



# Computation and efficiency of potential function minimizers of combinatorial congestion games

Pieter Kleer<sup>1</sup> · Guido Schäfer<sup>2,3</sup>

Received: 10 March 2019 / Accepted: 15 July 2020  
© The Author(s) 2020

## Abstract

We study the computation and efficiency of pure Nash equilibria in combinatorial congestion games, where the strategies of each player  $i$  are given by the binary vectors of a polytope  $P_i$ . Our main goal is to understand which structural properties of such *polytopal congestion games* enable us to derive an efficient equilibrium selection procedure to compute pure Nash equilibria with attractive social cost approximation guarantees. To this aim, we identify two general properties of the underlying *aggregation polytope*  $P_N = \sum_i P_i$  which are sufficient for our results to go through, namely the *integer decomposition property (IDP)* and the *box-totally dual integrality property (box-TDI)*. Our main results for polytopal congestion games satisfying IDP and box-TDI are as follows: (i) we show that pure Nash equilibria can be computed in polynomial time. In fact, we obtain this result through a general framework for separable convex function minimization, which might be of independent interest. (ii) We bound the inefficiency of these equilibria and show that this provides a tight bound on the price of stability. (iii) We also prove that these results extend to strong equilibria for the “bottleneck variant” of polytopal congestion games. Examples of polytopal congestion games satisfying IDP and box-TDI include common source network congestion games, symmetric totally unimodular congestion games, non-symmetric matroid congestion games and symmetric matroid intersection congestion games (in particular,  $r$ -arborescences and strongly base-orderable matroids).

**Mathematics Subject Classification** 90C27 · 90C10 · 91A10

---

A preliminary version of this paper appeared in the Proceedings of the 18th ACM Conference on Economics and Computation (EC 2017), pp. 223–240, ACM Press, 2017.

---

Pieter Kleer was supported by NWO Gravitation Project NETWORKS, Grant Number 024.002.003.

---

Extended author information available on the last page of the article

# 1 Introduction

## 1.1 Motivation and background

Congestion games constitute an important class of strategic games which have been studied intensively since their introduction by Rosenthal [43] in 1973. In a *congestion game*, a (finite) set of players compete over a (finite) set of resources. Each resource is associated with a non-negative and non-decreasing cost (or delay) function which specifies its cost depending on the total number of players using it. Every player chooses a subset of resources from a set of available resource subsets (corresponding to the player's strategies) and experiences a cost equal to the sum of the costs of the chosen resources. The goal of each player is to minimize her individual cost. In this context, a *pure Nash equilibrium* is a strategy profile such that no player can decrease her cost by unilaterally deviating to another feasible resource subset. Congestion games are both theoretically appealing and practically relevant. For example, they find their applications in network routing, resource allocation and scheduling problems.

The study of the existence, computation and inefficiency of pure Nash equilibria of these games has led to a surge of novel methodologies and techniques. Rosenthal [43] establishes the existence of pure Nash equilibria in congestion games. He proves this result through the use of an *exact potential function*, which assigns a value to each strategy profile such that the difference in potential value of any two strategy profiles corresponding to a unilateral deviation of a player is equal to the difference in cost experienced by that player. He shows that every congestion game admits an exact potential function, also known as *Rosenthal's potential*.<sup>1</sup> As a consequence, every *best response sequence*, i.e., where players iteratively play their best response, must converge to a pure Nash equilibrium (if the game is finite). Further, this shows that the set of pure Nash equilibria corresponds to the set of local minima of Rosenthal's potential (where the neighborhood of a strategy profile is defined by all unilateral player deviations). Especially this latter correspondence has helped to shed light on several important aspects of congestion games in recent years.

One aspect that has been studied intensively is the computational complexity of finding pure Nash equilibria in congestion games. In a seminal paper, Fabrikant et al. [21] show that the problem of finding a pure Nash equilibrium is PLS-complete, both for symmetric congestion games and non-symmetric network congestion games. In particular, this suggests that a polynomial time algorithm for finding a pure Nash equilibrium is unlikely to exist for these games. In their proof they construct instances of non-symmetric network congestion games where any best response sequence has exponential length from certain initial configurations. Ackermann et al. [1] strengthen this result by exhibiting instances of symmetric network congestion games for which every best response sequence (from certain initial configurations) has exponential length. On the positive side, they prove that best response dynamics converge in polynomial time for non-symmetric matroid congestion games, where the available resource subsets of the players correspond to bases of a given matroid (see below

<sup>1</sup> In fact, Monderer and Shapley [40] show that the class of congestion games is isomorphic to the class of strategic games for which an exact potential function exists.

for formal definitions). The authors also show that basically this is the only class of congestion games for which this property holds true.

Most previous works in this context focus on the analysis of decentralized dynamics to reach a pure Nash equilibrium (see, e.g., [1,9,12,20,21,24,34]); said differently, these works focus on finding a local minimum of Rosenthal's potential. Much less is known about the problem of computing a pure Nash equilibrium that corresponds to a *global* minimizer. Fabrikant et al. [21] use this idea to show that a pure Nash equilibrium can be computed in polynomial time for symmetric network congestion games. The authors observe that in this case a global minimizer of Rosenthal's potential can be computed by a reduction to a min-cost flow problem (assuming that the cost functions are non-decreasing). Note that this is in stark contrast with the fact that best response dynamics might take exponential time for this class of congestion games [1].

Only very recently, Del Pia et al. [16] make further progress along these lines. The authors consider congestion games where the strategy sets of the players are given implicitly by a *polyhedral description* (see also [8]). More precisely, for each player  $i$  the incidence vectors of the strategies are defined as the binary vectors in a polytope  $P_i = \{x_i : A_i x_i \leq b_i\}$ , where  $A_i$  is an integral matrix and  $b_i$  is an integral vector. They (mostly) focus on the case where the matrix  $A_i$  is *totally unimodular* (see below for formal definitions) and thus the describing polytope  $P_i$  is *integral*, i.e., all its extreme points are integral; they term these games *totally unimodular (TUM) congestion games*. For *symmetric* TUM congestion games, i.e., when all  $A_i, b_i$  are identical, they devise an aggregation/decomposition framework that reduces the problem of finding a global minimum of Rosenthal's potential to an integer linear programming problem. Using this framework, they show that pure Nash equilibria can be computed efficiently for symmetric TUM congestion games. The authors also show that this problem is PLS-complete for non-symmetric TUM congestion games. Further, they show that their framework can be adapted to the case of non-symmetric matroid congestion games.

Another important aspect that has been the subject of intensive research in recent years is the inefficiency of pure Nash equilibria in congestion games (see, e.g., [3,5,7,10,11,17,22,24,29,35,36,45]). Here the goal is to assess the *social cost* (defined as the sum of the costs of the players) of a pure Nash equilibrium relative to an optimal outcome. Koutsoupias and Papadimitriou [35] introduced the *price of anarchy* as the ratio between the worst social cost of a Nash equilibrium and the social cost of an optimum. Anshelevich et al. [4] defined the *price of stability* as the ratio between the best social cost of a Nash equilibrium and the social cost of an optimum.

Fotakis [24] reveals an intriguing connection between the price of stability of network congestion games and the price of anarchy of their *non-atomic* counterparts. More specifically, he shows that for *symmetric network congestion games* the ratio between the social cost of a (global) minimizer of Rosenthal's potential and the optimal social cost is at most  $\rho(\mathcal{D})$ , where  $\rho(\mathcal{D})$  is a bound on the price of anarchy for *non-atomic network congestion games* with cost functions in class  $\mathcal{D}$  introduced by Correa et al. [13]. In particular, this implies that the price of stability of symmetric network congestion games with cost functions in  $\mathcal{D}$  is at most  $\rho(\mathcal{D})$ . For example, this parameter equals  $4/3$  for the class of affine functions and  $(27 + 6\sqrt{3})/23 \approx 1.63$  for quadratic functions. Further, Fotakis [24] also shows that for symmetric network con-

gestion games on *extension-parallel graphs*,<sup>2</sup> every Nash equilibrium is a Rosenthal minimizer and thus the upper bound of  $\rho(\mathcal{D})$  even holds for the price of anarchy of these games.

Bounds of this type also fall within the *smoothness framework* of Roughgarden [45]. More recently, Feldman et al. [22] study the price of anarchy in *large games*. Basically, they show that the smoothness parameter introduced in [45] also provides an upper bound on the price of anarchy of many games (or mechanisms) when the number of involved players grows large.

## 1.2 Our contributions

In light of the discussion above a natural question that arises, and which we address in this paper, is:

Which structural properties of the strategy sets of the players are sufficient to (A) efficiently compute a global minimum of Rosenthal's potential, and (B) bound the inefficiency of the resulting pure Nash equilibrium?

In order to tackle this question, we use a polyhedral approach similar to the ones used in [8,16]. But in contrast to these works, we do not restrict our attention to polyhedral descriptions arising from totally unimodular matrices only. Instead, we identify more general polyhedral properties of the describing systems that are sufficient to achieve (A) and (B). By doing so, we unify and extend the results in [16,24] to a much larger class of polytopal congestion games.

More specifically, we consider *polytopal congestion games* in which the incidence vectors of the strategies of player  $i$  are given by the binary vectors in a polytope  $P_i = \{x_i : Ax_i \leq b_i\}$ , where  $A$  is an integral matrix and  $b_i$  is an integral vector. Given the polytopes of all players, a strategy profile naturally corresponds to an integral vector in the *aggregation polytope*  $P_N = \sum_i P_i$ . We identify two general properties of the aggregation polytope  $P_N$  which are sufficient for our results to go through, namely the *integer decomposition property (IDP)* and the *box-totally dual integrality property (box-TDI)* (formal definitions are given below). The integer decomposition property is needed to decompose a load profile in  $P_N$  to a respective strategy profile of the players; while this property is always satisfied for TUM games, it requires additional care if one wants to go beyond this class of games. Intuitively, the box-TDI property ensures that the intersection of  $P_N$  with an arbitrary integer box is an integral polytope; this property is mostly needed for technical reasons.

Our main contributions for polytopal congestion games are as follows:

- (1) We derive an efficient algorithm for computing a feasible load profile minimizing Rosenthal's potential for polytopal congestion games satisfying IDP and box-TDI (Sect. 3). The time complexity of this algorithm is polynomial in the number of players and resources, the encoding length of  $\sum_i b_i$  and the complexity of a separation oracle for the aggregation polytope.

<sup>2</sup> Extension-parallel graphs are special cases of series-parallel graphs, where in a series composition we may only attach a single edge to an extension-parallel graph.

- (2) We show that the price of stability of polytopal congestion games satisfying IDP and box-TDI is bounded by  $\rho(\mathcal{D})$  (Sect. 4). To this aim, we introduce a novel structural property (which we term the *symmetric difference decomposition property*) and show that it is satisfied by our games. We also prove that our bound is tight.
- (3) We give several examples of polytopal congestion games satisfying IDP and box-TDI (Sect. 5). These examples include symmetric TUM congestion games, common source network congestion games, non-symmetric matroid congestion games and certain symmetric matroid intersection congestion games (in particular,  $r$ -arborescences and strongly base-orderable matroids).
- (4) We show that our techniques can be used to extend some results on the computation and inefficiency of *strong equilibria* of the “bottleneck variant” of our polytopal congestion games (Sect. 6). In particular, we show that strong equilibria can be computed in (strongly) polynomial time for *polytopal bottleneck congestion games* satisfying IDP and box-TDI (see below for definitions).

To the best of our knowledge, all previous works addressed either (A) or (B), but not both. Note that the combination of our contributions (1) and (2) provides an efficient algorithm for the computation of a pure Nash equilibrium that comes with a provable inefficiency guarantee. Said differently, (1) and (2) can be seen as an efficient equilibrium selection procedure to find a pure Nash equilibrium whose social cost is at most  $\rho(\mathcal{D})$  times the optimal social cost.

### 1.3 Significance and implications of our results

Our algorithm in (1) unifies and extends the aggregation/decomposition framework of [16]. In particular, the symmetric TUM congestion games and non-symmetric matroid congestion games (considered separately in [16]) fall into our class of polytopal congestion games satisfying IDP and box-TDI. Similarly, all combinatorial TUM congestion games (i.e., network, matching, edge cover, vertex cover and stable set congestion games) and their respective extensions to the maximum (or minimum) cardinality versions considered in [16] can be handled by our framework.

Besides its implications for polytopal congestion games mentioned above, our algorithm (1) can alternatively be interpreted as an efficient approach to minimize a separable convex function over certain types of polytopes satisfying IDP and box-TDI, which might be of independent interest. We elaborate on this in more detail in Sect. 3.

By exploiting our symmetric difference decomposition property in (2), we can generalize the bound of Fotakis [24] on the price of stability for symmetric network congestion games to the larger class of polytopal congestion games satisfying IDP and box-TDI. Our bounds for polytopal congestion games (significantly) improve upon the ones known for general congestion games: For example, the price of stability of congestion games with cost functions from the class  $\mathcal{D}_d$  of polynomial functions of maximum degree  $d$  is well-understood (see [10]) and grows like  $d + 1$ . However, it is known that  $\rho(\mathcal{D}_d) \approx d / \log(d)$  for large  $d$  (see [44]). We thus obtain improved bounds for polytopal congestion games. Further,  $\rho(\mathcal{D})$  is well-understood for various classes

of cost functions  $\mathcal{D}$  and we can import these bounds without further analysis. For example, an exact (closed form) expression is known for  $\rho(\mathcal{D}_d)$  (see preliminaries).

Our upper bound of  $\rho(\mathcal{D})$  on the price of stability is (asymptotically) tight even for symmetric singleton congestion games if the class of cost functions  $\mathcal{D}$  contains all constant functions and is closed under dilations (see below for details). Note that singleton congestion games constitute a special case of all the examples of polytopal congestion games mentioned in (3). In particular, our results settle the exact price of stability for all these games.

Our contribution (4) generalizes a result by Harks et al. [32] who derive an algorithm to compute a strong equilibrium for certain special cases of bottleneck congestion games, using a *strategy packing oracle* as a subroutine (details are given in Appendix B). They give efficient packing oracles for symmetric network congestion games, non-symmetric matroid congestion games and a (slight) generalization of  $r$ -arborescences. We adapt their algorithm to compute a load profile of a strong equilibrium for bottleneck polytopal congestion games satisfying the IDP and box-TDI property. In particular, this leads to an improved algorithm (in terms of running time) to compute a strong equilibrium for this (larger) class of polytopal congestion games.

As a side result, which might be of independent interest, we derive a combinatorial approach for computing the symmetric difference decomposition for non-symmetric matroid congestion games. This also gives rise to a local search algorithm which can be seen as a natural generalization of best response dynamics. In particular, we show that, using a slightly altered neighborhood notion, every best response sequence arrives at a *global* optimum of Rosenthal's potential (in a polynomial number of steps). These results are presented in Appendix A.

