction of function α, party A cannot win any nondissolved district that receives/contains at least one voter from a dummy district. Furthermore, since the set of the $(r - q)$ dummy districts and the set of their neighboring districts build a bipartite induced subgraph, there are $(r-q)$ nondissolved districts which may receive/contain any voters from the dummy districts. Thus, party A loses at least $r-q$ nondissolved districts. Since $r_\alpha = (t+1)q$, party A loses *exactly* $r-q$ districts. In particular, each of the losing districts contains at least one voter (originally) from a dummy district. This implies that party A has to win each nondissolved set district, element district, inner district, or center district. However, the construction of α forbids A to win a center district or to win an inner district if one moves two B-supporters to it. Thus, we dissolve each center district and move exactly one B-supporter from this center district to each of its adjacent inner districts. As a direct consequence, all element districts are to be dissolved, and $t-1$ voters are moved from each element district to its adjacent inner districts such that A wins all $t \cdot q$ inner districts. There are $t \cdot q$ A-supporters left, one A-supporter from each element district. These voters are to be moved to a set of exactly q winning set districts each. Since each of these districts needs at least t A-supporters to win and has exactly t adjacent element districts, $C' := \{S \in \mathcal{C} \mid v_S \in R_\alpha\}$ partitions X. ☐

5. Special graph classes. First, in Section 5.1, Biased Dissolution on planar graphs is considered. This problem restriction is interesting especially in the political districting context since the neighborhood relation between voting districts on a map is typically planar. We will see that Dissolution (and thus Biased Dissolution) unfortunately remains NP-hard for many choices of s and Δ_s.

Second, in Section 5.2, we show that Biased Dissolution is polynomial-time solvable on cliques, that is, if voters may be moved unrestrictedly between dissolved districts and nondissolved districts.

Finally, in Section 5.3, we consider Biased Dissolution on graphs of bounded treewidth. This problem restriction is interesting in the context of distributed systems since computers are often interconnected using a tree, star, or bus topology. By presenting a formulation of Biased Dissolution in the monadic second-order logic of graphs, we show that Biased Dissolution is solvable in linear time on graphs of bounded treewidth when s and Δ_s are constant. This, however, should be understood as a pure classification result rather than as an implementable algorithm.

5.1. Planar graphs. Computing star partitions is known to be NP-hard even on subcubic grid graphs and split graphs [6]. By Proposition 3.4 in Section 3.2 it follows that Dissolution is also NP-hard on planar graphs because grid graphs are planar. However, the NP-hardness reduction on subcubic grid graphs requires stars with two leaves such that the NP-hardness does only transfer to computing $(1,2)$-dissolutions. Here, we show that NP-hardness for Dissolution holds for any constants s and Δ_s such that Δ_s divides s or s divides Δ_s.

By giving a polynomial-time reduction from the following NP-complete problem, it is easy to derive NP-hardness results for Dissolution.

Perfect Planar H-Matching
Input: A planar undirected graph $G = (V, E)$.
Question: Does G contain an *H-factor* $V_1, V_2, \ldots, V_{\lceil |V|/|V(H)| \rceil}$ that partitions the vertex set V such that $G[V_i]$ is isomorphic to H for all i?

Perfect Planar H-Matching is NP-complete for any connected outerplanar graph H with three or more vertices [5]. In particular, Perfect Planar H-Matching is NP-complete for any H being a star of size at least three. This makes it easy to prove the following theorem.

Theorem 5.1. Dissolution *on planar graphs is NP-complete for all $s \neq \Delta_s$ such that Δ_s divides s or s divides Δ_s. It is polynomial-time solvable for $s = \Delta_s$.*

Proof. We have already shown in Theorem 4.3 how to solve Dissolution in polynomial time for $s = \Delta_s$. Hence, now assume that $\Delta_s \neq s$ and s divides s. Let $x := s/\Delta_s \geq 2$. Due to Proposition 3.4 and the fact that Perfect Planar $K_{1,x}$-Matching is NP-complete [5], we can conclude that Dissolution is NP-complete even on planar graphs. ∎