Contribution (1) can also be regarded as a “black-box” approach for the computation of a pure Nash equilibrium. Given a congestion game that exhibits some combinatorial structure, checking whether our approach applies reduces to the following three tasks: (i) derive a polytopal description  $P_i$  for the strategy set of each player  $i$ , (ii) verify whether the resulting aggregation polytope  $P_N$  satisfies the IDP, (iii) check that the system describing the aggregation polytope  $P_N$  is box-TDI. In particular, if the integer decomposition of  $P_N$  can be done in polynomial time, then this approach provides an efficient algorithm to compute a pure Nash equilibrium. By exploiting this idea, we derive strongly polynomial time algorithms for the computation of potential function minimizers for all applications mentioned in (3).

It is interesting to note that the IDP seems to be the limiting property for our approach to apply. For example, non-symmetric network congestion games can naturally be modeled as polytopal congestion games satisfying box-TDI. But it is easy to see that the IDP does not hold. In fact, it is unlikely that an efficient algorithm to find a pure Nash equilibrium exists because this problem is PLS-complete [21].

## 1.4 Further comparison with related work

Our results in (1) regarding the computation of a minimizer of Rosenthal's potential function can also be interpreted as minimizing a separable convex function over a

polytope described by a system  $Ax \leq b$ . There is a large body of work on the latter problem and we elaborate only on a few related works in this context below.

Hochbaum and Shanthikumar [33] show that optimizing a separable convex function over a polytope described by a system  $Ax \leq b$  is not much harder than solving a linear function over the same polytope if the maximum subdeterminant of the matrix  $A$  is polynomially bounded. In fact, the respective results in Fabrikant et al. [21] and Del Pia et al. [16] could alternatively have been derived from [33] (in combination with an efficient procedure to decompose the obtained solutions into strategy profiles), as in both cases the matrix  $A$  is totally unimodular (and, hence, has subdeterminants  $0, \pm 1$ ). However, the setting of box-TDI systems that we consider here is not covered by the result in [33]. The column-doubling trick used in [33] is also exploited here (as well as in [16,21]); see Sect. 3 for details.

The special case of TUM systems where the underlying system describes a single-commodity network flow problem has also been studied extensively. In particular, in this case it is folklore that the problem can be reduced to a minimum cost network flow problem. This idea dates back to the works by Dantzig [15] and Ford and Fulkerson [23] (see also [38,39]). This idea is (implicitly) also exploited in [16,21].

Moreover, there has been an extensive line of work regarding the minimization of a separable convex function over a polymatroid polytope, see, e.g., the work of Fujishige [26] or Groenevelt [30]. In particular, these algorithms can be used alternatively in our application regarding non-symmetric matroid congestion games in Sect. 5 (as the aggregation of base matroid polytopes is a polymatroid polytope).

Fujishige et al. [27] draw an interesting connection between the convergence of best response dynamics and Rosenthal's potential function. In particular, they show that the fast convergence of best response dynamics for congestion games on extension-parallel networks shown by Fotakis [24] follows from the  $M$ -convexity of the potential function in this case. In particular, these dynamics correspond to a greedy algorithm for minimizing an  $M$ -convex function.

De Jong et al. [17] initiated the study of the price of anarchy of  $k$ -uniform matroid congestion games with affine cost functions. Lücking et al. [36] proved a tight bound of  $\frac{4}{3}$  on the price of anarchy for singleton congestion games ( $k = 1$ ) in this case. De Jong et al. [17] show that the price of anarchy is approximately 1.35188 for symmetric  $k$ -uniform matroid congestion games. In contrast, for the price of stability we provide a tight bound for *general* non-symmetric matroid congestion games with *arbitrary* cost functions.

## 2 Preliminaries

In this section, we introduce all the relevant notions and concepts that will be used subsequently.

## 2.1 Polytopes and describing systems

We review some basic definitions and results from polyhedral combinatorics which are used in this paper (see, e.g., [46] for a more detailed exposition).

A *polytope*  $P \subset \mathbb{R}^m$  is the convex hull of a finite set  $\{q_1, \dots, q_s\} \subset \mathbb{Q}^m$ , or, alternatively,  $P = \{x : Ax \leq b\}$  is a bounded set described by a system of rational inequalities.<sup>3</sup> For a non-zero vector  $c$  with  $\delta = \max\{c^T x : Ax \leq b\}$ , the affine hyperplane  $\{x : c^T x = \delta\}$  is called a *supporting hyperplane* of  $P$ . A subset  $F$  of  $P$  is called a *face* if  $F$  is the intersection of  $P$  with some supporting hyperplane of  $P$ , or  $F = P$ . The *minimal faces* of  $P$ , i.e., faces not contained in another face, are the *vertices (or extreme points)* of  $P$ . Moreover, an *edge* of  $P$  is a one-dimensional face of  $P$  (which is the line-segment between two vertices). We say that  $P$  is *integral* if all its extreme points are integral vectors.  $P$  is said to be *box-integral* if the intersection of  $P$  with any integral box, i.e.,  $P \cap \{x : c \leq x \leq d\}$  for arbitrary integral  $c$  and  $d$ , yields an integral polytope.

A matrix  $A \in \{0, \pm 1\}^{r \times m}$  is *totally unimodular (TUM)* if the determinant of each square submatrix of  $A$  is in  $\{0, \pm 1\}$ . If  $A$  is totally unimodular and  $b \in \mathbb{Z}^m$  is an integer vector, then the polyhedron  $P = \{x : Ax \leq b\}$  is integral [46, Theorem 19.1].

The work [19] introduced the powerful notion of total dual integrality. A rational system  $Ax \leq b$  with  $A \in \mathbb{Q}^{r \times m}$  and  $b \in \mathbb{Q}^r$  is *totally dual integral (TDI)* if for every integral  $c \in \mathbb{Z}^m$ , the dual of minimizing  $c^T x$  over  $Ax \leq b$ , i.e.,

$$\max\{y^T b : y \geq 0, y^T A = c^T\}, \tag{2.1}$$

has an integer optimum solution  $y$ , if it is finite. If  $Ax \leq b$  is a TDI-system and  $b$  is integral, then the polyhedron  $P = \{x : Ax \leq b\}$  is integral [46, Corollary 22.1c]. Note that TDI is a weaker sufficient condition for the integrality of  $P$  than TUM.

The system  $Ax \leq b$  is *box-totally dual integral (box-TDI)* if the system  $Ax \leq b, l \leq x \leq u$  is TDI for all rational vectors  $l$  and  $u$ . We say that a polytope  $P$  is *box-TDI*, if it can be described by a box-TDI system. If  $P$  has some box-TDI describing system, then every TDI-system describing  $P$  is also box-TDI [46, Theorem 22.8].

The main reason as to why box-TDI is useful, is that it serves as a sufficient condition to show that the polytope it describes is box-integral.

**Proposition 2.1** *If the system  $Ax \leq b$  describing a polytope  $P$  is box-TDI and  $b$  is integral, then  $P$  is box-integral.*

**Proof** By assumption, the describing system  $Ax \leq b$  of  $P$  is box-TDI. Thus the system  $Ax \leq b, l \leq y \leq u$  is TDI for all rational vectors  $l$  and  $u$ . In particular,  $Ax \leq b, c \leq y \leq d$  is TDI for arbitrary integral vectors  $c$  and  $d$ . Because  $b, c$  and  $d$  are integral, we can conclude that the polytope  $P \cap \{y : c \leq y \leq d\}$  is integral (see, e.g., [46, Corollary 22.1c]).  $\square$

We will use the following properties of box-TDI descriptions:

**Proposition 2.2** [46, Section 22.5] *The following statements are equivalent:*

<sup>3</sup> An inequality  $a^T x \leq b$  is rational if  $a \in \mathbb{Q}^m$  and  $b \in \mathbb{Q}$ .



1. The system  $Ax \leq b, x \geq 0$  is box-TDI.
2. The system  $Ax + \mu = b, \mu \geq 0, x \geq 0$  is box-TDI.
3. The system  $Ax \leq \alpha b, x \geq 0$  is box-TDI for all  $\alpha \geq 0$ .
4. The system  $a\zeta_0 + Ax \leq b$  is box-TDI, where  $a$  is a column of  $A$  and  $\zeta_0$  is a new variable.

Moreover, if a polytope  $P$  is box-integral, then every edge of  $P$  is in the direction of a  $\{0, \pm 1\}$ -vector.<sup>4</sup>

Finally, let  $N = [n]$ <sup>5</sup> and consider a finite collection of integral polytopes  $(P_i)_{i \in N}$  with a common constraint matrix but possibly different right-hand side vectors, i.e., there exists a matrix  $A$  such that for every  $i \in N$ ,

$$P_i = \{x_i : Ax_i \leq b_i\} \subseteq \mathbb{R}^m.$$

We define the aggregation polytope  $P_N$  induced by  $(P_i)_{i \in N}$  as

$$P_N = \left\{ y : Ay \leq \sum_{i \in N} b_i \right\} \subseteq \mathbb{R}^m.$$

The aggregation polytope is said to have the integer decomposition property (IDP) if every integral  $z \in P_N$  can be written as

$$z = \sum_{i \in N} z_i, \quad \text{where } z_i \in P_i \cap \mathbb{Z}^m \text{ for all } i \in N.$$

Note that in the symmetric case  $b_i = b_j$  for all  $i, j \in N$  this definition reduces to the integer decomposition property for a polytope  $P_N = nP$  as introduced in [6]. Moreover, if  $P_N$  has the IDP, then indeed  $P_N = \sum_i P_i$ , where the latter summation is the Minkowski sum of polytopes.

It seems that most 0/1-polytopes for which the integer decomposition property is known in the literature, also have a box-TDI describing system. We are not aware of any result showing that this is true in general, but it would imply that box-TDI, as an assumption, is redundant in all our statements in later sections.

## 2.2 Optimization over polytopes

We discuss some classical results regarding the problem of optimizing a linear function over a polytope, i.e., we consider the problem

$$\min c^T x \quad \text{s.t. } x \in P \tag{2.2}$$

for some  $c \in \mathbb{Q}^m$  and polytope  $P \subset \mathbb{R}^m$ . We first introduce some additional (computational) notions [31].

<sup>4</sup> That is, the edge is of the form  $\{a + \lambda b : 0 \leq \lambda \leq 1\}$  for vertices  $a, a + b$  of  $P$  where  $b$  is a  $\{0, \pm 1\}$ -vector.

<sup>5</sup> Throughout this paper, for any positive integer  $k$  we define  $[k] = \{1, \dots, k\}$ .

The *encoding length of an integer*  $z \in \mathbb{Z}$ , i.e., the space needed to represent  $z$  in binary representation, is

$$\langle z \rangle = 1 + \lceil \log_2(z + 1) \rceil.$$

The *encoding length of a rational number*  $p/q \in \mathbb{Q}$  is  $1 + \lceil \log_2(p+1) \rceil + \lceil \log_2(q+1) \rceil$ . The *encoding length of a vector*  $a \in \mathbb{Q}^m$  is  $\langle a \rangle = \sum_{i=1}^m \langle a_i \rangle$ , and the *encoding length of an inequality*  $a^\top x \leq b$  is  $\langle a \rangle + \langle b \rangle$ , for  $a \in \mathbb{Q}^m$  and  $b \in \mathbb{Q}$ .

For a positive integer  $\phi$ , we say that polytope  $P$  has *facet-complexity at most  $\phi$*  if there exists a system of inequalities with rational coefficients describing  $P$  such that every inequality has encoding length at most  $\phi$ . A triple  $(P; m, \phi)$  is called a *well-described polytope* if the polytope  $P \subset \mathbb{R}^m$  has facet-complexity at most  $\phi$ .

Finally, a (*strong*) *separation oracle for  $P$*  is an algorithm that, given a vector  $y \in \mathbb{Q}^m$ , decides whether  $y \in P$  or not, and in the latter case returns a vector  $a \in \mathbb{Q}^m$  such that  $a^\top x < a^\top y$  for all  $x \in P$ . If a separation oracle is used as a subroutine in an algorithm, this is referred to as a *call* to the oracle.

The following theorem summarizes a fundamental result in [25,31]. We give a formulation in terms of polytopes based on Theorem 6.6.5 in [31].

**Theorem 2.1** *There exists an algorithm that, for any well-described polytope  $(P; m, \phi)$  specified by a strong separation oracle, and for any given  $c \in \mathbb{Q}^m$ ,*

- (i) *solves (2.2), and*
- (ii) *finds an optimum vertex solution of (2.2) if one exists.*

*The number of elementary arithmetic operations<sup>6</sup> and calls of the separation oracle executed by the algorithm is bounded by a polynomial in  $\phi$ . All arithmetic operations are performed on numbers whose encoding length is bounded by a polynomial in  $\phi + \langle c \rangle$ .*

We give two remarks related to Theorem 2.1.

**Remark 2.1** For notational convenience, we use  $\text{poly}(\cdot)$  to denote a function that is polynomial in all its arguments. The algorithm of Theorem 2.1 runs in *strongly polynomial time* (for a class of problems) if the facet-complexity  $\phi$  can be upper bounded by a polynomial in  $m$ , i.e.,  $\phi = \text{poly}(m)$ .

**Remark 2.2** We do not always explicitly mention that all arithmetic operations are performed on numbers whose encoding length is bounded by a polynomial in  $\phi + \langle c \rangle$ . In all subsequent computational statements relying on Theorem 2.1, we implicitly assume that this property holds.

### 2.3 Congestion games and Rosenthal's potential

A congestion game  $\Gamma$  is given by a tuple  $(N, E, (\mathcal{S}_i)_{i \in N}, (c_e)_{e \in E})$ , where  $N = [n]$  is a finite set of players,  $E = [m]$  is a finite set of resources (or facilities),  $\mathcal{S}_i \subseteq 2^E$  is

<sup>6</sup> Here by *elementary arithmetic operations* we mean addition, subtraction, multiplication, division and comparison.

a set of strategies of player  $i \in N$ , and  $c_e : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$  is a cost function of resource  $e \in E$ . Unless stated otherwise, the cost functions are assumed to be non-negative and non-decreasing.

For a strategy profile  $s = (s_1, \dots, s_n) \in \times_i \mathcal{S}_i$ , we define  $x_e(s)$  as the number of players using resource  $e$ , i.e.,  $x_e(s) = |\{i \in N : e \in s_i\}|$ . We call  $x(s) = (x_e(s))_{e \in E}$  the *load profile corresponding to strategy profile  $s$* . More generally, we say that  $y \in \mathbb{N}^m$  is a *feasible load profile* for the tuple  $(N, E, (\mathcal{S}_i)_{i \in N})$  if there is some strategy profile  $s$  such that  $y = x(s)$ .

The cost of player  $i \in N$  under a strategy profile  $s = (s_1, \dots, s_n) \in \times_i \mathcal{S}_i$  is given by  $C_i(s) = \sum_{e \in s_i} c_e(x_e(s))$ . If  $\mathcal{S}_i = \mathcal{S}_j$  for all  $i, j \in N$ , the game is called *symmetric*. The *social cost*  $C(s)$  of a strategy profile refers to the sum of the players' individual costs, i.e.,

$$C(s) = \sum_{i \in N} C_i(s) = \sum_{e \in E} x_e(s) c_e(x_e(s)).$$

We say that  $\Phi : \times_i \mathcal{S}_i \rightarrow \mathbb{R}$  is an *exact potential function* for a congestion game  $\Gamma$  if for every strategy profile  $s \in \times_i \mathcal{S}_i$ , for every player  $i \in N$  and every unilateral deviation  $s'_i \in \mathcal{S}_i$  of  $i$  it holds that:

$$\Phi(s) - \Phi(s_{-i}, s'_i) = C_i(s) - C_i(s_{-i}, s'_i).$$

Rosenthal [43] shows that

$$\Phi(s) = \sum_{e \in E} \sum_{k=1}^{x_e(s)} c_e(k)$$

is an exact potential function. Subsequently, we refer to this potential function simply as *Rosenthal's potential*. Further, a strategy profile minimizing Rosenthal's potential is said to be a *Rosenthal minimizer*.

### 2.4 Polytopal congestion games

A *polytopal congestion game*  $\Gamma = (N, E, (\mathcal{S}_i)_{i \in N}, (c_e)_{e \in E})$  is a congestion game where the set of strategies  $\mathcal{S}_i$  of each player  $i \in N$  is given implicitly by a polytopal representation. More precisely, let  $\mathcal{X}_i$  be the finite set of all incidence vectors of the strategies of player  $i$ , i.e., for every  $i \in N$ ,

$$\mathcal{X}_i = \{\chi_i \in \{0, 1\}^m : \chi_{ie} = 1 \text{ iff } e \in s_i \text{ for } s_i \in \mathcal{S}_i\}.$$

The polytope  $P_i$  representing the strategies of player  $i$  is defined as the convex hull of  $\mathcal{X}_i$ , i.e.,  $P_i = \text{conv}(\mathcal{X}_i) \subseteq [0, 1]^m$ . We assume that  $P_i$  is given by

$$P_i = \{x_i : Ax_i \leq b_i\} \subseteq [0, 1]^m,$$

where  $A \in \mathbb{Z}^{r \times m}$  is an integral  $r \times m$ -matrix and  $b_i \in \mathbb{Z}^r$  is an integral vector. Note that  $\mathcal{X}_i = P_i \cap \{0, 1\}^m$ . For notational convenience, we subsequently use  $\mathcal{S}_i$  also to refer to the set of incidence vectors  $\mathcal{X}_i$ ; no confusion shall arise.

As defined above, the aggregation polytope induced by  $(P_i)_{i \in N}$  is

$$P_N = \left\{ y : Ay \leq \sum_{i \in N} b_i \right\} \subseteq [0, n]^m.$$

We say that  $(N, E, (\mathcal{S}_i)_{i \in N})$  is the *polytopal tuple* given by  $(P_i)_{i \in N}$ , where  $\mathcal{S}_i = P_i \cap \{0, 1\}^m$ . If  $b_i = b_j = b$  for all  $i, j \in N$ , the tuple is called *symmetric* and denoted by  $(N, E, \mathcal{S})$  where  $\mathcal{S} = P \cap \{0, 1\}^m$ , with  $P = \{x : Ax \leq b\}$ . If additionally we equip the tuple with cost functions  $(c_e)_{e \in E}$ , we call  $\Gamma = (N, E, (\mathcal{S}_i)_{i \in N}, (c_e)_{e \in E})$  the *polytopal congestion game* given by  $(P_i)_{i \in N}$ . For notational convenience, we often omit the explicit reference of the domain of the indices.

### Two main properties: IDP and box-TDI

Let  $\Gamma = (N, E, (\mathcal{S}_i), (c_e))$  be a polytopal congestion game with aggregation polytope  $P_N$ . We identify two crucial properties that the aggregation polytope  $P_N$  has to satisfy for our results to go through:

- (i)  $P_N$  has the integer decomposition property (IDP).
- (ii)  $P_N$  is box-totally dual integral (box-TDI).

If the aggregation polytope  $P_N$  satisfies both the IDP and box-TDI property, we also say that  $\Gamma$  is a *polytopal congestion game satisfying IDP and box-TDI*.

**Remark 2.3** Note that for a *symmetric* polytopal congestion games  $\Gamma = (N, E, \mathcal{S}, (c_e))$  given by a common polytope  $P$ , we have  $P_N = nP = \{y : y/n \in P\}$ . From Proposition 2.2(iii), it follows that the aggregation polytope  $P_N$  has a box-TDI description if and only if  $P$  has a box-TDI description. In particular, whenever we require below that a symmetric polytopal congestion game is box-TDI, then all we need is that the common polytope  $P$  is box-TDI.

The IDP is crucial to establish a correspondence between feasible load profiles for  $(N, E, (\mathcal{S}_i))$  and the integral vectors in  $P_N$ .

**Proposition 2.3** *If the aggregation polytope  $P_N$  of a polytopal tuple  $(N, E, (\mathcal{S}_i))$  has the IDP, then the feasible load profiles of the tuple correspond precisely to the integral vectors in  $P_N$ .*

**Proof** Let  $s = (s_1, \dots, s_n) \in \times_i \mathcal{S}_i$  be a strategy profile and let  $x$  be the load profile corresponding to  $s$ . It follows directly that  $x \in P_N$  by definition of  $P_N$ . Moreover, because of the IDP any integral vector  $z$  in  $P_N$  can be decomposed as  $z = \sum_{i \in N} z_i$  where  $z_i \in P_i \cap \mathbb{Z}^m$  for all  $i \in N$ . This implies that for every  $i$  the vector  $z_i$  is the incidence vector of some strategy of player  $i$  and thus  $z$  is a feasible load profile.  $\square$

### 2.5 Inefficiency of equilibria and the smoothness parameter

A strategy profile  $s$  is a *pure Nash equilibrium* if for every player  $i \in N$  it holds that  $C_i(s) \leq C_i(s'_i, s_{-i})$  for all  $s'_i \in \mathcal{S}_i$ . Further, a strategy profile  $s$  is a *strong equilibrium* if for every group of players  $I \subseteq N$  and every deviation  $s'_I \in \times_{i \in I} \mathcal{S}_i$  of the players in  $I$ , it holds that  $C_i(s) \leq C_i(s'_I, s_{-I})$  for some  $i \in I$ .

The price of anarchy (POA) and the price of stability (POS) of a game  $\Gamma$  are defined as

$$\text{POA}(\Gamma) = \frac{\max_{s \in \text{NE}(\Gamma)} C(s)}{\min_{s^* \in \times_i \mathcal{S}_i} C(s^*)} \quad \text{and} \quad \text{POS}(\Gamma) = \frac{\min_{s \in \text{NE}(\Gamma)} C(s)}{\min_{s^* \in \times_i \mathcal{S}_i} C(s^*)},$$

where  $\text{NE}(\Gamma)$  denotes the set of all pure Nash equilibria of  $\Gamma$ . For a collection of games  $\mathcal{H}$  we define

$$\text{POA}(\mathcal{H}) = \sup_{\Gamma \in \mathcal{H}} \text{POA}(\Gamma) \quad \text{and} \quad \text{POS}(\mathcal{H}) = \sup_{\Gamma \in \mathcal{H}} \text{POS}(\Gamma).$$

These notions naturally generalize to the solution concept of strong equilibria. A *non-atomic congestion game* (see, e.g., [41]) can be seen as the continuous counterpart of Rosenthal’s congestion game model. Here, there is a continuum of players that all control an infinitesimally small amount of flow (as opposed to one unit in Rosenthal’s model). The price of anarchy of these games has been studied intensively and is well-understood. In particular, it turns out that the price of anarchy is completely determined by the class of cost functions being used.

Correa et al. [13] show that for non-atomic *network* congestion games with cost functions in class  $\mathcal{D}$  the price of anarchy of an instance is at most

$$\rho(\mathcal{D}) := (1 - \beta(\mathcal{D}))^{-1}, \quad \text{where} \quad \beta(\mathcal{D}) = \sup_{d \in \mathcal{D}} \sup_{x \geq y > 0} \frac{y(d(x) - d(y))}{xd(x)}. \quad (2.3)$$

The value of  $\rho(\mathcal{D})$  is well-understood for many important classes of cost functions. For example, let

$$\mathcal{D}_d = \{g : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0} : g(\mu x) \geq \mu^d g(x) \ \forall \mu \in [0, 1]\}.$$

In particular,  $\mathcal{D}_d$  contains all polynomial cost functions with non-negative coefficients and maximum degree  $d$ . We have

$$\rho(\mathcal{D}_d) = \left(1 - \frac{d}{(d+1)^{(d+1)/d}}\right)^{-1}.$$

The parameter  $\rho(\mathcal{D})$  plays a crucial role in bounding the price of stability of our congestion games.

We say that a class of cost functions  $\mathcal{D}$  is *closed under dilations* if for every  $d \in \mathcal{D}$  and for every  $\gamma \in \mathbb{R}_{\geq 0}$ ,  $d_\gamma \in \mathcal{D}$ , where  $d_\gamma(x) = d(\gamma x)$  for all  $x \geq 0$ .

## 2.6 Matroids

We introduce some general terminology and facts for matroids; a more extensive treatment of matroids can be found, e.g., in [47]. Let  $E = [m]$  be a finite set of elements and  $\mathcal{I} \subseteq 2^E$  be a collection of subsets of  $E$  (called *independent sets*). The pair  $\mathcal{M} = (E, \mathcal{I})$  is a *matroid* if the following three properties hold:

- (i)  $\emptyset \in \mathcal{I}$ ,
- (ii) if  $A \in \mathcal{I}$  and  $B \subseteq A$ , then  $B \in \mathcal{I}$ ,
- (iii) if  $A, B \in \mathcal{I}$  and  $|A| > |B|$ , then there exists an  $a \in A \setminus B$  such that  $B + a \in \mathcal{I}$ .

An independent set  $B \in \mathcal{I}$  of maximum size is called a *basis*. We use  $\mathcal{B}$  to denote the set of all bases of  $\mathcal{M}$ . The matroid  $\mathcal{M}$  also has a rank function  $r : 2^E \rightarrow [m]$  which maps every subset  $A \subseteq E$  to the cardinality of the largest independent set contained in  $A$ .

The *base matroid polytope* is given by

$$P_{\mathcal{M}} = \{x : x(A) \leq r(A) \forall A \subseteq E, x(E) = r(E), x \geq 0\},$$

where  $x(A) = \sum_{a \in A} x_a$  for all  $A \subseteq E$ . It is the convex hull of the incidence vectors of the bases in  $\mathcal{B}$  [47]. If in the description above the equality  $x(E) = r(E)$  is replaced by  $x(E) \leq r(E)$ , we obtain the *independent set polytope* which is the convex hull of the incidence vectors of the independent sets.

We assume that the matroid is given by an *independence oracle* that takes as input a subset  $A \subseteq 2^E$  and returns whether or not  $A \in \mathcal{I}$ . Given an independence oracle, we can determine in time polynomial in  $|E|$  and the complexity of the oracle, whether a set is a basis and what the rank of a set is. Further, there exists a separation oracle for  $P_{\mathcal{M}}$  that runs in time polynomial in  $|E|$  and the complexity of an independence oracle. This follows from the fact that the *most violated inequality problem* can be solved in time polynomial in  $|E|$  and the complexity of an independence oracle. The most violated inequality problem takes as input a vector  $x \in \mathbb{Q}^m$  and returns whether or not  $x \in P$ , and if not, it returns a subset  $A$  for which  $r(A) - x(A)$  is minimized, see, e.g., [47, Section 40.3].

Given two matroids  $\mathcal{M}_1$  and  $\mathcal{M}_2$  on a common ground set  $E$ , the polytope

$$P_{\mathcal{M}_1, \mathcal{M}_2} = \{x : x(A) \leq r_i(A) \forall A \subseteq E, x(E) = r_i(E) \text{ for } i = 1, 2, x \geq 0\} \quad (2.4)$$

is the convex hull of the common bases of matroids  $\mathcal{M}_1$  and  $\mathcal{M}_2$ , see, e.g., [47, Corollary 41.12d]. It follows directly that  $P_{\mathcal{M}_1, \mathcal{M}_2}$  also has a separation oracle which runs in time polynomial in  $|E|$  and the complexity of the independence oracles for  $\mathcal{M}_1$  and  $\mathcal{M}_2$ .

## 3 Separable convex function minimization

In this section, we consider the problem of minimizing a separable convex function over a joint set of polytopes. We prove that this problem can be solved if the aggregation

polytope of the joint set of polytopes has the IDP and box-TDI property. As we will show, this problem is equivalent to finding a minimizer of Rosenthal’s potential.

### 3.1 General framework

Let  $n, m, r \in \mathbb{N}$  be fixed and define  $N = [n]$ . Let  $f_e : \mathbb{N} \rightarrow \mathbb{R}$  be a convex function for every  $e \in [m]$ ,  $A \in \{0, \pm 1\}^{r \times m}$  a matrix and  $b_i \in \mathbb{Z}^r$  a vector for every  $i \in N$ . Further, define  $P_i = \{x_i : Ax_i \leq b_i\} \subseteq [0, 1]^m$  for every  $i \in N$ . The elements of the vector  $x_i$  are indexed by  $x_{i,e}$  for  $e \in E$ .

We consider the following problem:

$$\begin{aligned}
 (P) \quad & \min \sum_{e \in [m]} f_e(x_e) \\
 & \text{s.t. } Ax_i \leq b_i \quad \forall i \in N \\
 & \quad x_e = \sum_{i \in [n]} x_{i,e} \\
 & \quad x_i \in \{0, 1\}^m \quad \forall i \in N.
 \end{aligned}$$

In order to solve this problem, we will use the aggregation/decomposition framework of Del Pia et al. [16]. In particular, if we assume that the aggregation polytope  $P_N = \sum_i P_i$  has the integer decomposition property, we can use a two-step approach in which we first compute a solution minimizing the aggregation problem

$$(Z) \quad \min \sum_{e \in [m]} f_e(y_e) \quad \text{s.t. } Ay \leq \sum_{i \in N} b_i, \quad y \in \mathbb{N}^m,$$

and then decompose the resulting solution into a solution for  $(P)$  (which is possible because of the IDP property).

Throughout this section, we assume that the aggregation polytope  $P_N$  is a well-described polytope  $(P_N; m, \phi)$  (as defined in Sect. 2.2) for which we have a separation oracle. Further, all operations are performed on numbers whose encoding length is bounded by a polynomial in  $\phi + \langle c \rangle$  (see Remark 2.2).