It seems to be challenging to transfer the dichotomy result for Dissolution on general graphs (Theorem 4.3) to the case of planar graphs. The main problem is that the proof of Theorem 4.3 exploits Exact Cover by t-Sets to be NP-hard for all $t \geq 3$. The reduction from Exact Cover by t-Sets to Dissolution produces a graph that contains the incidence graph of the Exact Cover by t-Sets instance as a subgraph. To obtain a reduction to Dissolution on planar graphs, it is necessary to have planar incidence graphs of Exact Cover by t-Sets. It is, however, unknown whether this problem variant, called Planar Exact Cover by t-Sets, is NP-hard for $t \geq 4$. One might be misled to think that Exact Cover by t-Sets is NP-hard for $t \geq 4$ since it already is NP-hard for $t = 3$. However, the closely related problem Planar 3-Sat, that is, 3-Sat with planar clause-literal incidence graphs, is NP-complete, whereas Planar 4-Sat is polynomial-time solvable: One can show that the clause-literal incidence graph of a Planar 4-Sat instance allows for a matching such that each clause is matched to some literal. These literals can then be simply set to true in order to satisfy all clauses. We consider the question of whether Planar Exact Cover by 4-Sets is NP-hard to be of independent interest.

5.2. Cliques. If the neighborhood graph is a clique, that is, the districts are fully connected such that voters can move from any dissolved district to any nondissolved district, then the existence of an (s, Δ_s)-dissolution depends only on the number $|V|$ of districts, the district size s, and the size increase Δ_s. Clearly, a Dissolution instance is a yes-instance if and only if $d := |V| \cdot \Delta_s/(s + \Delta_s)$ is an integer. We now show that Biased Dissolution is not as easy but still solvable in polynomial time if the neighborhood graph is a clique. The basic idea is to dissolve districts with a large number of A-supporters while minimizing the number of losing districts by letting the districts with the smallest number of A-supporters lose.

Theorem 5.2. Biased Dissolution *on cliques is solvable in $O(|V|^2)$ time.*

Proof. As a matter of fact, we show how to solve the optimization version of Biased Dissolution, where we maximize the number r_α of winning districts. Intuitively, it appears to be a reasonable approach to dissolve districts pursuing the following two objectives. Our *first objective* is that any losing district should contain as few A-supporters as possible. Our *second objective* is that any winning district should contain only as many A-supporters as necessary. Dissolving districts this way minimizes the number of "wasted" A-supporters.

We now show that this greedy strategy is indeed optimal. To this end, let $G = (V, \binom{V}{2})$ be a clique, let α be an A-supporter distribution over V, and let s and Δ_s be the district size and the district size increase. With G being a complete graph, we are free to move voters from any dissolved district to any nondissolved district. Let $\mu := \lfloor (s + \Delta_s)/2 \rfloor + 1$ be the minimum number of A-supporters required to win a district. Thus, a district with less than $(\mu - \Delta_s)$ A-supporters can never win. Define $\mathcal{L} := \{v \in V \mid \alpha(v) < \mu - \Delta_s\}$ to be the set of *nonwinnable* districts.

Our strategy can be sketched as follows (see also Figure 10 for an illustration). Assume that d districts have to be dissolved and ℓ districts have to lose, and let μ denote the number of A-supporters needed to win a district. Sort the districts according to the number of A-supporters. Mark the ℓ districts with the fewest number

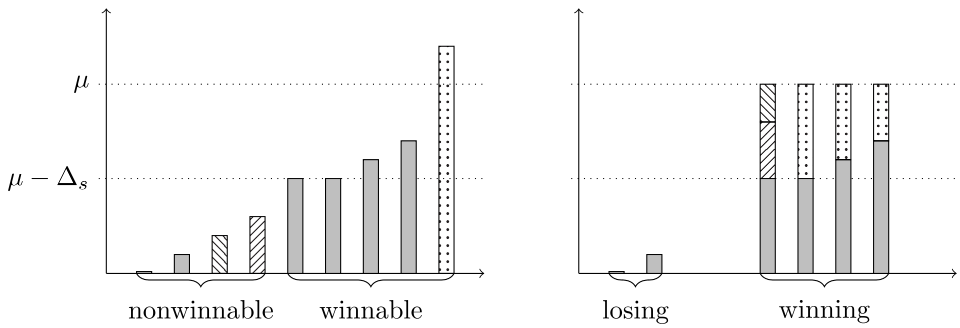

FIG. 10. *Assume that we want to find a 4-biased dissolution for the instance illustrated on the left-hand side, where each district is represented by a bar of height proportional to its number of A-supporters. Following our first objective, we dissolve the two nonwinnable districts with the most A-supporters and, following our second objective, we dissolve the winnable district with the most A-supporters. Nondissolved districts are represented by bars filled with solid gray. Each dissolved district is represented by a bar filled with an individual pattern. The diagram on the right illustrates the solution, where the A-supporters of the two dissolved nonwinnable districts moved to the first winnable district and the A-supporters of the dissolved winnable district moved to the three remaining winnable districts.*