**Theorem 3.1** *Let  $(P_i)_{i \in N}$  be as defined above and suppose that the aggregation polytope  $P_N = \sum_i P_i$  is a well-described polytope that has a box-TDI description. Then  $(Z)$  can be solved optimally using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls.*

In particular, note that Theorem 3.1 implies that we can solve  $(P)$  efficiently if we can decompose a solution of  $(Z)$  into a solution for  $(P)$  using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls.

We prove Theorem 3.1 in the remainder of this section.

By extending  $f_e$  for every  $e \in [m]$  to a convex function that is also defined at zero,<sup>7</sup> we can write for every positive  $y_e \in \mathbb{N}$ ,

$$f_e(y_e) = f_e(0) + \sum_{k=1}^{y_e} (f_e(k) - f_e(k - 1)).$$

Because  $f_e$  is convex, the function  $c_e : \mathbb{N} \rightarrow \mathbb{R}$  defined by  $c_e(k) = f_e(k) - f_e(k - 1)$  is non-decreasing, and, in fact, solving (Z) is equivalent (by leaving out the constant-valued sum over the  $f_e(0)$ ) to solving

$$(Z') \quad \min \sum_{e \in [m]} \sum_{k=1}^{y_e} c_e(k) \quad \text{s.t.} \quad Ay \leq \sum_{i \in N} b_i, \quad y \in \mathbb{N}^m.$$

This is a well-known reduction (see, e.g., [33] and references therein). We next show (using a column-doubling trick similar as in [15,23]) how to reduce (Z') to a linear program. This idea is (implicitly) also exploited in the works [16,21]. Here we use the same idea for our (more general) setting.

We introduce binary variables  $h_e^k \in \{0, 1\}$  for every  $k \in [n]$  and  $e \in [m]$ . The interpretation is that  $h_e^k = 1$  if and only if the value of  $x_e$  in (P) is at least  $k$ . Exploiting that the functions  $c_e$  are non-decreasing, the non-linear aggregation problem (Z') is equivalent to the problem (R) stated below:

$$(R) \quad \min \sum_{e \in [m]} \sum_{k \in [n]} c_e(k) h_e^k$$

$$\text{s.t.} \quad [A, A, \dots, A] (h_1^1, \dots, h_m^1, \dots, h_1^n, \dots, h_m^n)^T \leq \sum_{i \in N} b_i \quad (3.1)$$

$$h_e^k \in \{0, 1\} \quad \forall k \in [n], \forall e \in [m] \quad (3.2)$$

The equivalence of (Z) and (R) follows from the following observations: If  $f = (f_e) \in P_N \cap \mathbb{N}^m$  is optimal for (Z'), we define for every  $e \in [m]$ ,  $h_e^k = 1$  for  $k = 1, \dots, f_e$  and  $h_e^k = 0$  for  $k = f_e + 1, \dots, n$ . The resulting solution  $h = (h_e^k)$  is feasible for (R). Similarly, if  $h = (h_e^k)$  is an optimal solution for (R), then the vector  $f$  defined by  $f_e = \sum_{k=1}^n h_e^k$  is feasible for (Z'). Note that here we implicitly exploit that the functions  $c_e$  are non-decreasing.

We show that the integer program (R) can be solved efficiently.

**Lemma 3.1** *If  $P_N$  has a box-TDI description, then (R) can be solved using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls.*

**Proof** Define  $A' = [A, A, \dots, A] \in \mathbb{Z}^{r \times mn}$  and  $h = (h_e^k) \in \mathbb{Q}^{mn}$ . The relaxation of the integral system (3.1) and (3.2) can then be written as the system  $A'h \leq \sum_i b_i$ ,  $0 \leq h \leq 1$ . Let  $Q_N = \{h : A'h \leq \sum_i b_i, 0 \leq h \leq 1\}$  be the polytope described by this system.

<sup>7</sup> It is not hard to see that this can always be done.



We first show that  $Q_N$  is integral. By assumption the description of  $P_N = \{f : Af \leq \sum_i b_i\}$  is box-TDI. In particular, by applying Proposition 2.2(iv) repeatedly, we obtain that the system

$$[A, A, \dots, A](h_1^1, \dots, h_m^1, \dots, h_1^n, \dots, h_m^n)^T \leq \sum_i b_i$$

is box-TDI as well. In particular, this implies that the system  $A'h \leq \sum_i b_i, 0 \leq h \leq 1$  is TDI because the intersection of a box-TDI system with an arbitrary box yields a TDI system. Because  $\sum_i b_i$  and the restrictions on  $h$  are integral vectors, we conclude that  $Q_N$  is indeed integral.

We now show how to construct a separation oracle for  $Q_N$  from a separation oracle for  $P_N$ . For

$$h = (h_1^1, \dots, h_m^1, \dots, h_1^n, \dots, h_m^n) \in \mathbb{Q}^{mn},$$

let the aggregated vector  $f \in \mathbb{Q}^m$  be defined as  $f_e = \sum_{k=1}^n h_e^k$  for  $e \in [m]$ . Then  $h \in Q_N$  if and only if  $f \in P_N$ . We now give a separation oracle for  $Q_N$ . Let  $y = (y_e^k) \in \mathbb{Q}^{mn}$  be an arbitrary rational vector and let  $f$  be defined as above with respect to  $y$ . We use the separation oracle of  $P_N$  to check if  $f \in P_N$  or not. If  $f \in P_N$ , then also  $y \in Q_N$  and we are done. Otherwise if  $f \notin P_N$  the oracle returns a vector  $a \in \mathbb{Q}^m$  such that  $a^T x < a^T f$  for all  $x \in P_N$ . In particular this means that  $(a^T, a^T, \dots, a^T)z < (a^T, a^T, \dots, a^T)y$  for all  $z = (z_e^k) \in Q_N$ . Thus, we obtain a separation oracle for  $Q_N$ .

We conclude with an analysis of the running time. It is not hard to see that  $Q_N$  has a facet complexity that is at most a polynomial (in  $m$  and  $n$ ) factor larger than  $\phi$ . The claim that we only need a number of arithmetic operations and calls to a separation oracle for  $P_N$ , that is polynomial in  $n, m$ , and  $\phi$ , now follows immediately from Theorem 2.1. This concludes the proof. □

**Remark 3.1** If the facet-complexity  $\phi$  can be upper bounded by a polynomial in  $n$  and  $m$ , and if  $P_N$  has a separation oracle running in strongly polynomial time, then a solution to  $(Z')$  can be computed in strongly polynomial time (see also Remark 2.1).

Although we can solve  $(R)$  in (strongly) polynomial time as stated in Lemma 3.1, this does not directly imply that we can also solve  $(P)$  in (strongly) polynomial time. The IDP property guarantees that we can always decompose a solution to  $(R)$  into a solution of  $(P)$ . However, to the best of our knowledge, there is no universal algorithm that can perform integer decomposition of an arbitrary polytope satisfying the IDP using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls (even in the symmetric case when all right-hand side vectors  $b_i$  are the same). So whether this decomposition can be done efficiently has to be investigated on a case-by-case basis. As we show next, such a decomposition can always be done in polynomial time if a slightly stronger integer decomposition property holds. Here, we focus on the symmetric case when  $b_i = b_j = b$  for all  $i, j \in N$ ; but these arguments can be extended to the non-symmetric case as well.

We say that a polytope  $P$  satisfies the *middle integral decomposition property* (MIDP) [37] if for  $n \in \mathbb{N}$  and  $w \in \mathbb{Z}^m$ , the polytope  $P \cap (w - (n - 1)P)$  is integral.

If this property is satisfied, the decomposition algorithm of Baum and Trotter [6] can be used to perform the integer decomposition (as described in the proof below).

**Theorem 3.2** *If the polytope  $P = \{x : Ax \leq b\}$  satisfies MIDP and box-TDI. Then an optimal solution to  $(P)$  can be computed using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls.*

**Proof** Let  $f$  be a solution to  $(R)$ . Then  $f$  can be decomposed into  $n$  integer solutions in  $P$  by using the following decomposition algorithm by Baum and Trotter [6]: We start by computing an integral vector  $x^1 \in P \cap (f - (n-1)P)$ . By the middle integral decomposition property, we know that  $P \cap (f - (n-1)P)$  is integral, and therefore we can find an integral (extreme) point in time polynomial in  $n, m, \phi$  and the complexity of a separation oracle of  $P$  (using similar arguments as in the proof of Theorem 3.1). Using the same arguments, we can then find a vector  $x_2 \in P \cap ((f - x^1) - (n-2)P)$ . By repeating this procedure, we find the desired decomposition in the stated running time.  $\square$

### 3.2 Minimizing Rosenthal's potential

We use the results obtained in the previous section and apply them to the problem of computing a minimizer of Rosenthal's potential function for a polytopal congestion game  $\Gamma = (N, E, (S_i), (c_e))$ .

In particular, the following theorem follows immediately from Theorem 3.1 by observing that for polytopal congestion games, computing a load profile minimizing Rosenthal's potential is equivalent to the program  $(Z')$  stated above.

**Theorem 3.3** *Let  $\Gamma = (N, E, (S_i), (c_e))$  be a polytopal congestion game whose aggregation polytope  $P_N$  satisfies IDP and box-TDI. Then a feasible load profile minimizing Rosenthal's potential can be computed using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls.*

As noted in Remark 3.1, if the facet-complexity  $\phi$  can be upper bounded by a polynomial in  $n$  and  $m$ , and if  $P_N$  has a separation oracle running in strongly polynomial time, then a feasible load profile minimizing Rosenthal's potential can be computed in strongly polynomial time.

Note that by applying Theorem 3.3 we obtain a feasible *load profile*. We can turn such a load profile into a feasible *strategy profile* (corresponding to a pure Nash equilibrium) if the integer decomposition can be done in (strongly) polynomial time. As noted before, a universal algorithm for this seems elusive and whether this can be done efficiently has to be verified case-by-case.

Also here, if we make the stronger assumption of having the middle integral decomposition property, we obtain the following result for symmetric games.

**Theorem 3.4** *Let  $\Gamma = (N, E, S, (c_e))$  be a symmetric polytopal congestion game for which the common polytope  $P$  satisfies MIDP and box-TDI. Then a feasible strategy profile minimizing Rosenthal's potential can be computed using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls.*

We remark that all results in this section also hold for computing a social optimum of congestion games with *weakly convex* cost functions, since this problem can be reduced to computing a global optimum of Rosenthal’s potential (we refer to [16] for more details).

### 4 Price of stability

In this section, we will analyze the quality of a minimizer of Rosenthal’s potential compared to a socially optimal outcome in terms of the price of stability.

Recall that  $\rho(\mathcal{D})$  is defined as in (2.3) and refers to the price of anarchy of *non-atomic* network congestion games with cost function in class  $\mathcal{D}$ . The following is the main result of this section.

**Theorem 4.1** *Let  $\Gamma = (N, E, (\mathcal{S}_i), (c_e))$  be a polytopal congestion game satisfying IDP and box-TDI with cost functions in class  $\mathcal{D}$ . Then  $POS(\Gamma) \leq \rho(\mathcal{D})$ . Further, this bound is (asymptotically) tight even for symmetric singleton congestion games, if  $\mathcal{D}$  contains all non-negative constant functions and is closed under dilations.*

The remainder of this section is devoted to the proof of Theorem 4.1. We first introduce a novel structural property, which we term the *symmetric difference decomposition property*. We then show that the IDP and box-TDI properties of the aggregation polytope are sufficient to establish that the polytopal congestion game satisfies the symmetric difference decomposition property. This in turn allows us to adapt the proof of Fotakis [24] to bound the price of stability of these games.

#### 4.1 Symmetric difference decomposition property

Our novel property is defined as follows:

**Definition 4.1** A tuple  $(N, E, (\mathcal{S}_i))$  satisfies the *symmetric difference decomposition property (SDD)* if for all feasible load profiles  $f$  and  $g$ , there exist vectors  $a^1, \dots, a^q \in \{0, \pm 1\}^m$  such that  $g - f = \sum_{k=1}^q a^k$ , and for all  $k = 1, \dots, q$ ,

- (i) the load profile  $f + a^k$  is feasible, and
- (ii)  $a^k$  satisfies

$$a_e^k = -1 \Rightarrow f_e - g_e > 0 \quad \text{and} \quad a_e^k = 1 \Rightarrow f_e - g_e < 0. \quad (4.1)$$

As an example, let us consider symmetric network congestion games, where the common strategy set of all players is the set of all directed simple  $s, t$ -paths in some directed graph  $G = (V, A)$  with  $s, t \in V$ . Here each feasible load profile corresponds to an integral feasible  $s, t$ -flow of value  $n = |N|$ . The symmetric difference of two flows  $f$  and  $g$  can be written as the sum of unit circuit flows on cycles.<sup>8</sup> The incidence vectors of these unit circuit flows correspond to the vectors  $a^k$  in Definition 4.1.

The following theorem establishes the symmetric difference decomposition property for polytopal congestion games satisfying IDP and box-TDI.

<sup>8</sup> A *unit circuit flow* is a  $\{0, \pm 1\}$ -flow that satisfies flow-conservation at every node, including  $s$  and  $t$ .

**Theorem 4.2** *If the aggregation polytope  $P_N$  of a polytopal tuple  $(N, E, (S_i))$  satisfies IDP and box-TDI, then the tuple has the symmetric difference decomposition property.*

**Proof** We start by adding slack-variables to the system  $Ay \leq \sum_{i \in N} b_i$  describing  $P_N$ . Note that by Proposition 2.2(ii) box-TDI is preserved under adding slack variables. As a result, we obtain the polytope

$$Q_N = \left\{ (y, \mu) : Ay + \mu = \sum_{i=1}^n b_i, \mu \geq 0, y \geq 0 \right\}$$

for which its describing system is box-TDI. Also,  $Q_N$  is integral.

Let  $f$  and  $g$  be two feasible load profiles with  $f \neq g$ . By Proposition 2.3, we have  $f, g \in P_N$ . Therefore, there are non-negative integral slack vectors  $\tau, \sigma$  such that  $f' = (f, \tau), g' = (g, \sigma) \in Q_N$ . Observe that  $\tau$  and  $\sigma$  are integral because of the integrality of  $A, \sum_i b_i$  and  $f$  and  $g$ .

Note that the pairs  $f' = (f, \tau)$  and  $g' = (g, \sigma)$  are vectors in  $\mathbb{Z}^{m+r}$  since  $A$  is an  $r \times m$ -matrix. Let  $c, d \in \mathbb{Z}^{m+r}$  be vectors defined by  $c_j = \min\{f'_j, g'_j\}$  and  $d_j = \max\{f'_j, g'_j\}$  for  $j = 1, \dots, r + m$ , and let  $B$  be the integral box defined by  $B = \{z : c \leq z \leq d\} \subseteq \mathbb{R}^{m+r}$ . We first prove the following claim.

**Claim** The polytope  $Q_N \cap B$  is integral and every edge of  $Q_N \cap B$  is in the direction of a  $\{0, \pm 1\}$ -vector.

**Proof** The integrality follows from box-TDI of the integral system  $Q_N$ . For the second part of the claim, we first show that  $Q_N \cap B$  is box-integral. Note that  $Q_N$  is box-integral by Proposition 2.1. Let  $B' = \{x : \gamma \leq x \leq \delta\} \in \mathbb{Z}^{m+r}$  be an arbitrary integral box. Note that  $(Q_N \cap B) \cap B' = Q_N \cap (B \cap B')$  and that  $B \cap B'$  is again an integral box, since  $B$  is integral as well (because  $f'$  and  $g'$  are integral). It follows that  $Q_N \cap (B \cap B')$  is an integral polytope. Thus,  $(Q_N \cap B) \cap B'$  is integral which proves that  $Q_N \cap B$  is box-integral. The claim now follows from Proposition 2.2.  $\square$

Note that  $f', g' \in Q_N \cap B$ . Further, both  $f'$  and  $g'$  are extreme points of this polytope because they are extreme points of the box  $B$ . We now fix some edge of  $Q_N \cap B$  containing  $f'$ . Such an edge must exist because  $Q_N \cap B$  contains at least two elements (since  $f' \neq g'$ ). Let  $(a^1)' = (a^1, \mu^1)$  be the non-zero  $\{0, \pm 1\}$ -vector describing the direction of the edge.<sup>9</sup> Since  $Q_N \cap B$  is an integral polytope, we can show that  $f' + (a^1)' \in Q_N \cap B$ . To see this, let  $h(\lambda) = f' + \lambda \cdot (a^1)'$  be a parametrization of the edge for some range  $0 \leq \lambda \leq \lambda^*$ , where  $h' = h(\lambda^*)$  is the other extreme point of the edge  $(a^1)'$ . Since  $f'$  is integral and  $(a^1)'$  a  $\{0, \pm 1\}$ -vector, it must be that  $\lambda^*$  is a strictly positive integer. Thus,  $f' + (a^1)' \in Q_N \cap B$ , as claimed.

It follows that  $A(f + a^1) + (\tau + \mu^1) = \sum_i b_i$ . We have  $Aa^1 + \mu^1 = 0$  because  $Af + \tau = \sum_i b_i$ . Moreover, by construction of the box  $B$  it follows that for  $j = 1, \dots, r + m$ ,

$$(a^1)'_j = -1 \Rightarrow f'_j - g'_j > 0 \quad \text{and} \quad (a^1)'_j = 1 \Rightarrow g'_j - f'_j > 0. \quad (4.2)$$

<sup>9</sup> Here,  $\mu^1$  corresponds to the slack variables, and  $a^1$  to the original variables.

Exploiting that  $Aa^1 + \mu^1 = 0$ , it now also follows that  $g' - (a^1)'_j \in Q_N \cap B$ . To see this, note that

$$A(g - a^1) + (\sigma - \mu^1) = Ag + \sigma - (Aa^1 + \mu^1) = \sum_i b_i.$$

Moreover, we also have  $g' - (a^1)' \geq 0$  by construction, since if  $(a^1)'_j = 1$  for some  $j$  then  $g'_j > f'_j \geq 0$ , so in particular  $g'_j - 1 \geq 0$  (because of the integrality of  $g'_j$ ).

We can now apply the same argument to the vectors  $f'$  and  $g' - (a^1)'$  in order to obtain a vector  $(a^2)'$  satisfying (4.2) and for which  $f' + (a^2)'$ ,  $g' - (a^1)' - (a^2)'$   $\in Q_N$ . Repeating this procedure we find vectors  $(a^1)', \dots, (a^q)'$  satisfying (4.2), and such that  $g' - f' = \sum_{k=1}^q (a^k)'$  with  $f' + (a^k)' \in Q_N$  for  $k = 1, \dots, q$ .<sup>10</sup>

We argue that this process terminates. For the  $\ell$ -th step of this procedure, we have by construction of the  $(a^k)'$

$$T(\ell) = \left\| \left( g' - \sum_{k=1}^{\ell} (a^k)' \right) - f' \right\|_1 < \left\| \left( g' - \sum_{k=1}^{\ell-1} (a^k)' \right) - f' \right\|_1 = T(\ell - 1)$$

where  $\| \cdot \|_1$  is the  $L_1$ -norm. Since  $f'$ ,  $g'$  and the  $a^k$  are all integral this guarantees that the expression  $T(\ell)$  decreases by at least one in every step.

We conclude the proof by showing that  $f$  and  $g$  can be decomposed according to Definition 4.1. We have  $(a^k)' = (a^k, \mu^k)$  as defined before. It then follows that  $a^1, \dots, a^q$  are vectors satisfying (4.1) such that  $g - f = \sum_{k=1}^q a^k$  with  $f + a^k \in P_N$  for  $k = 1, \dots, q$ . Note that  $a^k$  might be the zero-vector, if  $(a^k)'$  only contained non-zero elements in the part of the vector corresponding to slack variables. These  $a^k$  can be left out.

It remains to show that  $f + a^k$  corresponds to a feasible strategy profile for  $k = 1, \dots, q$ . This follows directly from the fact that  $P_N$  has the IDP. The decomposition yields the strategies of the players.  $\square$

### 4.2 Upper bound on the price of stability

We prove the upper bound on the price of stability stated in Theorem 4.1. We first prove the following lemma, whose proof relies on the symmetric difference decomposition property.

**Lemma 4.1** *Let  $(N, E, (S_i))$  be a polytopal tuple that satisfies the symmetric difference decomposition property and let  $(c_e)_{e \in E}$  be arbitrary cost functions. Let  $f$  be a feasible load profile that minimizes Rosenthal's potential  $\Phi(\cdot)$ . Then for every feasible load profile  $g$*

$$\Delta(f, g) := \sum_{e \in E: f_e > g_e} (f_e - g_e)c_e(f_e) - \sum_{e \in E: f_e < g_e} (g_e - f_e)c_e(f_e + 1) \leq 0. \quad (4.3)$$

<sup>10</sup> This construction is essentially a *conformal circuit decomposition* (see, e.g., [42]).

**Proof** Let  $f$  be a global minimizer of Rosenthal’s potential and let  $g$  be an arbitrary feasible load profile. Then by the SDD property, there exist vectors  $a^1, \dots, a^q$  such that  $g - f = \sum_{k=1}^q a^k$  for some  $q$ . Moreover, for all  $k = 1, \dots, q$

$$\Phi(f) - \Phi(f + a^k) = \sum_{e: a_e^k = -1} c_e(f_e) - \sum_{e: a_e^k = 1} c_e(f_e + 1) \leq 0,$$

where the inequality holds because  $f$  minimizes Rosenthal’s potential  $\Phi$ . By adding up these inequalities for all  $k = 1, \dots, q$ , we obtain that  $\Delta(f, g) \leq 0$ . To see this, note that if  $e \in E$  with  $f_e > g_e$  then there are precisely  $f_e - g_e$  vectors  $a^k$  with  $a_e^k = -1$ ; similarly, if  $e \in E$  with  $g_e > f_e$  then there are precisely  $g_e - f_e$  vectors  $a^k$  with  $a_e^k = 1$ . □

We can now prove the upper bound on the price of stability.

**Proof (Theorem 4.1, upper bound)** The upper bound proof follows a similar line of arguments as the proof of Lemma 3 in [24]. We repeat the arguments here for the sake of completeness.

Let  $f$  be a minimizer of Rosenthal’s potential and let  $g$  be an arbitrary feasible load profile. Note that  $f$  is a pure Nash equilibrium.

Consider a resource  $e \in E$  with  $f_e > g_e$ . We have

$$\begin{aligned} f_e c_e(f_e) &= g_e c_e(f_e) + (f_e - g_e) c_e(f_e) \\ &\leq g_e c_e(g_e) + \beta(\mathcal{D}) f_e c_e(f_e) + (f_e - g_e) c_e(f_e), \end{aligned} \tag{4.4}$$

where the inequality follows from the definition of  $\beta(\mathcal{D})$  in (2.3), exploiting that  $f_e > g_e \geq 0$  and  $c_e \in \mathcal{D}$ .

Next, consider a resource  $e \in E$  with  $f_e < g_e$ . We have

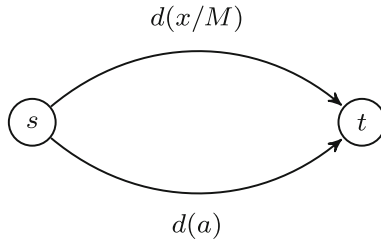
$$\begin{aligned} f_e c_e(f_e) &= g_e c_e(g_e) - g_e c_e(g_e) + f_e c_e(f_e) \\ &\leq g_e c_e(g_e) - (g_e - f_e) c_e(f_e + 1), \end{aligned} \tag{4.5}$$

where the inequality follows because  $c_e$  is non-decreasing and  $f_e + 1 \leq g_e$  by assumption.

Combining these inequalities, we obtain

$$C(f) \leq C(g) + \sum_{e \in E: f_e > g_e} \beta(\mathcal{D}) f_e c_e(f_e) + \Delta(f, g) \leq C(g) + \beta(\mathcal{D}) C(f)$$

where the first inequality follows from (4.4) and (4.5) and the definition of  $\Delta(f, g)$  in (4.3), and the last inequality holds because  $\Delta(f, g) \leq 0$  by Lemma 4.1. By rearranging terms, we obtain  $C(f)/C(g) \leq (1 - \beta(\mathcal{D}))^{-1} = \rho(\mathcal{D})$ , which proves the claim. □



**Fig. 1** The bottom and top arc  $a_1$  and  $a_2$ , respectively, have cost functions  $c_1(x) = d(a)$  and  $c_2(x) = d(x/M)$

### 4.3 Lower bound on the price of stability

We complete the proof of Theorem 4.1 by showing that the stated bound is asymptotically tight.

**Proof (Theorem 4.1, lower bound)** Our lower bound construction is similar to the one used by Correa et al. [14] to show tightness of the price of anarchy bound  $\rho(\mathcal{D})$  for non-atomic network congestion games with cost functions in class  $\mathcal{D}$ . But we need some adjustments to make it work for atomic (unsplittable) congestion games.

Let  $d \in \mathcal{D}$  and  $a \geq b > 0$  be chosen arbitrarily. We show that there exists an instance whose price of stability is arbitrarily close to

$$\left(1 - \frac{b(d(a) - d(b))}{ad(a)}\right)^{-1}. \tag{4.6}$$

Because of the continuity of  $d$ , we can take  $a, b \in \mathbb{Q}$  without loss of generality. Let  $M \in \mathbb{N}$  such that  $Ma, Mb \in \mathbb{N}$ . Consider the instance depicted in Fig. 1 with  $Ma$  players. Note that this is a symmetric singleton congestion game instance. Further, note that  $c_1 \in \mathcal{D}$  because it is a constant function and  $c_2 \in \mathcal{D}$  because  $\mathcal{D}$  is closed under dilations. A Nash equilibrium is given by the flow  $f = (f_1, f_2) = (0, Ma)$ . A feasible (not necessarily socially optimal) flow is given by  $g = (g_1, g_2) = (M(a - b), Mb)$ . We have

$$\frac{C(f)}{C(g)} = \frac{Ma \cdot d(a)}{M(a - b)d(a) + Mb \cdot d(b)} = \left(1 - \frac{b(d(a) - d(b))}{ad(a)}\right)^{-1}.$$

In order to get a lower bound on the price of stability, we make  $f$  the unique Nash flow of this game. This can be done by adding a small enough  $\epsilon > 0$  to the cost function of arc  $a_1$ , i.e., we take  $c_1(x) = d(a) + \epsilon$ . Doing the same analysis and sending  $\epsilon \rightarrow 0$ , then shows that we can get arbitrarily close to the expression in (4.6).  $\square$

## 5 Applications

We now give several examples of polytopal congestion games for which the aggregation polytope has the (middle) integer decomposition property, is box-TDI and admits an efficient separation oracle. As a consequence, our results on the price of stability (Theorem 4.1) and the computation of Rosenthal's potential minimizer (Theorems 3.3, 3.2) apply.

**Remark 5.1** In all applications considered below, the facet complexity  $\phi$  of the well-described (aggregation) polyhedra are polynomially bounded in  $n$  and  $m$ . In particular, all the matrices  $A$  considered are in fact  $\{0, \pm 1\}$ -valued matrices, and the right-hand side vectors  $b_i$  are always integral valued.

### 5.1 Common source network congestion games

In a *common source network congestion game* we are given a directed graph  $G = (V, A)$  and a source  $s \in V$ . The strategy set of player  $i \in N$  is the set of all directed  $s, t_i$ -paths for some  $t_i \in V$ . Ackermann et al. [1] already showed that one can compute a global optimum of Rosenthal's potential function for these games. We outline how this case can be cast in our framework. The strategies of player  $i$  can be described by a polytope  $P_i = \{x : Ax = b_i, 0 \leq x \leq 1\}$ , where  $A$  is the arc-incidence matrix of the network  $G$ , and  $b$  is the vector with  $(b_i)_s = 1$ ,  $(b_i)_{t_i} = -1$  and zero otherwise.<sup>11</sup> The aggregation polytope is then  $P_N = \{y : Ay = \sum_{i \in N} b_i, 0 \leq y \leq n\}$ . Any feasible load profile minimizing Rosenthal's potential can be decomposed efficiently into a feasible strategy profile, using a similar argument as in [1]. Further, the describing system of  $P_N$  is totally unimodular and thus box-TDI.<sup>12</sup>

### 5.2 Symmetric totally unimodular congestion games

*Symmetric totally unimodular congestion games* [16] capture a wide range of combinatorial congestion games. Here the common strategy set of the players is described by a polytope  $P = \{x : Ax \leq b\}$  with a totally unimodular  $r \times m$ -matrix  $A$  and an integral vector  $b$ . In particular, such a system satisfies the IDP and is box-TDI: The integer decomposition property was shown in [6]. We argue that the system is box-TDI. The constraint matrix describing the intersection of  $P$  with  $\{x : c \leq x \leq d\}$  for  $c, d \in \mathbb{Q}^m$  is again totally unimodular [48]. Any totally unimodular system is TDI (see, e.g., [46, Section 22.1]), and therefore the system  $Ax \leq b, c \leq x \leq d$  is TDI. We conclude that the system  $Ax \leq b$  is box-TDI. If (as in [16]) the parameter  $r$  is considered as part of the input size as well, then there is a trivial (strongly) polynomial separation oracle that simply checks all inequalities of the system  $Ax \leq b$ . For all combinatorial applications considered in [16] (i.e., network, matching, edge cover,

<sup>11</sup> Technically, this polytope can also contain paths with a finite number of disjoint cycles, but these can always be removed in the end.

<sup>12</sup> Note that common source network congestion games are not symmetric and are thus not captured by the class of totally unimodular congestion games considered below.



vertex cover and stable set congestion games on bipartite graphs, and their respective extensions to the maximum (or minimum) cardinality versions) the parameter  $r$  is actually polynomially bounded in  $n$  and  $m$ , so then this assumption is justified.