of A-supporters as losing. Dissolve all nonmarked nonwinnable districts. If necessary, then also dissolve winnable districts beginning with those with the most A-supporters until d districts have been dissolved. Finally, check whether this gives a solution.

Our first claim corresponds to the first objective above, that is, the losing districts should contain a minimal number of A-supporters.

CLAIM 1. *Let $v, w \in V$ be two districts with $\alpha(v) \leq \alpha(w)$. If there exists an r_α-biased dissolution where v is winning and w is losing, then there also exists an r_α-biased dissolution where v is losing and w is winning.*

To verify Claim 1, let $(D, z, z_\alpha, R_\alpha)$ be an r_α-biased dissolution. Let $v \in R_\alpha$ and $w \in (V \setminus D) \setminus R_\alpha$ be two districts such that $\alpha(v) \leq \alpha(w)$. Now, simply exchange v and w; that is, set $R'_\alpha := (R_\alpha \setminus \{v\}) \cup \{w\}$ and define for all $(x, y) \in Z(D, G)$,

$$z'(x,y) := \begin{cases} z(x,w) & \text{if } y = v, \\ z(x,v) & \text{if } y = w, \\ z(x,y) & \text{else,} \end{cases} \qquad z'_\alpha(x,y) := \begin{cases} z_\alpha(x,w) & \text{if } y = v, \\ z_\alpha(x,v) & \text{if } y = w, \\ z_\alpha(x,y) & \text{else.} \end{cases}$$

Since $\alpha(v) \leq \alpha(w)$, it is clear that $(D, z', z'_\alpha, R'_\alpha)$ is also a well-defined r_α-biased dissolution.

The next claim basically corresponds to the second objective above, in the sense that districts with a large number of A-supporters (possibly too large, that is, more than the required μ) should be dissolved in order to move the voters more efficiently.

CLAIM 2. *Let $v, w \in V$ be two districts with $\alpha(v) \leq \alpha(w)$. Assume that there exists an r_α-biased (s, Δ_s)-dissolution with maximum r_α. If v is dissolved, then the following hold:*

(i) *If w is losing, then there also exists an r_α-biased dissolution where w is dissolved and v is losing.*

(ii) *If w is winning and v is winnable, that is, $v \notin \mathcal{L}$, then there exists an r_α-biased dissolution where w is dissolved and v is winning.*

Claim 2 also holds by an exchange argument similar to the one above: Let $(D, z, z_\alpha, R_\alpha)$ be an r_α-biased dissolution, and let $v \in D$, $w \in V \setminus D$ be two districts such that $\alpha(v) \leq \alpha(w)$. Again, we exchange v and w by setting $D' := D \setminus \{v\} \cup \{w\}$. Since $\sum_{x \in D'} \alpha(x) \geq \sum_{x \in D} \alpha(x)$ and since we are free to move voters arbitrarily between districts, it is clear that it is always possible to find an r_α-biased dissolution such that D' is the set of dissolved districts. In particular, if v is a winnable district, then it is always possible to make v a winning district.

Using Claims 1 and 2 above, we now show how to compute an optimal biased dissolution. In order to find a biased dissolution with the maximum number of winning districts, we search for a dissolution that loses a minimum number of remaining districts. Thus, for each $\ell \in \{0, \ldots, r\}$, we check whether it is possible to dissolve d districts such that at most ℓ of the remaining r districts lose. To this end, assume that the districts $v_1, \ldots, v_n$ are ordered by increasing number of A-supporters, that is, $\alpha(v_1) \leq \alpha(v_2) \leq \cdots \leq \alpha(v_n)$, and let $V_\ell := \{v_1, \ldots, v_\ell\}$. Now, if there exists an $(r - \ell)$-biased dissolution, then there also exists an $(r - \ell)$-biased dissolution where the losing districts are exactly V_ℓ. This follows by repeated application of the exchange arguments of Claim 1 and Claim 2(i). Hence, given ℓ, we have to check whether there is a set $D \subseteq V \setminus V_\ell$ of d districts that can be dissolved in such a way that all nondissolved districts in $V \setminus (V_\ell \cup D)$ win and the districts in V_ℓ lose.