### 5.3 Non-symmetric matroid congestion games

In a *non-symmetric matroid congestion game*  $\Gamma = (N, E, (\mathcal{S}_i), (c_e))$ , the strategy set of player  $i$  is given by the bases  $\mathcal{B}_i$  of a matroid  $\mathcal{M}_i = (E, \mathcal{I}_i)$  for  $i \in N$ .<sup>13</sup> The incidence vectors of the bases of  $\mathcal{B}_i$  can be described by the base matroid polytope

$$P_i = \{x : x(A) \leq r_i(A), A \subset E, x(E) = r_i(E), x \geq 0\}$$

as introduced in the preliminaries. That is, for every player we have a polytope of the form  $P_i = \{x : Ax \leq b_i, x \geq 0\}$  where  $b_i$  is the rank function  $r_i$  of the matroid  $\mathcal{M}_i$ . In particular, it follows that the aggregation polytope is given by

$$P_N = \{y : y(A) \leq \sum_i r_i(A), A \subset E, y(E) = \sum_i r_i(E), y \geq 0\}.$$

The polytope  $P_N$  has a box-TDI description, which follows from [47, Theorem 46.2].<sup>14</sup> The integer decomposition property is also satisfied (see, e.g., [47, Corollary 46.2c]). Using similar arguments as for  $r$ -arborescences (see below), we derive a strongly polynomial time algorithm to compute a minimum of Rosenthal’s potential. We also prove a result that is of independent interest: For non-symmetric matroid congestion games, we can derive a local search algorithm to compute a global minimum of Rosenthal’s potential in strongly polynomial time. This local search algorithm can be seen as a natural generalization of best response dynamics (which are known to arrive at a local optimum in polynomial time [1]). Our algorithm is based on a combinatorial approach to compute the symmetric difference decomposition for these games (which is of a specific form). The details are given in Appendix A.

### 5.4 Symmetric matroid intersection congestion games

In *symmetric matroid intersection congestion games*  $\Gamma = (N, E, \mathcal{S}, (c_e))$  the (symmetric) strategy set  $\mathcal{S}$  of all players is given by the common bases of two matroids  $\mathcal{M}_1 = (E, \mathcal{I}_1)$  and  $\mathcal{M}_2 = (E, \mathcal{I}_2)$  over a common element set  $E$ . The polytope  $P$  of the players corresponds to the common base polytope  $P_{\mathcal{M}_1, \mathcal{M}_2}$  as defined in (2.4), i.e.,

$$P = \{x : x(A) \leq r_i(A) \forall A \subset E, x(E) = r_i(E) \text{ for } i = 1, 2, x \geq 0\}.$$

<sup>13</sup> Our framework also captures the *independent set congestion games* studied in [16]. However, we mainly focus on non-negative cost functions here (because of the inefficiency measures) and then these games are trivial.

<sup>14</sup> To see this, we use the fact that the rank function is submodular and that the sum of submodular functions is again submodular. We can then apply Theorem 46.2 in [47].

The describing system of  $P$  is box-TDI (see, e.g., [47, Corollary 41.12e]). Further, as noted in the preliminaries there is a separation oracle for  $P$  (and thus  $P_N$ ) which runs in time polynomial in  $|E|$  and the complexity of the independence oracles for  $\mathcal{M}_1$  and  $\mathcal{M}_2$ . However, it is not precisely known for which cases of matroid intersection the integer decomposition property holds.

**Example 5.1** (*r*-Arborescences) Let  $D = (V, A)$  be a directed graph. An *r*-arborescence in  $D$  is a directed spanning tree rooted in  $r \in V$ . The set of all *r*-arborescences can be seen as the set of common bases of two matroids. The first matroid  $\mathcal{M}_1$  is the graphic matroid on the undirected graph  $D' = (V, A')$ , where  $A'$  is the set formed by replacing every directed arc in  $A$  with its undirected version, i.e.,  $A' = \{\{u, v\} : (u, v) \in A\}$ . The second matroid  $\mathcal{M}_2$  is the partition matroid in which independent sets are given by sets of arcs for which there is at most one incoming arc at every node  $v \neq r$  (we assume there are no incoming arcs at  $r$ ). Thus, the common base polytope  $P_{\mathcal{M}_1, \mathcal{M}_2}$  describes the arborescences of  $D$  and we let  $P = P_{\mathcal{M}_1, \mathcal{M}_2}$ .

We argue that there is a strongly polynomial time algorithm for computing a minimum of Rosenthal's potential. First note that the describing system of  $P_{\mathcal{M}_1, \mathcal{M}_2}$  is box-TDI (see [47, Corollary 41.12e]). Also,  $P_{\mathcal{M}_1, \mathcal{M}_2}$  satisfies the integer decomposition property, which follows from Edmonds' Disjoint Arborescences Theorem [18]. By Theorem 3.3, we can compute a minimum of Rosenthal's potential in time polynomial in  $n, m, \sum_i \langle b_i \rangle$  and the complexity of a separation oracle for  $P_{\mathcal{M}_1, \mathcal{M}_2}$ . The elements of the vector  $b$  are bounded by  $|E|$ , by the definition of the rank functions. Moreover, it is not hard to see that there exist independence oracles for both  $\mathcal{M}_1$  and  $\mathcal{M}_2$  that run in time polynomial in  $m$ . These oracles can be used for separation oracles as described in the preliminaries. It is not hard to see that if both base matroid polytopes have a polynomial time separation oracle, then the intersection of these polytopes has one too. This shows that there is an algorithm for computing an optimal feasible load profile in time polynomial in  $n$  and  $m$ . Integer decomposition can also be done in time polynomial in  $n$  and  $m$  [28].

**Example 5.2** (**Intersection of strongly base-orderable matroids**) A matroid  $\mathcal{M} = (E, \mathcal{I})$  is strongly base-orderable if for every pair of bases  $B_1, B_2 \in \mathcal{B}$  there exists a bijection  $\tau : B_1 \rightarrow B_2$  such that for every  $X \subseteq B_1$ , we have  $B_1 - X + \tau(X) \in \mathcal{B}$ . As in the previous example, a box-TDI description follows from [47, Corollary 41.12e]. It is also known that the independent set polytope of the intersection of strongly base-orderable matroids has the integer decomposition property [37, Theorem 5.1].<sup>15</sup>

## 6 Bottleneck congestion games

A *bottleneck congestion game*  $\Gamma = (N, E, (\mathcal{S})_{i \in N}, (c_e)_{e \in E})$  is defined similarly to a congestion game, with the only difference that the objective of a player is to minimize the *maximum* (rather than the aggregated) congestion over all resources that she

<sup>15</sup> This also implies that the common base polytope has the integer decomposition property, since the integer decomposition property is preserved if we restrict ourselves to a face of a polytope with the integer decomposition property.

occupies. Formally, the cost of player  $i \in N$  under strategy profile  $s = (s_1, \dots, s_n)$  is given by  $C_i(s) = \max_{x_e \in s_i} c_e(x_e(s))$ .

Harks et al. [32] give a *dual greedy algorithm* to compute a strong equilibrium, which uses a *strategy packing oracle* as a subroutine (see Appendix B for details). They give efficient packing oracles for symmetric network congestion games, non-symmetric matroid congestion games and (a slight generalization of)  $r$ -arborescences. In particular, this leads to polynomial time algorithms for computing a strong equilibrium in these cases.

We adapt their algorithm to compute a load profile of a strong equilibrium for bottleneck polytopal congestion games satisfying the IDP and box-TDI property.

**Theorem 6.1** *Let  $\Gamma = (N, E, (S_i), (c_e))$  be a polytopal bottleneck congestion game whose aggregation polytope  $P_N$  satisfies IDP and box-TDI. Then a load profile of a strong equilibrium can be computed using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls.*

We first need to adapt the definition of the strategy packing oracle of [32] (see Appendix B) to load profiles. Below, we assume that  $P_N$  is an aggregation polytope of a polytopal congestion game.

**Load profile oracle**  $\mathfrak{D}(E = T \cup L, P_N, (u_e)_{e \in E}) :$

**Input:** A finite set of resources  $E = T \cup L$  with upper bounds  $(u_e)_{e \in E}$  and an aggregation polytope  $P_N$ .

**Output:** YES, if there exists a feasible load profile  $f \in P_N$  such that  $f_e = u_e$  for all  $e \in T$  and  $f_e \leq u_e$  for all  $e \in L$ ; NO otherwise.

Our adaptation of the dual greedy algorithm is given in Algorithm 1. Although the ideas are similar to the ones in [32], our algorithm only works with load profiles. In particular, we do not have to explicitly compute decompositions of feasible load profiles in intermediate steps of the algorithm, which (significantly) improves the running time. Intuitively, our algorithm works as follows. We start with capacities of  $n$  on every resource. In every step we pick a resource  $e' \in L$  with maximum cost among all resources that are called *loose*, and check whether there is a feasible load profile if we reduce the capacity on  $e'$  by one. If this is not possible, we remove  $e'$  from  $L$  and add  $e'$  to the set  $T$  of so-called *tight* resources. Note that after the algorithm has terminated, all resources are in the set  $T$ .

The following lemma shows that the load profile output by our algorithm corresponds to a strong equilibrium. Its proof is similar to the correctness proof of the dual greedy algorithm by Harks et al. [32].

**Lemma 6.1** *Algorithm 1 computes a load profile of a strong equilibrium in time polynomial in  $n, m$  and the complexity of the load profile oracle.*

**Proof** Clearly, Algorithm 1 can be executed in time polynomial in  $n, m$  and the complexity of the load profile oracle. Let  $f$  be the load profile output by the algorithm. Assume without loss of generality that the resources in  $E = [m]$  are added to the set  $T$  in the order  $(1, 2, \dots, m)$ . Let  $s = (s_1, \dots, s_n)$  be a strategy profile corresponding

---

**ALGORITHM 1:** Load profile-dual greedy algorithm.

---

**Input** : Bottleneck congestion game  $\Gamma = (N, E, (S_i), (c_e))$ , load profile oracle  $\mathcal{D}$   
**Output** : Load profile of strong equilibrium of  $\Gamma$

```

1 set  $N' = N, u_e = n$  for all  $e \in E, T = \emptyset, L = E$ 
2 while  $\{e \in L : u_e > 0\} \neq \emptyset$  do
3   choose  $e' \in \operatorname{argmax}\{c_e(u_e) : e \in L, u_e > 0\}$ 
4    $u_{e'} := u_{e'} - 1$ 
5   if  $\mathcal{D}(T \cup L, P_N, (u_e)_{e \in E}) = \text{NO}$  then
6      $u_{e'} := u_{e'} + 1$ 
7      $L = L \setminus \{e'\}, T = T \cup \{e'\}$ 
8   end
9 end
10 return  $(u_e)_{e \in E}$ 

```

---

to the load profile  $f$ . We define  $\sigma_e = \{i \in N : e \in s_i\}$  as the set of players using resource  $e$ . Moreover, we set  $N_1 = \sigma_1$ , and define

$$N_j = \sigma_j \setminus (N_1 \cup \dots \cup N_{j-1}) \quad \text{for } j = 1, \dots, m.$$

We show that players in  $N_1$  will never participate in a coalitional deviation in which every player strictly improves.

Let  $D \subset N$  be a coalition of players that can profitably deviate to strategy profile  $t = (t_D, s_{-D})$ . Remember that  $x_e(t)$  denotes the number of players using resource  $e$  in strategy profile  $t$ . Let  $(u_e)_{e \in E}$  be the capacity vector for which the load profile oracle returns NO for the first time in line 5. We consider two cases:

**Case 1**  $x_e(t) \leq u_e$  for all  $e \in E$ . In this case, there must be  $|N_1|$  players using resource 1; otherwise, the oracle would have returned YES because the load profile of  $t$  would have been feasible for the capacities  $(u_e)$ . Further, for all players  $i \in N_1$ , we have  $C_i(s) = c_e(x_e(s))$ , which is in particular the highest player cost in the strategy profile  $s$ . In particular, this implies that if a player in  $N_1$  would strictly improve, then she cannot use resource 1 in  $t$ . This means that another player is now using resource 1 in  $t$ , but that player can never have strictly better cost than it had in  $s$ .

**Case 2**  $x_e(t) > u_e$  for some  $e \in E$ . Using similar arguments, we can show that  $x_1(t) < u_1$  if some player in  $N_1$  is also part of  $D$  (since  $c_1(x_1(s))$  is the maximum cost resource in  $s$ ). Since the algorithm iteratively reduces the capacities of resources with maximum cost, we must have that  $c_e(x_e(t)) \geq c_1(u_1)$ . Further, since  $x_e(t) > u_e$ , at least one player in  $D$  must be using resource  $e$  in  $t$ . But this player cannot have strictly improved then.

We can now use induction to show that no player in  $N_j$  will ever participate in a coalitional deviation. Assume that the players in  $N_1 \cup \dots \cup N_{j-1}$  will never participate in a coalitional deviation. By using similar arguments as above, we can show that the players in  $N_j$  will also never participate in a coalitional deviation.  $\square$

**Proof** (Theorem 6.1) By Lemma 6.1, Algorithm 1 computes a load profile of a strong equilibrium in time polynomial in  $n, m$  and the complexity of a load profile oracle.

Based on a separation oracle of  $P_N$ , we now show that there is an efficient load profile oracle. Given that  $P_N$  has a box-TDI description, it follows that the polytope

$$\{y : Ay \leq \sum_{i \in N} b_i\} \cap \{y_e = u_e : e \in T\} \cap \{0 \leq y_e \leq u_e : e \in L\}$$

is integral. We can then use a separation oracle for  $P_N$  to find an integral vector in this polytope using at most  $\text{poly}(n, m, \phi)$  arithmetic operations and separation oracle calls. This concludes the proof.  $\square$

Once we have obtained the feasible load profile, we can use an integer decomposition algorithm to find the corresponding strategies of the players. If the integer decomposition can be done within the same time bounds as stated in Theorem 6.1, we obtain a (strongly) polynomial algorithm for computing a strong equilibrium in a polytopal bottleneck congestion game. In particular, this applies to all applications mentioned in Sect. 5.

We end this section by showing that for *matroid bottleneck congestion games*, we can use our techniques from the previous sections to derive an upper bound on the *strong price of stability (SPoS)*.

The proof of the following theorem exploits that Algorithm 1 in fact computes a global optimum of Rosenthal’s potential in the case of matroid bottleneck congestion games.

**Theorem 6.2** *Let  $\Gamma = (N, E, (S_i), (c_e))$  be a non-symmetric matroid bottleneck congestion game with cost functions in class  $\mathcal{D}$ . Then  $\text{SPoS}(\Gamma) \leq \rho(\mathcal{D})$ .*

**Proof** Let  $f$  be the load profile returned by the algorithm.  $f$  is a strong equilibrium by Lemma 6.1. We prove that  $f$  is also a global optimum of Rosenthal’s potential. We then obtain a bound of  $\rho(\mathcal{D})$  on the strong price of stability by using similar arguments as in the proof of Theorem 4.1.