First, note that in order to achieve this, all districts in $\mathcal{L} \setminus V_\ell$ have to be dissolved because they cannot win in any way. Clearly, if $|\mathcal{L} \setminus V_\ell| > d$, then it is simply not possible to lose only ℓ districts, and we can immediately go to the next iteration with $\ell := \ell + 1$. Therefore, we assume that $|\mathcal{L} \setminus V_\ell| \leq d$ and let $d' := d - |\mathcal{L} \setminus V_\ell|$ be the number of additional districts to dissolve in $V \setminus (\mathcal{L} \cup V_\ell)$. By Claim 2(ii), it follows that we can assume that the d' districts with the maximum number of A-supporters are dissolved, that is, $V^{d'} := \{v_{n-d'+1}, \ldots, v_n\}$. Thus, we set $D := \mathcal{L} \setminus V_\ell \cup V^{d'}$ and check whether there are enough A-supporters in D to let all $r - \ell$ remaining districts in $V \setminus (V_\ell \cup D)$ win.

Sorting the districts by the number of A-supporters (in a preprocessing step) requires $O(n \log n)$ comparisons. Then, for up to n values of ℓ, to check whether the remaining districts in $V \setminus (V_\ell \cup D)$ can win requires $O(n)$ arithmetic operations each. Thus, assuming constant-time arithmetic, we end up with a total running time in $O(n^2)$. $\square$

5.3. Graphs of bounded treewidth. Yuster [29, Theorem 2.3] showed that H-Factor is solvable in linear time on graphs of bounded treewidth when the size of H is constant. This includes the case of finding x-star partitions, that is, $(x, 1)$-dissolutions (respectively, $(1, x)$-dissolutions) when x is constant. We can show that the more general problem Biased Dissolution is solvable in linear time on graphs of bounded treewidth when s and Δ_s are constants. In terms of parameterized complexity analysis [9, 14, 24], this shows that Biased Dissolution is fixed-parameter tractable with respect to the combined parameter (t, s, Δ_s), where t is the treewidth of the neighborhood graph. Note that these results are basically for classification only since the corresponding algorithms come along with enormous constants hidden in the O-notation.

Theorem 5.3. Biased Dissolution *is solvable in linear time on graphs of constant treewidth when s and Δ_s are constants.*

To prove Theorem 5.3, we exploit a general result that a maximum-cardinality set satisfying a constant-size formula in monadic second-order logic for graphs can be computed in linear time on graphs of constant bounded treewidth [3]. The set whose size we want to maximize is the set R_α of winning districts. For the remainder of this

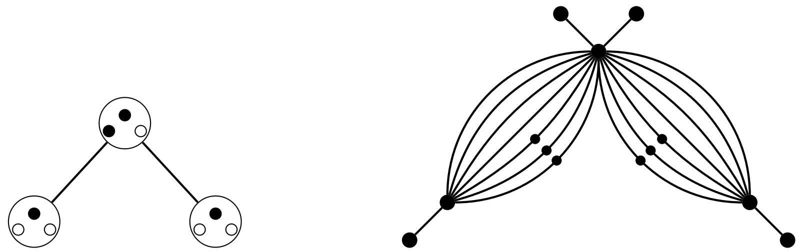

FIG. 11. *Illustration of transforming a* BIASED DISSOLUTION *instance (left) into an instance of the auxiliary multigraph problem (right).*

subsection, we consider multigraphs; that is, our graphs may contain multiple edges between any two vertices.

DEFINITION 5.4 (monadic second-order logic for graphs). *A formula ϕ of the monadic second-order logic for graphs may consist of the logic operators $\vee, \wedge, \neg$, vertex variables, edge variables, set variables, quantifiers $\exists$ and $\forall$ over vertices, edges, and sets, and the predicates*

(i) *$x \in X$ for a vertex or edge variable x and a set X,*
(ii) *$\mathrm{inc}(e, v)$, being true if e is an edge incident to the vertex v,*
(iii) *$\mathrm{adj}(v, w)$, being true if v and w are adjacent vertices, and*
(iv) *equality of vertex variables, edge variables, and set variables.*

We point out that a constant-size formula in monadic second-order logic for a problem does not prove only the mere existence of a linear-time algorithm on graphs of bounded treewidth; the formula itself can be converted into a linear-time algorithm [8, Chapter 6].

Proof of Theorem 5.3. We model BIASED DISSOLUTION as a formula in monadic second-order logic. Since monadic second-order logic does not allow us to count the number of voters moved from one district to another or to count how many A-supporters a district contains, we first model BIASED DISSOLUTION as a problem on an auxiliary graph. For constant s and Δ_s, the transformation of a BIASED DISSOLUTION instance to this auxiliary graph can be done in linear time and works as follows (see Figure 11):

1. For each input district of BIASED DISSOLUTION, introduce a vertex and attach to it as many degree-one vertices as the district has A-supporters.
2. Between two neighboring districts, add $s+1$ (multiple) edges between their representing vertices. The $s+1$ (multiple) edges represent potential moves of voters from one district to another.
3. Finally, connect each pair of vertices representing a pair of neighboring districts by s parallel subdivided edges. These represent potential moves of A-supporters.

Note that, by adding $s+1$ (multiple) edges between any two vertices representing neighboring districts, we ensure that, in the graph resulting from the above construction, a vertex has degree one if and only if it represents an A-supporter. The vertex representing an A-supporter belongs to the district represented by its neighbor. Moreover, a vertex has degree two if and only if it represents a possible movement of an A-supporter of one district to another.

A dissolution now does not contain a function z moving voters from one district to another (see Definition 2.1) but a set Z of selected edges representing such movements. Similarly, the A-supporter movement is no longer modeled as a function z_α

(see Definition 2.2) but as a set of vertices Z_α representing such movements. Hence, we search for a maximum vertex-set R_α that satisfies the following formula in monadic second-order logic of graphs:

$$\begin{aligned}\max R_\alpha \text{ s.t. } \exists D \exists Z \exists Z_\alpha[&\text{movements} \wedge \text{A-movements} \wedge \text{districts}\\ &\wedge \text{prop-}a \wedge \text{prop-}b \wedge \text{prop-}c \wedge \text{prop-}d \wedge \text{prop-}e],\end{aligned}$$

where prop-a, prop-b, prop-c, prop-d, and prop-e will be predicates ensuring that properties (a) and (b) of Definition 2.1 and properties (a)–(c) of Definition 2.2 of (biased) dissolution are satisfied, D will be the set of dissolved districts, Z the set of voter movements, and Z_α the set of A-supporter movements. To ensure this, we define

$$\text{districts} := \forall v[(v \in D \vee v \in R_\alpha) \implies \text{degree-greater-two}(v)]$$

so that it is true if and only if each element in $D \cup R_\alpha$ is a vertex with degree more than two, that is, it represents a district, where

$$\begin{aligned}\text{degree-greater-two}(v) := \exists v_1 \exists v_2 \exists v_3 [\, &v_1 \neq v_2 \wedge v_1 \neq v_3 \wedge v_2 \neq v_3\\ &\wedge \text{adj}(v_1, v) \wedge \text{adj}(v_2, v) \wedge \text{adj}(v_3, v)]\end{aligned}$$

is true if and only if v has at least three neighbors. Moreover, we define

$$\begin{aligned}\text{movements} := \forall e[\, e \in Z \implies \exists v_1 \exists v_2 [&\text{inc}(e, v_1) \wedge \text{inc}(e, v_2)\\ &\wedge \text{degree-greater-two}(v_1) \wedge v_1 \in D\\ &\wedge \text{degree-greater-two}(v_2) \wedge v_2 \notin D]]\end{aligned}$$

so that it is true if and only if each element in the set Z is an edge representing a movement and

$$\begin{aligned}\text{A-movements} := \forall a[\, a \in Z_\alpha \implies \exists v_1 \exists v_2 [&\text{adj}(a, v_1) \wedge \text{adj}(a, v_2)\\ &\wedge v_1 \in D \wedge v_2 \notin D \wedge \neg\text{degree-greater-two}(a)]]\end{aligned}$$