Suppose for contradiction that  $f$  is not a global optimum. Then there exist resources  $a$  and  $b$  such that  $c_a(f_a) > c_b(f_b + 1)$  and for which the load profile  $f'$ , defined by  $f'_a = f_a - 1$ ,  $f'_b = f_b + 1$  and  $f'_e = f_e$  for all  $e \in E \setminus \{a, b\}$ , is feasible. This claim follows from similar arguments as given in the proof of Theorem A.2 (Appendix A).

Now, consider the point in execution of the algorithm where the capacity of resource  $a$  was fixed at  $f_a$ . Since  $c_a(f_a) > c_b(f_b + 1)$ , we must have had  $f_b + 1 \leq u_b$  at that point (since the algorithm iteratively reduces the capacity of resources with maximum cost). But this contradicts the fact that the load profile oracle returned NO at this point; to see this note that  $f'$  would have been a feasible load profile for the capacity vector in which  $u_a$  was reduced by 1 (as in line 4 of the algorithm).  $\square$

## 7 Concluding remarks

We identified two structural properties of polytopal congestion games which are sufficient to efficiently compute a global minimizer of Rosenthal’s potential: IDP and box-TDI. Further, we proved that the computed Nash equilibria obtain a social cost approximation guarantee of  $\rho(\mathcal{D})$  if the cost functions belong to class  $\mathcal{D}$ . As we

showed, this also establishes a tight bound on the price of stability for polytopal congestion games satisfying IDP and box-TDI. Intuitively, these games thus inherit the social cost approximation guarantee of non-atomic network routing games [13]. In our inefficiency proofs, we crucially exploited the symmetric difference decomposition property of polytopes; we believe that this new notion might be useful also in other contexts. Finally, we provided several examples of classes of congestion games that can be cast into our framework and showed that some of the results also extend to bottleneck congestion games.

For future work it would be interesting to see whether our techniques extend to other classes of games, e.g., to special cases of weighted congestion games. Note that, although having an *exact* potential function turned out to be convenient in this paper, our approach per se is not limited by this requirement. In fact, it would be interesting to see how different (ordinal) potential functions impact the inefficiency guarantee of the respective global potential function minimizers.

We feel that in general the power of polyhedral techniques to compute good Nash equilibria in games is not well-understood and worth being investigated more intensively. In particular, research in this direction opens up an intriguing connection between the fields of polyhedral combinatorics and computational game theory.

**Acknowledgements** Open access funding provided by Projekt DEAL. We thank Carla Groenland, Tobias Harks, Bart de Keijzer and Daniel Dadush for helpful discussions and suggestions. Further, we thank the anonymous reviewers for their valuable comments on the paper.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## Appendix A: Combinatorial symmetric difference decomposition of non-symmetric matroid congestion games

In this section, we describe a combinatorial approach for computing the symmetric difference decomposition of non-symmetric matroid congestion games. Our analysis also provides a local search algorithm which can be seen as a natural generalization of best response dynamics.

Throughout this section, we let  $\Gamma = (N, E, (\mathcal{S}_i), (c_e))$  be a non-symmetric matroid congestion game, where the strategy set  $\mathcal{S}_i$  of each player  $i \in N$  is given by the bases  $\mathcal{B}_i$  of a matroid  $\mathcal{M}_i = (E, \mathcal{I}_i)$ .

### A.1 Symmetric difference decomposition

We start by deriving the symmetric difference decomposition.

Let  $s = (s_1, \dots, s_n)$  and  $t = (t_1, \dots, t_n)$  be two feasible strategy profiles with (feasible) load profiles  $f$  and  $g$ , respectively. We need the following result for matroids.

**Proposition A.1** [47] *Let  $\mathcal{M} = (E, \mathcal{I})$  be a matroid and let  $\mathcal{B}$  denote the set of bases of  $\mathcal{M}$ . Then for all  $B, B' \in \mathcal{B}$ , there exists a bijection  $\tau : B \setminus B' \rightarrow B' \setminus B$  such that  $B - x + \tau(x) \in \mathcal{B}$  for all  $x \in B \setminus B'$ .*

For  $i \in N$ , let  $\tau^i : s_i \setminus t_i \rightarrow t_i \setminus s_i$  be a bijection satisfying Proposition A.1. Let  $G = (V, A)$  be a directed multigraph defined by  $V = E$ , and the multiset

$$A = \bigcup_{i \in N} \{(e, \tau^i(e)) : e \in s_i \setminus t_i\}.$$

Note that, implicitly, every arc corresponds to a unique player. Since, for a fixed player  $i$  the bases  $s_i$  and  $t_i$  have the same size, it follows that

$$K := \sum_{e: f_e > g_e} (f_e - g_e) = \sum_{e: g_e > f_e} (g_e - f_e).$$

In particular, the graph  $G$  can be decomposed into  $K$  edge-disjoint paths (or chains)  $(P_j)_{j=1, \dots, K}$  with the property that they start at an *overloaded* resource  $e$  with  $f_e > g_e$  and end at an *underloaded* resource  $e$  with  $g_e > f_e$  (and possibly some cycles). It follows that we can write

$$g - f = \sum_{j=1}^K x^j, \quad \text{where for each path } P_j = (a_j, \dots, b_j) \text{ we have}$$

$$x_e^j = \begin{cases} 1 & \text{if } e = b_j, \\ -1 & \text{if } e = a_j, \\ 0 & \text{otherwise.} \end{cases}$$

We claim that the load profiles  $f + x^j$  are again feasible.

Let us first introduce some more terminology. Let  $\mathcal{M} = (E, \mathcal{I})$  be a matroid and let  $I \in \mathcal{I}$ . Let  $\mathcal{D}_{\mathcal{M}}(I) = (E, A_{\mathcal{M}}(I))$  be the directed *exchange* graph defined by

$$A_{\mathcal{M}}(I) = \{(y, z) : y \in I, z \in E \setminus I, I - y + x \in \mathcal{I}\}.$$

The following proposition will be used below.

**Proposition A.2** [47] *Let  $\mathcal{M} = (E, \mathcal{I})$  be a matroid and let  $I \in \mathcal{I}$ . Let  $D_{\mathcal{M}}(I)$  be as defined above, and let  $J \subseteq E$  be such that  $|I| = |J|$  and such that  $A_{\mathcal{M}}(I)$  contains a unique perfect matching on  $I \Delta J$ . Then  $J \in \mathcal{I}$ .*

Moreover, for matroids  $\mathcal{M}_i = (E, \mathcal{I}_i)$ ,  $i \in N$ , let  $\mathcal{D}(I_1, \dots, I_n) = (E, A)$  be the multigraph defined by the

$$A = \bigcup_{i \in N} A_{\mathcal{M}_i}(I_i).$$

We implicitly label every  $a \in A$  with a player  $i$ , namely the player for which  $a \in \mathcal{M}_i(I_i)$ . For a path  $Q = (e_1, \dots, e_p)$  in  $\mathcal{D}$ , we denote by  $A_Q^i$  the set of arcs corresponding to player  $i$ , i.e.,

$$A_Q^i = \{a \in Q : a = (e_j, e_{j+1}) \text{ has label } i\}.$$

Further, we let  $T_Q^i$  and  $H_Q^i$  contain the tails and heads of the arcs in  $A_Q^i$ , respectively, i.e.,

$$T_Q^i = \{v \in E : a = (v, w) \in A_Q^i\} \quad \text{and} \quad H_Q^i = \{w \in E : a = (v, w) \in A_Q^i\}.$$

We let  $\sigma_Q^i : T_Q^i \rightarrow H_Q^i$  denote the bijection that maps every tail to its head. We say that a shift over the path  $Q$  is feasible, if for  $i = 1, \dots, n$ , it holds that  $I_i - T_Q^i + H_Q^i \in \mathcal{I}_i$ .<sup>16</sup>

Let us now come back to the paths  $P_j$ . By definition of  $\mathcal{D}(s_1, \dots, s_n)$ , every path  $P_j$  is contained in the graph  $\mathcal{D}(s_1, \dots, s_n)$ . In particular this means that there is at least one path from  $a_j$  to  $b_j$ . Then there is also a *shortest path* (in terms of number of arcs) from  $a_j$  to  $b_j$ . By Lemma A.1 (given below), the load profile  $f + x^j$  is feasible since we can shift players over some shortest  $(a_j, b_j)$  path  $Q_j$  such that the resulting bases are again feasible for all players. That is, we apply Lemma A.1 with the bases  $I_i = s_i$ .

It remains to prove Lemma A.1.

**Lemma A.1** *Let  $\mathcal{D}(I_1, \dots, I_n) = (V, A)$  be as defined above, and let  $a, b \in V$ . If  $Q = (a, \dots, b)$  is a shortest  $(a, b)$ -path, then  $I_i - T_Q^i + H_Q^i \in \mathcal{I}_i$  for all  $i = 1, \dots, n$ .<sup>17</sup>*

**Proof** Fix some  $i$ . We let  $I = I_i$  and  $J = I_i - T^i + H^i (= s_i - T^i + \sigma^i(H^i))$ . In particular, the function  $\sigma^i$  as defined above gives a perfect matching on  $I \Delta J$ . We claim that  $\sigma^i$  is the unique perfect matching between  $T^i$  and  $H^i$ . It follows from Proposition A.2 that  $J \in \mathcal{I}$ . Let  $\rho$  be an arbitrary perfect matching. Let  $(v, w)$  be the first arc on  $Q$  corresponding to player  $i$ . If  $\rho(v) \neq w$ , then this means that  $Q$  was not a shortest  $(a, b)$ -path, so we must have  $\rho(v) = w$ . A similar argument can be given for the second arc corresponding to  $i$ , then the third arc, etc. We find that  $\rho = \sigma^i$ , and this concludes the proof.  $\square$

## A.2 Local search algorithm

We can derive a local search algorithm based on the analysis above. We first introduce some terminology. We say that the difference between two strategy profiles  $s$  (with load profile  $f$ ) and  $s'$  (with load profile  $f'$ ) is *minimal*, if there exist resources  $a, b$

<sup>16</sup> Subsequently, we omit the subscript  $Q$  if it is clear from the context which path is meant.

<sup>17</sup> A similar statement is shown in [2, Lemma 4.5]. However, our proof is different and seems (much) shorter because of the fact that we use the result in [47].



such that

$$f_e - f'_e = \begin{cases} 1 & \text{if } e = a \\ -1 & \text{if } e = b \\ 0 & \text{otherwise.} \end{cases}$$

We define the neighborhood of a strategy profile  $s$  by

$$\mathcal{N}(s) = \{s\} \cup \{s' \in \times_i \mathcal{S}_i : \text{the difference between } s \text{ and } s' \text{ is minimal}\}. \quad (7.1)$$

Note that by definition the load profiles  $f$  and  $f'$  of two neighboring strategy profiles  $s, s'$  with  $s' \in \mathcal{N}(s)$ , respectively, must differ by one on exactly two resources. However, this load difference might not be achievable by a single-player deviation. In fact, it is not hard to construct examples, where a sequence of unilateral deviations is needed to reach  $s'$  from  $s$ .

We prove the following lemma.

**Lemma A.2** *A strategy profile  $s$  minimizes Rosenthal's potential if and only if  $s$  is a local minimum of Rosenthal's potential with respect to the neighbourhood  $\mathcal{N}(s)$ .*

**Proof** First, let  $s$  be a strategy profile minimizing Rosenthal's potential. It follows directly that  $s$  is a local minimum with respect to  $\mathcal{N}(s)$ , since  $s$  is a global optimum of the potential function.

Conversely, let  $s$  be a local minimum with respect to  $\mathcal{N}(s)$  and suppose that  $s$  is not a minimizer of Rosenthal's potential. We claim that there exists a strategy profile  $s'$  such that

$$\Delta(f, g) := \sum_{e: f_e > g_e} (f_e - g_e)c_e(f_e) - \sum_{e: f_e < g_e} (g_e - f_e)c_e(f_e + 1) > 0, \quad (7.2)$$

where  $f$  and  $g$  are the load profiles of  $s$  and  $s'$ , respectively. Assume for contradiction that  $\Delta(f, g) \leq 0$  for all feasible load profiles  $g$ . Then

$$\begin{aligned} \Phi(f) - \Phi(g) &= \sum_{e: f_e > g_e} \sum_{k=g_e+1}^{f_e} c_e(k) - \sum_{e: g_e > f_e} \sum_{k=f_e+1}^{g_e} c_e(k) \\ &\leq \sum_{e: f_e > g_e} (f_e - g_e)c_e(f_e) - \sum_{e: f_e < g_e} (g_e - f_e)c_e(f_e + 1) \leq 0, \end{aligned}$$

where the first inequality holds because the cost functions are non-decreasing and non-negative. But then  $f$  minimizes the potential function  $\Phi$ , which is a contradiction.

Let the paths  $W_1, \dots, W_K$  form the path decomposition of the multi-graph  $G$  (as described above) for the strategies  $s$  and  $s'$ . Because of (7.2) there must be some path  $W_j = (a_j, \dots, b_j)$  such that  $c_a(f_a) - c_b(f_b + 1) > 0$ . This contradicts the fact that  $s$  is a local minimum with respect to  $\mathcal{N}(s)$ .  $\square$

We next show that, given an arbitrary strategy profile  $s$ , we can determine in polynomial time whether there is an improving move with respect to the altered neighborhood defined in (7.1).

**Lemma A.3** *Assume that for every player  $i \in N$  we have a polynomial independence oracle for matroid  $\mathcal{M}_i = (E, \mathcal{I}_i)$ . Then for every strategy profile  $s$ , we can check in time polynomial in  $m$  and the independence oracles of the matroids  $\mathcal{M}_i$ , whether or not there exists a strategy profile  $s' \in \mathcal{N}(s)$  with  $\Phi(s') < \Phi(s)$ .*

**Proof** Note that there are at most  $m(m - 1)$  possibilities for the resources  $a$  and  $b$  in the description of strategy profiles with minimal difference. For  $a$  and  $b$  fixed, with  $c_a(f_a) > c_b(f_b + 1)$ , we can in polynomial time check whether or not there exists a chain starting at  $a$  and ending in  $b$ . For example, we can run an all-pairs shortest path algorithm on the graph  $\mathcal{D}(s_1, \dots, s_n)$ , which can be constructed in strongly polynomial time using the polynomial independence oracles. By construction, we know that every shortest path yields a new strategy profile  $s'$  with  $f'_a = f_a - 1$  and  $f'_b = f_b + 1$  (and all the other loads remain the same). Note that  $s'$  can be constructed in polynomial time. □

Exploiting the insights above, we conclude that we can find a global optimum of  $\Phi$  in strongly polynomial time as follows: Starting from an arbitrary strategy profile  $s_0$ , iteratively perform local improvement steps with respect to the neighborhood  $\mathcal{N}(\cdot)$  as defined in (7.1) until a local optimum is reached. By Lemma A.2, the final strategy profile is a global optimum of Rosenthal’s potential. Further, by Lemma A.3, each improving move can be done in polynomial time. The fact that this local search takes only a strongly polynomial number of steps follows from arguments similar to the ones in [1] (showing that any better-response sequence in matroid congestion games has polynomial length).