so that it is true if and only if each element in the set Z_α is a vertex representing a movement of an A-supporter. It remains to give the definitions of the predicates prop-a, prop-b, prop-c, prop-d, and prop-e. We define

$$\text{prop-}a := \forall v[v \in D \implies \exists Z'[\text{card}_s(Z') \wedge (\forall e[e \in Z' \iff \text{move-from}(e, v)])]]$$

so that it is true if and only if for each dissolved district v there is a set of s edges representing movements out of v, where

$$\text{move-from}(e, v) := v \in D \wedge e \in Z \wedge \text{inc}(e, v)$$

is true if and only if e is an edge representing a movement out of v and

$$\begin{aligned}\text{card}_i(X) := \exists x_1 \exists x_2 \dots \exists x_i \Bigg[&\Big(\bigwedge_{j=1}^{i} x_i \in X\Big) \wedge \Big(\bigwedge_{j=1}^{i} \bigwedge_{k=j+1}^{i} (x_j \neq x_k)\Big)\\ &\wedge \forall x\Big[x \in X \implies \bigvee_{j=1}^{i} x_j = x\Big]\Bigg]\end{aligned}$$

for $1 \leq i \leq s$ is a constant-size formula that is true if and only if the set X has cardinality i. Next, we define

$$\begin{aligned}\text{prop-}b := \forall v[\,(\text{degree-greater-two}(v) \land v \notin D)\\ \implies \exists Z'[\text{card}_{\Delta_s}(Z') \land (\forall e[e \in Z' \iff \text{move-to}(e, v)])]]\end{aligned}$$

so that it is true if and only if there is a set Z' of Δ_s edges representing movements to each nondissolved district v, where

$$\text{move-to}(e, v) := v \notin D \land e \in Z \land \text{inc}(e, v)$$

is true if and only if e is an edge representing a movement to v. Next, we define

$$\begin{aligned}\text{prop-}c := \forall v \forall u[v \in D \land u \notin D \land \text{adj}(v, u) \implies \exists Z' \exists Z'_\alpha[&\text{smaller-equal}(Z'_\alpha, Z')\\ &\land (\forall e[e \in Z' \iff \text{move-from}(e, v)\\ &\qquad\qquad\land \text{move-to}(e, u)])\\ &\land (\forall a[a \in Z'_\alpha \iff \text{A-move-from}(a, v)\\ &\qquad\qquad\land \text{A-move-to}(a, u)])]]\end{aligned}$$

so that it is true if and only if the number of vertices representing A-supporters movements from v to u is at most the number of edges representing movements from v to u, where

$$\text{smaller-equal}(X, Y) := \bigvee_{i=1}^{s} \bigvee_{j=i}^{s} (\text{card}_i(X) \land \text{card}_j(Y))$$

is a constant-size formula that is true if and only if $|X| \leq |Y|$ and

$$\begin{aligned}\text{A-move-from}(a, v) &:= v \in D \land a \in Z_\alpha \land \text{adj}(v, a),\\ \text{A-move-to}(a, u) &:= u \notin D \land a \in Z_\alpha \land \text{adj}(u, a)\end{aligned}$$

are true if and only if a is a vertex representing an A-supporter movement from v or to u, respectively. Next, we define

$$\begin{aligned}\text{prop-}d := \forall v[v \in D \implies \exists Z'_\alpha \exists A[&\text{equal-card}(Z'_\alpha, A)\\ &\land \forall a[a \in A \iff \text{A-supporter-of}(a, v)]\\ &\land \forall a[a \in Z'_\alpha \iff \text{A-move-from}(a, v)]]]\end{aligned}$$

so that it is true if and only if the number of A-supporter movements out of a district v equals the number of its A-supporters, where

$$\text{equal-card}(X, Y) := \bigvee_{i=1}^{s} (\text{card}_i(X) \land \text{card}_i(Y))$$

is a constant-size formula that is true if and only if $|X| = |Y|$ and

$$\text{A-supporter-of}(a, v) := \text{adj}(a, v) \land \forall u[\text{adj}(a, u) \implies u = v]$$

is true if and only if a is a vertex representing an A-supporter in district v. Finally, we define

$$\begin{aligned}\text{prop-}e := \forall v[v \in R_\alpha \implies \exists A[&\text{card}_{>(s+\Delta_s)/2}(A)\\ &\land \forall a[a \in A \iff \text{A-supporter-of}(a, v)\\ &\qquad\qquad \lor \text{A-move-to}(a, v)]]]\end{aligned}$$

so that for each district $v \in R_\alpha$ there are more than $(s + \Delta_s)/2$ vertices which either represent A-supporters of district v or represent A-supporter movements to district v, where

$$\mathrm{card}_{>i}(X) := \bigvee_{j=\lfloor i \rfloor+1}^{s+\Delta_s} \mathrm{card}_j(X)$$

is a constant-size formula that is true if and only if $i < |X| \leq s + \Delta_s$ with $i < s + \Delta_s$. ◻

Without providing any details, we claim that one can also prove Theorem 5.3 by using an explicit dynamic programming algorithm that works on a so-called tree decomposition of a graph. The algorithm runs in $(\Delta_s + s)^{O(t^2)} \cdot n^{O(1)}$ time, but it is very technical and its correctness proof is very tedious, while practical applicability still seems out of reach.

6. Conclusion. We initiated a graph-theoretic approach to concrete redistribution problems with potential applications in such diverse areas as political districting, green computing, and economization of work processes. Obviously, the two basic problems Dissolution and Biased Dissolution concern highly simplified situations and will not be able to model all interesting aspects of redistribution scenarios. For instance, our constraint that before and after the dissolution all vertex loads should be perfectly balanced may be too restrictive for many applications. All in all, we consider our simple (and yet fairly realistic) models as a first step into a promising direction for future research. In particular, this may yield a stronger linking of graph-theoretic concepts with districting scenarios and other application scenarios.

We end with a few specific challenges for future research. We left open whether the P versus NP dichotomy for general graphs fully carries over to the planar case: It might be possible that planar graphs allow for some further tractable cases with respect to the relation between old and new district sizes. To this end, it might help to answer the question of whether Planar Exact Cover by 4-Sets is NP-hard. Since Planar Exact Cover by 4-Sets is a very natural and simple problem on planar graphs, we believe that this question is of independent interest. Moreover, with redistricting applications in mind it might be of interest to study special cases of planar graphs (such as grid-like structures) in the quest of finding polynomial-time solvable special cases of network-based vertex dissolution problems. Having identified several NP-complete special cases of Dissolution and Biased Dissolution, it is a natural endeavor to investigate their polynomial-time approximability and their parameterized complexity; in the latter case one also needs to identify fruitful parameterizations. Motivated by our results, parameters measuring the distance to acyclic graphs (cf. Theorem 5.3) or to complete graphs (cf. Theorem 5.2) seem promising in the spirit of distance from triviality parameterizations [17, 25]. Furthermore, also the maximum degree of a vertex should not be too large in many applications. On the one hand, since the partition of a graph into paths of length three, which is a special case of our Dissolution problem, is already NP-hard on graphs with maximum degree at most three [20, 23], the parameter "maximum degree" is not interesting as a single parameter. On the other hand, the maximum degree might be worth considering in combination with other parameters.

Acknowledgments. We thank the anonymous referees for helpful comments.

REFERENCES

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice–Hall, Englewood Cliffs, NJ, 1993.

[2] M. Altman, *Districting Principles and Democratic Representation*, Ph.D. thesis, California Institute of Technology, Pasadena, CA, 1998.

[3] S. Arnborg, J. Lagergren, and D. Seese, *Easy problems for tree-decomposable graphs*, J. Algorithms, 12 (1991), pp. 308–340.

[4] J. J. Bartholdi III, C. A. Tovey, and M. A. Trick, *How hard is it to control an election?*, Math. Comput. Modelling, 16 (1992), pp. 27–40.

[5] F. Berman, D. Johnson, T. Leighton, P. W. Shor, and L. Snyder, *Generalized planar matching*, J. Algorithms, 11 (1990), pp. 153–184.

[6] R. van Bevern, R. Bredereck, L. Bulteau, J. Chen, V. Froese, R. Niedermeier, and G. J. Woeginger, *Star partitions of perfect graphs*, in Proceedings of the 41st International Colloquium on Automata, Languages, and Programming (ICALP' 14), Lecture Notes in Comput. Sci. 8572, Springer, 2014, pp. 174–186; full version available from arXiv:1402.2589 [cs.DM].

[7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed., The MIT Press, Cambridge, MA, 2009.

[8] B. Courcelle and J. Engelfriet, *Graph Structure and Monadic Second-Order Logic—A Language-Theoretic Approach*, Encyclopedia Math. Appl. 138, Cambridge University Press, Cambridge, UK, 2012.

[9] R. G. Downey and M. R. Fellows, *Fundamentals of Parameterized Complexity*, Springer, London, 2013.

[10] J. C. Duque, *Design of Homogeneous Territorial Units: A Methodological Proposal and Applications*, Ph.D. thesis, University of Barcelona, Barcelona, Spain, 2004.

[11] J. C. Duque, R. Ramos, and J. Surinach, *Supervised regionalization methods: A survey*, Int. Reg. Sci. Rev., 30 (2007), pp. 195–220.

[12] J. Edmonds and E. L. Johnson, *Matching: A well-solved class of integer linear programs*, in Combinatorial Structures and Their Applications, Gorden and Breach, New York, 1970, pp. 89–92.

[13] P. Faliszewski and J. Rothe, *Control and bribery in voting*, in Handbook of Computational Social Choice, Cambridge University Press, Cambridge, UK, 2015.

[14] J. Flum and M. Grohe, *Parameterized Complexity Theory*, Springer, Berlin, 2006.

[15] H. N. Gabow, *Data structures for weighted matching and nearest common ancestors with linking*, in Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '90), 1990, pp. 434–443.

[16] M. R. Garey and D. S. Johnson, *Computers and Intractability*, W. H. Freeman, San Francisco, 1979.

[17] J. Guo, F. Hüffner, and R. Niedermeier, *A structural view on parameterizing problems: Distance from triviality*, in Proceedings of the 1st International Workshop on Parameterized and Exact Computation, Lecture Notes in Comput. Sci. 3162, Springer, Berlin, 2004, pp. 162–173.

[18] P. Hell, D. Kirkpatrick, J. Kratochvíl, and I. Kříž, *On restricted two-factors*, SIAM J. Discrete Math., 1 (1988), pp. 472–484.

[19] Z. Landau and F. E. Su, *Fair division and redistricting*, Soc. Choice Welfare, 32 (2009), pp. 479–492.

[20] M. Małafiejski and P. Żyliński, *Weakly cooperative guards in grids*, in Proceedings of the International Conference on Computational Science and Its Applications (ICCSA '05), Lecture Notes in Comput. Sci. 3480, Springer, Berlin, 2005, pp. 647–656.

[21] M. Maravalle and B. Simeone, *A spanning tree heuristic for regional clustering*, Comm. Stat. Theor. Meth., 24 (1995), pp. 625–639.

[22] A. Mehrota, E. L. Johnson, and G. L. Nemhauser, *An optimization based heuristic for political districting*, Manag. Sci., 44 (1998), pp. 1100–1114.

[23] J. Monnot and S. Toulouse, *The path partition problem and related problems in bipartite graphs*, Oper. Res. Lett., 35 (2007), pp. 677–684.

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

[25] R. Niedermeier, *Reflections on multivariate algorithmics and problem parameterization*, in Proceedings of the 27th International Symposium on Theoretical Aspects of Computer Science (STACS'10), LIPIcs. Leibniz Int. Proc. Inform. 5, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Wadern, Germany, 2010, pp. 17–32.

[26] C. Puppe and A. Tasnádi, *A computational approach to unbiased districting*, Math. Comput.

Modelling, 48 (2008), pp. 1455–1460.

[27] C. Puppe and A. Tasnádi, *Optimal redistricting under geographical constraints: Why "pack and crack" does not work*, Econ. Lett., 105 (2009), pp. 93–96.

[28] V. Vassilevska Williams, *Multiplying matrices faster than Coppersmith-Winograd*, in Proceedings of the 44th ACM Symposium on Theory of Computing (STOC' 12), 2012, pp. 887–898.

[29] R. Yuster, *Combinatorial and computational aspects of graph packing and graph decomposition*, Comput. Sci. Rev., 1 (2007), pp. 12–26.