### A.3 Example

We give an example illustrating the analysis above.

Let us consider the (non-symmetric) game in Fig. 2. The matroid of player  $a$  is the graphic matroid on the complete graph  $K_4$  (with spanning trees as bases). The matroid of player  $b$  is a 1-uniform matroid on the set  $E_2 = \{\{1, 4\}, \{1, 2\}\}$ . The strategies of the players in  $s$  are given by the bold edges, and the strategies in  $t$  by the dotted edges.

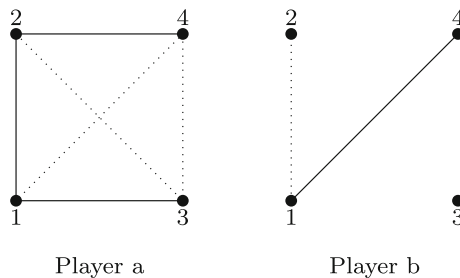


Fig. 2 Matroids and strategies of players  $a$  and  $b$

Let  $\tau^a$  be given by  $\tau^a(\{2, 4\}) = \{1, 4\}$ ,  $\tau^a(\{1, 3\}) = \{2, 3\}$  and  $\tau^a(\{1, 2\}) = \{3, 4\}$  (note that  $\tau^a$  satisfies the condition in Proposition A.1). For  $\tau^b$  there is only a unique choice defined by  $\tau^b(\{1, 4\}) = \{1, 2\}$ . The (unique) path decomposition for  $s$  and  $t$  is given in Fig. 3.

The path  $P_1$  from  $\{2, 4\}$  to  $\{3, 4\}$  can be replaced by the shorter chain  $P'_1$  as in Fig. 4.

## Appendix B: Omitted material from Sect. 6

### The algorithm of Harks et al. [32]

We briefly summarize the *dual greedy algorithm* of Harks et al. [32] for computing a strong equilibrium in bottleneck congestion games (see [32] for more details). Their algorithm is based on a *strategy packing oracle*.

**Strategy packing oracle**  $\mathfrak{D}(E, (\mathcal{S}_i)_{i \in N}, (u_e)_{e \in E})$  [32]:

**Input:** A finite set of resources  $E$  with upper bounds  $(u_e)_{e \in E}$  and strategy sets  $(\mathcal{S}_i)_{i \in N}$  (given implicitly by some combinatorial property).

**Output:** Strategy profile  $s \in \times_{i \in N} \mathcal{S}_i$  such that  $x_e(s) \leq u_e$  for all  $e \in E$ , or  $\emptyset$  if no such strategy profile exists.

Basically, the *dual greedy algorithm* (Algorithm 2) works as follows: In every step of the algorithm, we have capacity constraints  $(u_e)_{e \in E}$  for which it is known that there exists a strategy profile respecting these capacities. The algorithm then selects a resource  $e'$  with maximum cost (line 3) and checks if there exists a feasible strategy profile for the capacities in which  $u_{e'}$  is decreased by 1. If so, then  $u_{e'}$  is updated and we continue with the new vector of capacity constraints. Otherwise, the strategies of the  $u_{e'}$  players using resource  $e'$  are fixed (and all variables are updated accordingly).

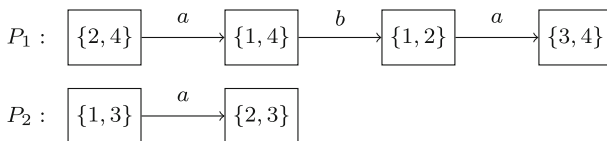


Fig. 3 Path decomposition for the profiles  $s$  and  $t$ , based on the bijections  $\tau^1$  and  $\tau^2$

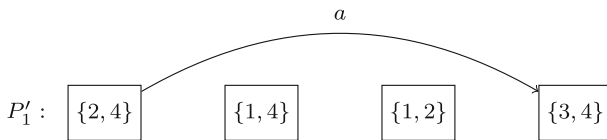


Fig. 4 The chain  $P'_1$  that is a feasible path from  $\{2, 4\}$  to  $\{3, 4\}$  as in Fig. 3

We say that resource  $e'$  becomes *frozen*. Note that the players of which the strategies are fixed do not change anymore; these players are frozen too.

---

**ALGORITHM 2:** Dual greedy algorithm of Harks et al. [32].

---

**Input** : Bottleneck congestion game  $\Gamma = (N, E, (\mathcal{S}_i), (c_e))$ , strategy packing oracle  $\mathcal{D}$   
**Output** : Strong equilibrium of  $\Gamma$

```

1 set  $N' = N, u_e = n, x_e = 0 \forall e \in E$ , and  $s' = \mathcal{D}(E, (\mathcal{S}_i)_{i \in N'}, (u_e))$ 
2 while  $\{e \in E : u_e > 0\} \neq \emptyset$  do
3   choose  $e' \in \operatorname{argmax}\{c_e(u_e + x_e) : e \in E, u_e > 0\}$ 
4    $u_{e'} = u_{e'} - 1$ 
5   if  $\mathcal{D}(E, (\mathcal{S}_i)_{i \in N'}, (u_e)) = \emptyset$  then
6      $u_{e'} = u_{e'} + 1$ 
7     foreach  $j \in N'$  with  $e' \in s'_j$  do
8        $s_j = s'_j$ 
9       set  $x_e = x_e + 1, u_e = u_e - 1$  for all  $e \in s'_j$ 
10       $N' = N' \setminus \{j\}$ 
11     end
12   end
13    $s' = \mathcal{D}(E, (\mathcal{S}_i)_{i \in N'}, (u_e))$ 
14 end
15 return  $s = (s_1, \dots, s_n)$ 

```

---

## References

- Ackermann, H., Röglin, H., Vöcking, B.: On the impact of combinatorial structure on congestion games. *J. ACM* **55**(6), 25:1–25:22 (2008)
- Aigner-Horev, E., Carmesin, J., Fröhlich, J.O.: Infinite matroid union. (2012). [arXiv:1111.0602](https://arxiv.org/abs/1111.0602)
- Aland, S., Dumrauf, D., Gairing, M., Monien, B., Schoppmann, F.: Exact price of anarchy for polynomial congestion games. *SIAM J. Comput.* **40**(5), 1211–1233 (2011)
- Anshelevich, E., Dasgupta, A., Kleinberg, J., Tardos, E., Wexler, T., Roughgarden, T.: The price of stability for network design with fair cost allocation. *SIAM J. Comput.* **38**(4), 1602–1623 (2008)
- Awerbuch, B., Azar, Y., Epstein, A.: The price of routing unsplittable flow. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, pp. 57–66 (2005)
- Baum, S.P., Trotter, L.E.: Integer rounding and polyhedral decomposition for totally unimodular systems. In: Henn, R., Korte, B., Oettli, W. (eds.) Optimization and Operations Research: Proceedings of a Workshop Held at the University of Bonn, October 2–8, 1977, pp. 15–23 (1978)
- Caragiannis, I., Flammini, M., Kaklamanis, C., Kanellopoulos, P., Moscardelli, L.: Tight bounds for selfish and greedy load balancing. *Algorithmica* **61**(3), 606–637 (2011)
- Chan, H., Jiang, A.X.: Congestion games with polytopal strategy spaces. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence, pp. 165–171 (2016)
- Chien, S., Sinclair, A.: Convergence to approximate Nash equilibria in congestion games. In: Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 169–178 (2007)
- Christodoulou, G., Gairing, M.: Price of stability in polynomial congestion games. *ACM Trans. Econ. Comput.* **4**(2), 10:1–10:17 (2016)
- Christodoulou, G., Koutsoupias, E.: The price of anarchy of finite congestion games. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, pp. 67–73 (2005)
- Christodoulou, G., Mirrokni, V.S., Sidiropoulos, A.: Convergence and approximation in potential games. *Theor. Comput. Sci.* **438**, 13–27 (2012)
- Correa, J.R., Schulz, A.S., Stier-Moses, N.E.: Selfish routing in capacitated networks. *Math. Oper. Res.* **29**(4), 961–976 (2004)

14. Correa, J.R., Schulz, A.S., Stier-Moses, N.E.: A geometric approach to the price of anarchy in nonatomic congestion games. *Games Econ. Behav.* **64**(2), 457–469 (2008)
15. Dantzig, G.B.: *Linear Programming and Extensions*. Princeton University Press, Princeton (1963)
16. Del Pia, A., Ferris, M., Michini, C.: Totally unimodular congestion games. In: *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 577–588 (2017)
17. de Jong, J., Kern, W., Steenhuisen, B., Uetz, M.: The asymptotic price of anarchy for k-uniform congestion games. In: *Approximation and Online Algorithms*, pp. 317–328 (2018)
18. Edmonds, J.: *Combinatorial Optimization—Eureka, You Shrink! Papers Dedicated to Jack Edmonds*, chap. Submodular Functions, Matroids, and Certain Polyhedra (2003)
19. Edmonds, J., Giles, R.: A min–max relation for submodular functions on graphs. In: *Studies in Integer Programming, Annals of Discrete Mathematics*, vol. 1, pp. 185–204. Elsevier (1977)
20. Even-Dar, E., Kesselman, A., Mansour, Y.: Convergence time to Nash equilibrium in load balancing. *ACM Trans. Algorithms* **3**(3), 1–32 (2007)
21. Fabrikant, A., Papadimitriou, C., Talwar, K.: The complexity of pure Nash equilibria. In: *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pp. 604–612 (2004)
22. Feldman, M., Immorlica, N., Lucier, B., Roughgarden, T., Syrgkanis, V.: The price of anarchy in large games. In: *Proceedings of the 48th Annual ACM Symposium on Theory of Computing*, pp. 963–976 (2016)
23. Ford, D.R., Fulkerson, D.R.: *Flows in Networks*. Princeton University Press, Princeton (2010)
24. Fotakis, D.: Congestion games with linearly independent paths: convergence time and price of anarchy. *Theory Comput. Syst.* **47**(1), 113–136 (2010)
25. Frank, A., Tardos, É.: An application of simultaneous diophantine approximation in combinatorial optimization. *Combinatorica* **7**(1), 49–65 (1987)
26. Fujishige, S.: Lexicographically optimal base of a polymatroid with respect to a weight vector. *Math. Oper. Res.* **5**(2), 186–196 (1980)
27. Fujishige, S., Goemans, M., Harks, T., Peis, B., Zenklusen, R.: Congestion games viewed from m-convexity. *Oper. Res. Lett.* **43**(3), 329–333 (2015)
28. Gabow, H.: A matroid approach to finding edge connectivity and packing arborescences. *J. Comput. Syst. Sci.* **50**(2), 259–273 (1995)
29. Gairing, M., Lücking, T., Mavronicolas, M., Monien, B., Rode, M.: Nash equilibria in discrete routing games with convex latency functions. *J. Comput. Syst. Sci.* **74**(7), 1199–1225 (2008)
30. Groenevelt, H.: Two algorithms for maximizing a separable concave function over a polymatroid feasible region. *Eur. J. Oper. Res.* **54**(2), 227–236 (1991)
31. Grötschel, M., Lovász, L., Schrijver, A.: *Geometric Algorithms and Combinatorial Optimization*. Springer, Berlin (1988)
32. Harks, T., Hoefer, M., Klimm, M., Skopalik, A.: Computing pure nash and strong equilibria in bottleneck congestion games. *Math. Program.* **141**(1), 193–215 (2013)
33. Hochbaum, D.S., Shanthikumar, J.G.: Convex separable optimization is not much harder than linear optimization. *J. ACM* **37**(4), 843–862 (1990)
34. Jeong, S., McGrew, R., Nudelman, E., Shoham, Y., Sun, Q.: Fast and compact: a simple class of congestion games. In: *Proceedings of the 20th National Conference on Artificial Intelligence*, pp. 489–494 (2005)
35. Koutsoupias, E., Papadimitriou, C.: Worst-case equilibria. In: *Proceedings of the 16th Annual Conference on Theoretical Aspects of Computer Science*, pp. 404–413 (1999)
36. Lücking, T., Mavronicolas, M., Monien, B., Rode, M.: A new model for selfish routing. *Theor. Comput. Sci.* **406**(3), 187–206 (2008)
37. McDiarmid, C.: Integral decomposition in polyhedra. *Math. Program.* **25**(2), 183–198 (1983)
38. Meyer, R.: A class of nonlinear integer programs solvable by a single linear program. *SIAM J. Control Optim.* **15**(6), 935–946 (1977)
39. Minoux, M.: Solving integer minimum cost flows with separable convex cost objective polynomially. *Math. Prog. Study* **26**, 237–239 (1986)
40. Monderer, D., Shapley, L.S.: Potential games. *Games Econ. Behav.* **14**(1), 124–143 (1996)
41. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V.: *Algorithmic Game Theory*. Cambridge University Press, New York (2007)
42. Onn, S., Rothblum, U.G., Tangir, Y.: Edge-directions of standard polyhedra with applications to network flows. *J. Glob. Optim.* **33**(1), 109–122 (2005)

43. Rosenthal, R.W.: A class of games possessing pure-strategy Nash equilibria. *Int. J. Game Theory* **2**, 65–67 (1973)
44. Roughgarden, T.: The price of anarchy is independent of the network topology. *J. Comput. Syst. Sci.* **67**(2), 341–364 (2003)
45. Roughgarden, T.: Intrinsic robustness of the price of anarchy. *J. ACM* **62**(5), 32 (2015)
46. Schrijver, A.: *Theory of Linear and Integer Programming*. Wiley, New York (1986)
47. Schrijver, A.: *Combinatorial optimization: polyhedra and efficiency*, vol. B, Matroids, trees, stable sets. Chapters 39–69. *Algorithms and combinatorics*. Springer (2003)
48. Veinott, A.F., Dantzig, G.B.: *Integral Extreme Points*. Defense Technical Information Center (1968)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Affiliations

Pieter Kleer<sup>1</sup> · Guido Schäfer<sup>2,3</sup>

✉ Pieter Kleer  
pkleer@mpi-inf.mpg.de  
Guido Schäfer  
g.schaefer@cw.nl

- <sup>1</sup> Max Planck Institute for Informatics, Saarland Informatics Campus (SIC), Saarbrücken, Germany
- <sup>2</sup> Networks and Optimization Group, Centrum Wiskunde & Informatica, Amsterdam, The Netherlands
- <sup>3</sup> Department of Econometrics and Operations Research, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands