Equilibrium computation in resource allocation games

Tobias Harks, Veerle Timmermans

Angaben zur Veröffentlichung / Publication details:

Harks, Tobias, and Veerle Timmermans. 2022. "Equilibrium computation in resource allocation games." *Mathematical Programming* 194 (1-1): 1–34. https://doi.org/10.1007/s10107-020-01604-z.

Mathematical Programming (2022) 194:1–34
https://doi.org/10.1007/s10107-020-01604-z

FULL LENGTH PAPER

Series A

Equilibrium computation in resource allocation games

Tobias Harks[1] · Veerle Timmermans[2]

Received: 6 April 2018 / Accepted: 4 December 2020 / Published online: 17 February 2021

Abstract

We study the equilibrium computation problem for two classical resource allocation games: atomic splittable congestion games and multimarket Cournot oligopolies. For atomic splittable congestion games with singleton strategies and player-specific affine cost functions, we devise the first polynomial time algorithm computing a pure Nash equilibrium. Our algorithm is combinatorial and computes the *exact* equilibrium assuming rational input. The idea is to compute an equilibrium for an associated *integrally-splittable* singleton congestion game in which the players can only split their demands in integral multiples of a common packet size. While integral games have been considered in the literature before, no polynomial time algorithm computing an equilibrium was known. Also for this class, we devise the first polynomial time algorithm and use it as a building block for our main algorithm. We then develop a polynomial time computable transformation mapping a multimarket Cournot competition game with firm-specific affine price functions and quadratic costs to an associated atomic splittable congestion game as described above. The transformation preserves equilibria in either game and, thus, leads – via our first algorithm – to a polynomial time algorithm computing Cournot equilibria. Finally, our analysis for integrally-splittable games implies new bounds on the difference between real and integral Cournot equilibria. The bounds can be seen as a generalization of the recent bounds for single market oligopolies obtained by Todd (Math Op Res 41(3):1125–1134 2016, https://doi.org/10.1287/moor.2015.0771).

Keywords Atomic splittable congestion games · Multimarket cournot competition · Equilibrium computation

An extended abstract of parts of this paper appeared in the Proceedings of the 19th International IPCO Conference on Integer Programming and Combinatorial Optimization under the title "Equilibrium Computation in Atomic Splittable Singleton Congestion Games".

✉ Veerle Timmermans
veerletimmermans@gmail.com

1 Department of Mathematics, Augsburg University, Augsburg, Germany

2 Department of Management Science, RWTH Aachen, Aachen, Germany

Mathematics Subject Classification 91A10 · 91A46 · 91B32

1 Introduction

One of the core topics in computational economics, operations research and optimization is the computation of equilibria. As pointed out by several researchers (e.g. [11,15]), the computational tractability of a solution concept contributes to its credibility as a plausible prediction of the outcome of competitive environments in practice. The most accepted solution concept in non-cooperative game theory is the Nash equilibrium – a strategy profile, from which no player wants to unilaterally deviate. While a Nash equilibrium generally exists only in mixed strategies, the practically important class of congestion games admits pure Nash equilibria, see Rosenthal [41]. In the classical model of Rosenthal, a pure strategy of a player consists of a subset of resources, and the congestion cost of a resource depends only on the number of players choosing the same resource.

While the complexity of computing equilibria for (discrete) congestion games has been intensively studied over the last decade (cf. [2,9,10,12,18,43]), the equilibrium computation problem for the *continuous* variant, that is, for *atomic splittable congestion games* is much less explored. In such a game, a player is associated with a positive demand and a collection of allowable subsets of the resources. A strategy for a player is a (possibly fractional) distribution of the player-specific demand over the allowable subsets. This quite basic model has been extensively studied, starting in the 80's in the context of traffic networks (Haurie and Marcotte [24]) and later for modeling communication networks (cf. Orda et al. [38] and Korilis et al. [27,28]), and logistics networks (Cominetti et al. [13]). Regarding polynomial time algorithms for equilibrium computation, we are only aware of four works: (1) For affine player-independent cost functions, there exists a convex potential whose global minima are pure Nash equilibria, see Cominetti et al. [13]. Thus, for any $\epsilon > 0$ one can compute an ϵ-approximate equilibrium in polynomial time by convex programming methods. (2) Huang [25] also considered affine player-independent cost functions, and he devised a combinatorial algorithm computing an exact equilibrium for routing games on symmetric s-t graphs that are so-called *well-designed*. This condition is met for instance by series-parallel graphs. His proof technique also uses the convex potential. (3) After the initial publication of the conference version of this article, Bhaskar and Lolakapuri [6] proposed two algorithms with exponential worst-case complexity that compute approximate Nash equilibria in games with convex costs, when set systems consist of singletons only. (4) Klimm and Warode [26] recently proved that computing a pure Nash equilibrium for atomic splittable and integer-splittable network congestion games with affine player-specific costs is PPAD-complete (see [39]). In light of these hardness results, it becomes clear that some restrictions on the strategy space are likely to be necessary to obtain polynomial time algorithms for equilibrium computation.

1.1 Our results and techniques

Atomic Splittable Congestion Games. We study atomic splittable congestion games as defined above, where the set systems consist of singletons only, and cost functions are player-specific, increasing and affine. We call these games *atomic splittable singleton congestion games* and for these games we develop the first polynomial time algorithm computing a pure Nash equilibrium. From now on we use equilibrium as shortcut for pure Nash equilibrium. Our algorithm is purely combinatorial and computes an *exact* equilibrium. The main ideas and constructions are as follows. By analyzing the first order necessary optimality conditions of an equilibrium, it can be shown that any equilibrium is *rational* as it is a solution to a system of linear equations with rational coefficients (assuming rational input). Using that equilibria are unique for singleton games (see Richmann and Shimkin [40] and Bhaskar et al. [5]), we further derive that the constraint matrix of the equation system is non-singular, allowing for an explicit representation of the equilibrium by Cramer's rule (using determinants of the constraint- and their sub-matrices). This way, we obtain an explicit lower bound on the minimum demand value for any used resource in the equilibrium. We further show that the unique equilibrium is also the unique equilibrium for an associated *integrally-splittable* game in which the players may only distribute the demands in *integer multiples* of a common *packet size* of some value $k^* \in \mathbb{Q}_{>0}$ over the resources. While we are not able to compute k^* exactly, we can efficiently compute some sufficiently small $k_0 \leq k^*$ with the property that an equilibrium for the k_0-integrally-splittable game allows us to determine the set of resources on which a player will put a positive amount of load in the atomic splittable equilibrium. Once these *support sets* are known, an atomic splittable equilibrium can be computed in polynomial time by solving a system of linear equations. This way, we can reduce the problem of computing the exact equilibrium for an atomic splittable game to computing an equilibrium for an associated k_0-integrally-splittable game.

The class of integrally-splittable congestion games has been studied before by Tran-Thanh et al. [47] for the case of player-independent convex cost functions and later by Harks et al. [23] (for the more general case of polymatroid strategy spaces and player-specific convex cost functions). In particular, Harks et al. devised an algorithm with running time $n^2 m(\delta/k_0)^3$, where n is the number of players, m the number of resources, and δ is an upper bound on the maximum demand of the players (cf. Theorem 5.2 [23]). As δ is encoded in binary, however, the algorithm is only pseudo-polynomial even for player-specific affine cost functions.

We devise a polynomial time algorithm for integrally-splittable singleton congestion games with player-specific affine cost functions. Our algorithm works as follows. For a game with initial packet size k_0, we start by finding an equilibrium for packet size $k = k_0 \cdot 2^q$ for some q of order $O(\log(\delta/k_0))$, satisfying only a part of the player-specific demands. Then we repeat the following two actions:

1. We halve the packet size from k to $k/2$ and construct a $k/2$-equilibrium using the k-equilibrium. Here, a k-equilibrium denotes an equilibrium for an integrally-splittable game with common packet size k. We show that this can be done in

polynomial time by repeatedly performing the following operations given a k-equilibrium:

(a) Among players who can improve, we find the player that benefits most by moving one packet of size $k/2$;
(b) If necessary, we perform a sequence of backward-shuffles of packets to correct the *load decrease* caused by the first packet movement (this is called a *backward path*);
(c) If necessary, we perform a sequence of forward-shuffles of packets to correct the load increase caused by the first packet movement (this is called a *forward path*);

(a)–(c) is iterated until a $k/2$-equilibrium for the currently scheduled demand is reached. For strategy profile x we define $\Delta(x)$ to be a vector that contains the cost for moving one packet to the currently cheapest resource, for each combination of a player and resource. We show that after each iteration $\Delta(x)$ lexicographically increases, which implies that we converge to a $k/2$-equilibrium.

2. For each player i we repeat the following step: if the current packet size k is smaller than the currently unscheduled demand of player i, we add one more packet for this particular player to the game and recompute the equilibrium. This part of the algorithm has also been used in the algorithm by Tran-Thanh et al. [47] and Harks et al. [23].
3. After q iterations, we have scheduled all demands and obtain an equilibrium for the desired packet size k_0.

Key to the analysis of the correctness and the running time of the algorithm are several structural results on the sensitivity of equilibria with respect to different integral packet sizes $k \in \mathbb{Q}_{>0}$ and $k/r \in \mathbb{Q}_{>0}$ for some $r \in \mathbb{N}$. Specifically, we derive bounds on the difference of resulting global load vectors as well as individual load vectors of players in any respective equilibrium. These sensitivity results may be of independent interest as they show how equilibria gradually behave in terms of the discretization granularity.

We use these structural insights to show that $\Delta(x)$ reaches a lexicographical maximum in a polynomial number of steps. Overall, compared to the existing algorithms of Tran-Thanh et al. [47] and Harks et al. [23], our algorithm has two main innovations: packet sizes are decreased exponentially (yielding polynomial running time in δ) and k-equilibrium computation for an intermediate packet size k is achieved via a careful construction of a sequence of single packet movements (backward- and forward paths) from a given $2k$-equilibrium (ensuring its polynomial length).

Multimarket Cournot Oligopolies. We then study the equilibrium computation problem for Cournot oligopolies. In the basic model of Cournot [14] introduced in 1838, firms produce homogeneous goods and sell them in a *common* market. The selling price of the goods depends on the total amount of goods that is offered in the market. Each firm aims to maximize its profit, which is equal to the revenue minus the production costs. In a *multimarket oligopoly* (cf. Bulow [8]), firms compete over a *set* of markets and each firm has access to a firm-specific subset of the markets.

For multimarket oligopolies, we develop a poly-time computable isomorphism mapping a multimarket Cournot competition game to an associated atomic splittable singleton congestion game. The isomorphism is payoff invariant (up to constants) and thus preserves equilibria in either games. As a consequence, we can apply the isomorphism and the polynomial time algorithm for atomic splittable congestion games to efficiently compute Cournot equilibria for models with firm-specific affine price functions and quadratic production costs. In addition, our analysis for integrally-splittable games also implies new bounds on the difference between real and integral Cournot equilibria complementing and extending recent results of Todd [44]. The case of affine price functions with quadratic cost functions is a well-studied model in economics, see Moulin et al. [35] and further references therein.

1.2 Related work

Discrete Congestion Games. As the first seminal work regarding the computational complexity of equilibrium computation in congestion games, Fabrikant et al. [18] showed that the problem of computing a pure Nash equilibrium is PLS-complete for network congestion games. Ackermann et al. [2] strengthened this result to hold even for network congestion games with linear cost functions. On the other hand, there are polynomial algorithms for symmetric network congestion games (cf. Fabrikant et al. [18]), for matroid congestion games with player-specific cost functions (Ackermann et al. [2,3]) and for so-called total unimodular congestion games (see Del Pia et al. [16]).

In particular, there is a pseudo-polynomial time algorithm that computes pure Nash equilibria for polymatroid congestion games with player-specific cost functions and polynomially bounded demands (Harks et al. [23]). As mentioned in Sect. 1.1, their results plays a significant role in this paper. The algorithm by Harks et al. starts with the trivial equilibrium for the game where all player-specific demands are zero. Then, they sequentially add packets to the game. After a packet is added, additional packet exchanges might be executed to recompute the equilibrium. For the special case of affine cost functions and singleton strategy spaces we construct an alternative algorithm that can compute equilibria in polynomial time.

Further results regarding the computation of approximate equilibria in congestion games can be found in Caragiannis et al. [9,10], Chien and Sinclair [12] and Skopalik and Vöcking [43].

Atomic Splittable Congestion Games. Atomic splittable congestion games on networks with player-independent cost functions have been studied (seemingly independently) by Orda et al. [38] and Haurie and Marcotte [24] and Marcotte [31]. Both lines of research mentioned that Rosen's existence result for concave games on compact strategy spaces implies the existence of pure Nash equilibria via Kakutani's fixed-point theorem. Cominetti et al. [13] presented the first upper bounds on the price of anarchy in atomic splittable congestion games. These were later improved by Harks [21] and finally shown to be tight by Schoppmann and Roughgarden [42].

For the computation of equilibria, Marcotte [31] proposed four numerical algorithms and showed local convergence results. Meunier and Pradeau [32] developed a pivoting-algorithm (similar to Lemke's algorithm) for nonatomic network congestion games with affine player-specific cost functions. Polynomial running time was, however, not shown and seems unlikely to hold. Gairing et al. [19] considered nonatomic routing games on parallel links with affine player-specific cost functions. They developed a convex potential function that can be minimized within arbitrary precision in polynomial time. Deligkas et al. [17] considered general concave games with compact action spaces and investigated algorithms computing an approximate equilibrium. Roughly speaking, they discretized the compact strategy space and use the Lipschitz constants of utility functions to show that only a finite number of representative strategy profiles need to be considered for obtaining an approximate equilibrium (see also Lipton et al. [30] for a similar approach). The running time of the algorithm, however, depends on an upper bound of the norm of strategy vectors, thus, implying only a pseudo-polynomial algorithm for our setting.

Note that the problem of computing pure Nash equilibria in atomic splittable congestion games with singleton strategies and affine cost functions can be written as a *linear complementary problem*, but does not seem to fall in any of the classes for which a solution can be found in polynomial time.

Multimarket Cournot Oligopolies The existence of equilibria in single market Cournot models (beyond quasi-polynomial utility functions) has been studied extensively in the past decades (see Vives [49] for a good survey). E.g., Novshek [37] proved that equilibria exists whenever the marginal revenue of each firm is decreasing in the aggregate quantities of the other firms. Then, several works (cf. Topkis [45], Amir [4], Kukushkin [29], Milgrom and Roberts [33], Milgrom and Shannon [34], Topkis [46] and Vives [48]) proved existence of equilibria when the underlying game is supermodular, i.e., when the strategy space forms a lattice and the marginal utility of each firm is increasing in any other firm's output. Using supermodularity, one can obtain existence results without assuming that the utility functions are quasi-convex. Very recently, Todd [44] considered Cournot competition on a single market, where the price functions are linear and cost functions are quadratic. For such games, he proved that equilibria exist and can be computed in time $O(n\log(n))$, where n denotes the number of firms. Additionally, he analyzed the maximum differences of production quantities of real and integral equilibria, respectively.

Abolhassani et al. [1] devised several polynomial time algorithms for multimarket Cournot oligopolies, partly using algorithms for solving nonlinear complementarity problems. In contrast to our work, they assume that price functions are firm-independent. Bimpikis et al. [7] provided a characterization of the production quantities at the unique equilibrium, when price functions are player-independent and concave, and cost functions are convex. They study the impact of changes in the competition structure on the firm's profit. This framework can be used to either identify opportunities for collaboration and expanding in new markets. Harks and Klimm [22] studied the existence of Cournot equilibria, under the condition that each firm can only sell its items to a limited number of markets simultaneously. They proved that equilibria exist when production cost functions are convex, marginal return functions strictly decrease for strictly increased own quantities and non-decreased aggregated quantities and when for every firm, the firm specific market reaction functions across markets are identical up to market-specific shifts.

2 Preliminaries

Atomic Splittable Singleton Games. An atomic splittable singleton congestion game is defined by a tuple: $\mathcal{G} := \big(N, E, (d_i)_{i\in N}, (E_i)_{i\in N}, (c_{i,e})_{i\in N, e\in E_i}\big)$, where $E = \{e_1, \dots, e_m\}$ is a finite set of resources and $N = \{1, \dots, n\}$ is a finite set of players. Each player $i \in N$ is associated with a demand $d_i \in \mathbb{Q}_{\geq 0}$ and a set of allowable resources $E_i \subseteq E$. A strategy for player $i \in N$ is a (possibly fractional) distribution of the demand d_i over the singletons in E_i. Thus, one can represent the strategy space of every player $i \in N$ by the polytope:

$$\mathcal{S}_i(d_i) := \left\{ x_i \in \mathbb{R}_{\geq 0}^{|E_i|} \mid \sum_{e\in E_i} x_{i,e} = d_i \right\}.$$

The combined strategy space is denoted by $\mathcal{S} := \times_{i\in N} \mathcal{S}_i(d_i)$ and $x = (x_i)_{i\in N}$ is the overall strategy profile. We define $x_{i,e} := (x_i)_e$ as the load of player i on $e \in E_i$ and $x_{i,e} = 0$ when $e \in E \backslash E_i$. The total load on resource e is given as $x_e := \sum_{i\in N} x_{i,e}$. Resources have player-specific affine cost functions $c_{i,e}(x_e) = a_{i,e}x_e + b_{i,e}$ with $a_{i,e} \in \mathbb{Q}_{>0}$ and $b_{i,e} \in \mathbb{Q}_{\geq 0}$ for all $i \in N$ and $e \in E$. The total cost of player i in strategy distribution x is defined as: $\pi_i(x) = \sum_{e\in E_i} c_{i,e}(x_e)\, x_{i,e}$. We write $\mathcal{S}_{-i}(d_{-i}) = \times_{j\neq i} \mathcal{S}_j(d_j)$ and we write $x = (x_i, x_{-i})$ for each $i \in N$, meaning that $x_i \in \mathcal{S}_i(d_i)$ and $x_{-i} \in \mathcal{S}_{-i}(d_{-i})$. A strategy profile x is an *equilibrium* if $\pi_i(x) \leq \pi_i(y_i, x_{-i})$ for all $i \in N$ and $y_i \in \mathcal{S}_i(d_i)$. A pair $\big(x, (y_i, x_{-i})\big) \in \mathcal{S} \times \mathcal{S}$ is called an *improving move* of player i, if $\pi_i(x_i, x_{-i}) > \pi_i(y_i, x_{-i})$. We define $\mu_{i,e}(x) = c_{i,e}(x_e) + x_{i,e}c'_{i,e}(x_e) = a_{i,e}(x_e + x_{i,e}) + b_{i,e}$ to be the *marginal cost* for player i on resource e. We obtain the following sufficient and necessary equilibrium condition.

Lemma 1 (cf. Harks [21]) *Strategy profile x is an equilibrium if and only if the following holds for all $i \in N$: if $x_{i,e} > 0$, then $\mu_{i,e}(x) \leq \mu_{i,f}(x)$ for all $f \in E_i$.*

Using that the strategy space is compact and cost functions are convex, Kakutani's fixed point theorem implies the existence of an equilibrium. Uniqueness is proven by Richmann and Shimkin [40] and Bhaskar et al. [5].

Game $\mathcal{G}$ is called symmetric whenever $E_i = E$ for all $i \in N$. We can project any asymmetric game $\mathcal{G}$ on a symmetric game $\mathcal{G}^*$ by for all $i \in N$ and $e \in E\backslash E_i$ setting $c^*_{i,e}(x_e)$ to $c_{i,e}(x_e)$ whenever $e \in E_i$, and to $x_e + (n+2)(a_{\max})^2$ otherwise. Here,

$$a_{\max} := \max\{\{a_{i,e}, b_{i,e} \mid i \in N, e \in E_i\}, \{d_i \mid i \in N\}, 1\}.$$

In this case $\mu_{i,e}(0) \geq \mu_{i,f}(x_e)$ for any $e \in E\backslash E_i$, $f \in E_i$, $i \in N$ and $x \in \mathcal{S}$. Thus, in an equilibrium y for game $\mathcal{G}^*$ no player i puts load on any resource $e \in E\backslash E_i$. Hence, y is also an equilibrium for game $\mathcal{G}$. In the rest of this paper we project every asymmetric game on a symmetric game using the construction above.

Integral Singleton Games. A k-integral singleton game is compactly defined by the tuple $\mathcal{G}_k := \big(N, E, (d_i)_{i\in N}, (c_{i,e})_{i\in N, e\in E}\big)$ with $k \in \mathbb{Q}_{>0}$. Here, players cannot split their load fractionally, but only in multiples of k. Assume d_i is a multiple of k, then the strategy space for player i is the following set:

$$\mathcal{S}_i(d_i, k) := \left\{x_i \in \mathbb{Q}_{\geq 0}^{|E|} \mid x_{i,e} = kq_{i,e}, q_{i,e} \in \mathbb{N}_{\geq 0}, \sum_{e \in E} x_{i,e} = d_i\right\}.$$

In this game, k is also called the *packet size*. When k and d_i are clear from the context, we refer to $\mathcal{S}_i(d_i, k)$ as $\mathcal{S}_i$. When E, N and $(c_{i,e})_{i \in N, e \in E}$ are clear from the context, we also refer to the game as $\mathcal{G}_k((d_i)_{i \in N})$. For player-specific affine cost functions the (discrete) marginal increase and decrease are defined as follows:

$$\mu_{i,e}^{+k}(x) = (x_{i,e} + k)c_{i,e}(x_e + k) - x_{i,e}c_{i,e}(x_e), \tag{1}$$

$$\mu_{i,e}^{-k}(x) = \begin{cases} x_{i,e}c_{i,e}(x_e) - (x_{i,e} - k)c_{i,e}(x_e - k), \text{ we half the} & \text{if } x_{i,e} > 0 \\ -\infty, & \text{if } x_{i,e} \leq 0. \end{cases} \tag{2}$$

Here, $\mu_{i,e}^{+k}(x)$ is the cost for player i to add one packet of size k to resource e and $\mu_{i,e}^{-k}(x)$ is the gain for player i of removing a packet from resource e. Assuming cost functions are affine, we write $\mu_{i,e}^{+k}(x) = k(a_{i,e}(x_e + x_{i,e} + k) + b_{i,e})$ and $\mu_{i,e}^{-k}(x) = k(a_{i,e}(x_e + x_{i,e} - k) + b_{i,e})$ if $x_{i,e} > 0$.

Lemma 2 (cf. Groenevelt [20]) *Strategy profile x is an equilibrium in a k-integral congestion game if and only if for all $i \in N$ it holds that if $x_{i,e} > 0$, then also $\mu_{i,e}^{-k}(x) \leq \mu_{i,f}^{+k}(x)$ for all $f \in E$.*

Define $\mu_{i,\min}^{+k}(x) := \min_{e \in E}\{\mu_{i,e}^{+k}(x)\}$ and $\mu_{i,\max}^{-k}(x) := \max_{e \in E}\{\mu_{i,e}^{-k}(x)\}$. Then x is an equilibrium in a k-integral game if and only if $\mu_{i,\max}^{-k}(x) \leq \mu_{i,\min}^{+k}(x)$ for all $i \in N$.

We also introduce some new notation. For two vectors $x_i, y_i \in \mathbb{R}^{|E|}$, we define

$$H(x_i, y_i) := \sum_{e \in E} |x_{i,e} - y_{i,e}|$$

to be their Hamming distance. For two strategies x, y, we write $H(x, y) := \sum_{e \in E} |x_e - y_e|$. For two resources $e^-, e^+ \in E$, we denote $y_i = (x_i)_{e^- \to e^+}$ if it holds that $y_{i,e} = x_{i,e}$ for all $e \in E \backslash \{e^-, e^+\}$, $y_{i,e^-} = x_{i,e^-} - k$ and $y_{i,e^+} = x_{i,e^+} + k$. If x is a strategy profile for some game $\mathcal{G}_k$ and $y_i = (x_i)_{e^- \to e^+}$, we denote $(y_i, x_{-i}) = x_{i:e^- \to e^+}$.

We define a *restricted improving move* and a *restricted best response* as follows:

Definition 1 Let x be a strategy profile for game $\mathcal{G}_k((d_i)_{i \in N})$.

1. A strategy x_i' is called a *restricted improving move* to x for player i, if

$$x_i' \in \{y_i \in \mathcal{S}_i(d_i, k) \mid H(x_i, y_i) = 2k \text{ and } \pi_i(y_i, x_{-i}) < \pi_i(x_i, x_{-i})\}.$$

2. A strategy x_i' is called a *restricted best response* to x for player i, if

$$x_i' \in \arg\min_{y_i \in \mathcal{S}_i(d_i, k)} \{\pi_i(y_i, x_{-i}) \mid H(x_i, y_i) = 2k\}.$$

Note that both a restricted improving move and a restricted best response can be executed by moving a single packet.

3 Sensitivity results for equilibria

In Sect. 4, we show that computing an equilibrium for atomic splittable games can be reduced to the problem of computing an equilibrium of an associated *integrally-splittable* game with small enough packet size. For such a class of discrete games, we will develop a polynomial time *scaling algorithm*, where we write the total demand as a power of two and then iteratively scale down the allowed packet size and recompute equilibria for the resulting integrally-splittable games. The key for the well-definedness and further analysis of this algorithm is a structural result on the sensitivity of equilibria for integrally-splittable games with respect to changed packet sizes. In the following, we derive such sensitivity results between equilibria of an integrally-splittable game $\mathcal{G}_k$ with packet size $k \in \mathbb{Q}_{>0}$ and those of a game $\mathcal{G}_{\frac{k}{r}}$ with $r \in \mathbb{N}$. These results may be of independent interest in the area of comparative statics, where the influence of parameters w.r.t. to resulting equilibria are analyzed.

Theorem 1 *Let x_k be an equilibrium for game $\mathcal{G}_k$, and $x_{k/r}$ be an equilibrium for game $\mathcal{G}_{k/r}$. Then $|(x_k)_e - (x_{k/r})_e| < (1 + \frac{1}{r})mk$ for all $e \in E$.*

Proof In order to prove the theorem we need to show that both:

1. $(x_k)_e - (x_{k/r})_e < (1 + \frac{1}{r})mk$ and
2. $(x_{k/r})_e - (x_k)_e < (1 + \frac{1}{r})mk$.

As the proofs for both statements are very similar, we only prove the first statement here. On the contrary, assume that there exists a resource e_1 with $(x_k)_{e_1} - (x_{k/r})_{e_1} \geq (1 + \frac{1}{r})mk$. We introduce two player sets N_e^+, N_e^- for every resource $e \in E$, where:

$$N_e^+ = \{i \in N | (x_k)_{i,e} > (x_{k/r})_{i,e}\} \text{ and } N_e^- = \{i \in N | (x_k)_{i,e} \leq (x_{k/r})_{i,e}\}.$$

Note that for every $i \in N_{e_1}^+$, we have:

$$(x_{k/r})_{e_1} + (x_{k/r})_{i,e_1} < (x_k)_{e_1} + (x_k)_{i,e_1} - (1 + \tfrac{1}{r})mk. \tag{3}$$

Using the player sets, we obtain:

$$\sum_{i \in N_{e_1}^+} ((x_k)_{i,e_1} - (x_{k/r})_{i,e_1}) + \sum_{i \in N_{e_1}^-} ((x_k)_{i,e_1} - (x_{k/r})_{i,e_1}) = (x_k)_{e_1} - (x_{k/r})_{e_1} \geq (1 + \tfrac{1}{r})mk.$$

As $\sum_{i \in N_{e_1}^-} ((x_k)_{i,e_1} - (x_{k/r})_{i,e_1}) \leq 0$, we have:

$$\sum_{i \in N_{e_1}^+} ((x_k)_{i,e_1} - (x_{k/r})_{i,e_1}) \geq (1 + \tfrac{1}{r})mk.$$

The total load distributed by a player does not change, therefore:

$$\sum_{f \neq e_1} \sum_{i \in N_{e_1}^+} ((x_k)_{i,f} - (x_{k/r})_{i,f}) \leq -(1 + \tfrac{1}{r})mk.$$

For every resource $f \in E \backslash \{e_1\}$ we further subdivide $N_{e_1}^+$ in two parts $N_{e_1}^+ \cap N_f^-$ and $N_{e_1}^+ \cap N_f^+$ and obtain:

$$\sum_{f \neq e_1} \sum_{i \in N_{e_1}^+ \cap N_f^-} ((x_k)_{i,f} - (x_{k/r})_{i,f}) + \sum_{i \in N_{e_1}^+ \cap N_f^+} ((x_k)_{i,f} - (x_{k/r})_{i,f}) \leq -(1 + \tfrac{1}{r})mk.$$

Using the definition of N_f^+, we obtain:

$$\sum_{f \neq e_1} \sum_{i \in N_{e_1}^+ \cap N_f^-} ((x_k)_{i,f} - (x_{k/r})_{i,f}) \leq -(1 + \tfrac{1}{r})mk. \tag{4}$$

As $(x_k)_{e_1} - (x_{k/r})_{e_1} \geq (1 + \frac{1}{r})mk$, we have $\sum_{f \neq e_1} ((x_k)_f - (x_{k/r})_f) \leq -(1 + \frac{1}{r})mk$. Therefore:

$$\sum_{f \neq e_1} \sum_{i \in N_{e_1}^+ \cap N_f^-} ((x_k)_f - (x_{k/r})_f) \leq -|N_{e_1}^+ \cap N_f^-|(1 + \tfrac{1}{r})mk.$$

We add this to Eq. (4) to obtain the following:

$$\sum_{f \neq e_1} \sum_{i \in N_{e_1}^+ \cap N_f^-} ((x_k)_f - (x_{k/r})_f) + ((x_k)_{i,f} - (x_{k/r})_{i,f})$$
$$\leq -(|N_{e_1}^+ \cap N_f^-| + 1)(1 + \tfrac{1}{r})mk.$$

By using the pigeonhole principle on the number of resources $f \in E \backslash \{e_1\}$, there exists an $f \in E \backslash \{e_1\}$ such that:

$$\sum_{i \in N_{e_1}^+ \cap N_f^-} ((x_k)_f - (x_{k/r})_f) + ((x_k)_{i,f} - (x_{k/r})_{i,f}) < -(|N_{e_1}^+ \cap N_f^-| + 1)(1 + \tfrac{1}{r})k.$$

Using the pigeonhole principle again on the number of players in $N_{e_1}^+ \cap N_f^-$, there exists an $i \in N_{e_1}^+ \cap N_f^-$ such that

$$((x_k)_f - x_f) + ((x_k)_{i,f} - x_{i,f}) < (1 + \tfrac{1}{r})k. \quad (5)$$

We combine Eqs. (3, 5) and the fact that x_k is an equilibrium for packet size k to obtain:

$$\mu_{i,e_1}^{+k/r}(x_{k/r}) < \frac{1}{r}\mu_{i,e_1}^{-k}(x_k) \le \frac{1}{r}\mu_{i,f}^{+k}(x_k) \le \mu_{i,f}^{-k/r}(x_{k/r}). \quad (6)$$

Hence, we have found a player i that has a restricted improving move in $x_{k/r}$, which contradicts the fact that $x_{k/r}$ is an equilibrium strategy. □

With

$$\lim_{k\to 0}\frac{1}{k}\mu_{i,e}^{+k}(x) = \lim_{k\to 0}\frac{1}{k}\mu_{i,e}^{-k}(x) = \mu_{i,e}(x),$$

we immediately obtain the following statement from Theorem 1.

Corollary 1 *Let x be the unique equilibrium for an atomic splittable game, and x_k be an equilibrium for a k-integral splittable game. Then $|(x_k)_e - x_e| < mk$ for all $e \in E$.*

We obtain a similar result for player-specific load differences:

Theorem 2 *Let x_k be an equilibrium for game $\mathcal{G}_k$, and $x_{k/r}$ be an equilibrium for game $\mathcal{G}_{k/r}$. Then $|(x_k)_{i,e} - (x_{k/r})_{i,e}| < (1 + \frac{1}{r})m^2k$ for all $e \in E$.*

Proof In order to prove the theorem we need to show that both:

1. $(x_k)_{i,e} - (x_{k/r})_{i,e} < (1 + \frac{1}{r})m^2k$ and
2. $(x_{k/r})_{i,e} - (x_k)_{i,e} < (1 + \frac{1}{r})m^2k$

We again only prove the first statement here. Assume by contradiction that there exists a resource e_1 with $(x_k)_{i,e_1} - (x_{k/r})_{i,e_1} \ge (1 + \frac{1}{r})m^2k$. By Theorem 1, we know that for all $i \in N$ and $e \in E$ it holds that $(x_{k/r})_e \le (x_k)_e + (1 + \frac{1}{r})mk$. Thus:

$$(x_{k/r})_{e_1} + (x_{k/r})_{i,e_1} \le (x_k)_{e_1} + (x_k)_{i,e_1} - (m-1)(1 + \tfrac{1}{r})mk. \quad (7)$$

The total load distributed by all player is the same in both $x_{k/r}$ and x_k. Thus, we obtain:

$$\textstyle\sum_{e \ne e_1}((x_{k/r})_e + (x_{k/r})_{i,e}) \ge \sum_{e \ne e_1}((x_k)_e + (x_k)_{i,e}) + (m-1)(1 + \tfrac{1}{r})mk.$$

By the pigeonhole principle, there must exist at least one resource $f \in E$ such that:

$$(x_{k/r})_f + (x_{k/r})_{i,f} \ge (x_k)_f + (x_k)_{i,f} + (1 + \tfrac{1}{r})mk. \quad (8)$$

Note that $(x_{k/r})_{i,f} > 0$, as $(x_{k/r})_{i,f} = 0$ implies $(x_{k/r})_f > (x_k)_f + (1+\frac{1}{r})mk$, which contradicts the fact that $|(x_k)_f - (x_{k/r})_f| \leq (1+\frac{1}{r})mk$. We obtain:

$$\mu_{i,e_1}^{+k/r}(x_{k/r}) < \frac{1}{r}\mu_{i,e_1}^{-k}(x_k) \leq \frac{1}{r}\mu_{i,f}^{+k}(x_k) \leq \mu_{i,f}^{-k/r}(x_{k/r}). \tag{9}$$

As $(x_{k/r})_{i,f} > 0$, player i has a restricted improving move from resource f to resource e_1. This contradicts the fact that $x_{k/r}$ is an equilibrium strategy. Hence, it cannot happen that $(x_k)_{i,e_1} - (x_{k/r})_{i,e_1} \geq (1+\frac{1}{r})m^2k$. Similarly, we can find a restricted improving move whenever $(x_{k/r})_{i,e} - (x_k)_{i,e} \geq (1+\frac{1}{r})m^2k$. Thus, for any equilibrium x_k for game $\mathcal{G}_k$, and equilibrium $x_{k/r}$ for game $\mathcal{G}_{k/r}$, we have that $|(x_k)_{i,e} - (x_{k/r})_{i,e}| < (1+\frac{1}{r})m^2k$ for all $e \in E$. □

Again, we immediately obtain the following statement from Theorem 2.

Corollary 2 *Let x be the unique equilibrium for an atomic splittable game, and x_k be an equilibrium for the corresponding k-integral splittable game. Then $|(x_k)_{i,e} - x_{i,e}| < m^2k$ for all $i \in N$ and $e \in E$.*

To complement Theorems 1 and 2, we provide a lower bound example where

$$|(x_k)_{i,e} - (x_{k/r})_{i,e}| = |(x_k)_e - (x_{k/r})_e| = (m-1)\frac{k}{r}.$$

Example 1 Consider a $k-$splittable congestion game $\mathcal{G}_k$ with player set $N = \{1\}$ and resource set $\{e_1, \ldots, e_m\}$. Let $d_1 = (m-1)k$, and the cost functions are defined as follows:

$$c_{1,e}(x_e) := \begin{cases} \frac{x_e}{2(r-1)(m-1)} & \text{if } e = e_m, \\ x_e & \text{otherwise.} \end{cases}$$

In game $\mathcal{G}_k$, a best response x_k for player 1 is to put all $m-1$ packets on resource e_m. Alternatively, if the packet size is $\frac{k}{r}$ instead of k, strategy

$$x_{k/r} := (\tfrac{k}{r}, \ldots, \tfrac{k}{r}, (m-1)(k - \tfrac{k}{r})),$$

is an equilibrium strategy for player 1.

We end this section with a corollary that follows from the proof of Theorem 1. We need this specific statement in Sect. 6.

Corollary 3 *Let x_k be an equilibrium for game $\mathcal{G}_k$, and $x_{k/2}$ be an arbitrary strategy profile for game $\mathcal{G}_{k/2}$. If for some $e \in E$ it holds that $(x_k)_e - (x_{k/2})_e \geq \frac{3}{2}mk$, then there exists a player $i \in N$ that has a restricted improving move in game $\mathcal{G}_{k/2}$, where a packet of size $k/2$ is moved from some resource f to resource e. If for some $e \in E$, it holds that $(x_{k/2})_e - (x_k)_e \geq \frac{3}{2}mk$, then there exists a player $i \in N$ that has a restricted improving move in game $\mathcal{G}_{k/2}$, where a packet of size $k/2$ is moved from resource e to another resource f.*

Proof Recall that the proof of Theorem 1 was done by contradiction: For any resource e for which $(x_k)_e - (x_{k/r})_e \geq (1 + \frac{1}{r})mk$ holds, we showed that there is a player i that has a restricted improving move in $x_{k/r}$ moving a single packet of size k/r to resource e from some other resource f. Similarly, whenever $(x_{k/r})_e - (x_k)_e \geq (1 + \frac{1}{r})mk$ holds, there is a player i that has a restricted improving move in $x_{k/r}$ moving a single packet of size k/r from e to some other resource f. If we take $r = 2$, the corollary follows. □

4 Reduction to integrally-splittable games

We show that the problem of finding an equilibrium for an atomic splittable game reduces to the problem of finding an equilibrium for a k_0-integral game for some $k_0 \in \mathbb{Q}_{>0}$.

Theorem 3 *Let x be the unique equilibrium of an atomic splittable singleton game $\mathcal{G}$. Then, there exists a $k^* \in \mathbb{Q}_{>0}$ such that x is also the unique equilibrium for the k^*-integral splittable game $\mathcal{G}_{k^*}$.*

Proof We define the support set $I_i := \{e \in E \mid x_{i,e} > 0\}$ for each player $i \in N$. Lemma 1 implies that if x is an equilibrium, and $x_{i,e}, x_{i,f} > 0$, then $\mu_{i,e}(x) = \mu_{i,f}(x)$.

Define $p := \sum_{i \in N} |I_i| \leq nm$. Then, if the correct support set I_i of each player is known, the equilibrium can be computed by solving the following set of p linear equations on p variables.

1. For every player we have an equation that makes sure the demand of that player is satisfied. Thus, for each player $i \in N$ we have $\sum_{e \in I_i} x_{i,e} = d_i$.
2. For every player $i \in N$, there are $|I_i| - 1$ equations of type $\mu_{i,e}(x) = \mu_{i,f}(x)$ for $e, f \in I_i$, which we write as $a_{i,e}(x_e + x_{i,e}) - a_{i,f}(x_f + x_{i,f}) = b_{i,e} - b_{i,f}$. Note that x_e is not an extra variable, but an abbreviation for $\sum_{i \in N} x_{i,e}$.

We refer to this set of equalities as $Ax = b$, where A is a $p \times p$ matrix. Note that as the equilibrium exists and is unique, matrix A is non-singular. Then, using Cramer's Rule, the unique solution of this system is given by: $x_{i,e} = \det(A_{i,e})/\det(A) = |\det(A_{i,e})|/|\det(A)|$, where $A_{i,e}$ is the matrix formed by replacing the column that corresponds to value $x_{i,e}$ in A by b. We define $Q := \{\{a_{i,e}, b_{i,e} \mid i \in N, e \in E_i\} \cup \{d_i \mid i \in N\} \cup \{1\}\}$ as the set of input values and $a_{\text{gcd}} := \max\{a \in \mathbb{Q}_{>0} \mid \forall q \in Q,\ \exists \ell \in \mathbb{N}$ such that $q = a \cdot \ell\}$ as the *greatest common divisor* of Q.

Then, as all values in A and b depend on adding and subtracting values in Q, $|\det(A_{i,e})|$ is an integer multiple of $(a_{\text{gcd}})^p$ and, hence, an integer multiple of $(a_{\text{gcd}})^{nm}$. Thus, all player-specific loads are integer multiples of $(a_{\text{gcd}})^{nm}/|\det(A)|$ and, hence, if we define $k^* = (a_{\text{gcd}})^{nm}/|\det(A)|$, x is an equilibrium for the k^*-integral splittable game. Note that we can compute a_{gcd} in running time $O(nm \log a_{\max})$.

It is left to prove that x is the unique equilibrium for the k^*-integral splittable game. Assume, on the contrary, that there are two different equilibria x and y, where x is the equilibrium for the atomic splittable game. We define two different resource sets: $E^+ := \{e \in E | x_e > y_e\}$ and $E^- := \{e \in E | x_e \leq y_e\}$, and two corresponding player sets $N^+ := \{i \in N | \sum_{e \in E^+} (x_{i,e} - y_{i,e}) > 0\}$ and

$N^- := \{i \in N \mid \sum_{e \in E^-} \left(x_{i,e} - y_{i,e}\right) < 0\}$. Clearly $N^+ \neq \emptyset$, and as each player distributes the same amount of load in x and y we have $N^+ = N^-$. Choose a player $i \in N^+ = N^-$, then there exist resources e and f such that $x_e > y_e$, $x_{i,e} > y_{i,e}$, $x_f \leq y_f$ and $x_{i,f} < y_{i,f}$. Then, we have:

$$\begin{aligned}
\mu_{i,e}^{+k^*}(y) &< \mu_{i,e}^{+k^*}(x) - (k^*)^2 a_{i,e} && (\text{as } x_e \geq y_e + k^* \text{ and } x_{i,e} \geq y_{i,e} + k^*)\\
&= k^* \cdot \mu_{i,e}(x) && (\text{by rewriting})\\
&\leq k^* \cdot \mu_{i,f}(x) && (\text{as } x \text{ is the atomic splittable equilibrium})\\
&= \mu_{i,f}^{-k^*}(x) + (k^*)^2 a_{i,f} && (\text{by rewriting})\\
&\leq \mu_{i,f}^{-k^*}(y). && (\text{as } x_f \leq y_f \text{ and } x_{i,f} \leq y_{i,f} - k^*)
\end{aligned}$$

This contradicts the fact that y is an equilibrium. Thus, x is the unique k^*-integral splittable equilibrium. □

Note that we do not know matrix A beforehand, but we do know that $2a_{\max}$ is an upper bound on the values occurring in A. Using Hadamard's inequality we find that $|\det(A)| \leq (2a_{\max})^{nm}(nm)^{nm/2}$. Hence, we can find a lower bound of k^*:

$$k^* \geq a_{gcd}^{nm}/((2a_{\max})^{nm}(nm)^{nm/2}).$$

By Corollaries 1 and 2, we know that for any atomic splittable equilibrium x and any k-integral-splittable equilibrium x_k, there exist bounds on $|x_e - (x_k)_e|$ and $|x_{i,e} - (x_k)_{i,e}|$ in terms of k and m. Thus, if we compute an equilibrium for a sufficiently small k_0, this k_0-integral-splittable equilibrium should be fairly similar to the unique k^*-integral splittable equilibrium. Hence, it enables us to find the correct support sets. Then, given the equilibrium for some sufficiently small k_0, we are able compute the correct support set of each player and compute the exact atomic splittable equilibrium by solving system $Ax = b$ as described earlier.

Theorem 4 *Given an atomic splittable congestion game $\mathcal{G}$ and an equilibrium x_{k_0} for k_0-splittable game $\mathcal{G}_{k_0}$, where: $k_0 := (a_{gcd}^{nm})/(2m^2\lceil(2a_{\max})^{nm}(nm)^{nm/2}\rceil)$. We can compute in $O((nm)^3)$ the unique atomic splittable equilibrium x for game $\mathcal{G}$.*

Proof First note that all demands d_i are integer multiples of k_0, as d_i is an integer multiple of a_{gcd}, and both $2m^2$ and $\lceil(2a_{\max})^{nm}(nm)^{nm/2}\rceil$ are integers. Theorem 3 implies that there exists a k^* such that the atomic splittable equilibrium is also an equilibrium for the k^*-integral splittable game. In the following we show that there is a load-threshold m^2k_0 that enables us to decide whether or not a resource receives any demand from player i in the equilibrium of the atomic splittable game.

1. If $(x_{k_0})_{i,e} < m^2k_0$, then $x_{i,e} = 0$. Assume by contradiction that $x_{i,e} > 0$. Remember that the atomic splittable equilibrium is also a k^*-equilibrium and thus, if $x_{i,e} > 0$, then the inequality $x_{i,e} \geq k^*$ must hold. We obtain $x_{i,e} - (x_{k_0})_{i,e} > k^* - m^2k_0 \geq m^2k_0$, which contradicts Corollary 2. Thus, $x_{i,e} = 0$.
2. If $(x_{k_0})_{i,e} \geq m^2k_0$, then we prove that $x_{i,e} > 0$. On the contrary, we assume that $x_{i,e} = 0$. In this case we have $(x_{k_0})_{i,e} - x_{i,e} \geq m^2k_0$, which contradicts Corollary 2. Thus, $x_{i,e} > 0$.

Hence, given an equilibrium (x_{k_0}) for k_0-splittable game $\mathcal{G}_{k_0}$, we can compute the correct support sets $I_i = \{e \in E \mid (x_{k_0})_{i,e} \geq m^2 k_0\}$ for all $i \in N$. Given the correct support sets, we can easily compute the correct, exact equilibrium by solving the system $Ax = b$ of at most nm linear equations in running time $O((nm)^3)$ using Gaussian elimination [36]. □

It is left to compute an equilibrium x_{k_0} for integral game $\mathcal{G}_{k_0}$. Such integral games have been studied in the literature before, see Harks et al. [23]. In particular, [23, Algorithm 1] has running time $O(nm(\delta/k_0)^3)$. Here, δ is an upper bound on the player-specific demands. In general, δ is not bounded in k_0, thus, the running time is not polynomially bounded in the size of the input.

5 A polynomial algorithm for integral games

The goal of this section is to develop a *polynomial time* algorithm that computes an equilibrium for any k-integral splittable singleton game with player-specific affine cost functions. We use elements of [47, Algorithm 1] and [23, Algorithm 1] to construct a new algorithm with polynomial running time $O(n^2 m^{14} \log(\delta/k))$. This algorithm works as follows. For a game with initial packet size k_0, we start by finding an equilibrium for packet size $k = k_0 \cdot 2^q$ for some q of order $O(\log(\delta/k_0))$, satisfying only a part of the player-specific demands. Then we repeat the following two actions:

1. **Subroutine** Restore. We half the packet size from k to $k/2$ and construct a $k/2$-equilibrium using the k-equilibrium. Here, a k-equilibrium denotes an equilibrium for an integrally-splittable game with common packet size k.
2. **Subroutine** Add. For each player i we repeat the following step: if the current packet size k is smaller than the currently unscheduled demand of player i, we add one more packet for this particular player to the game and recompute the equilibrium. This part of the algorithm has also been used in the algorithm by Tran-Thanh et al. [47] and Harks et al. [23].
3. After q iterations, we have scheduled all demands and obtain an equilibrium for the desired packet size k_0.

We describe the two subroutines Add and Restore.

5.1 Add

The first subroutine, Add, is described in Algorithm 1 and consists of lines 4-10 of [23, Algorithm 1]. Given an equilibrium x_k for game $\mathcal{G}_k((d_i)_{i\in N})$, it computes an equilibrium for the game, where the demand for some player j is increased by a packet of size k. First it decides on the best resource f for player j to put her new packet. In effect, the load on resource f increases and only those players with $x_{i,f} > 0$ can potentially decrease their cost by a deviation. In this case, Harks et al. proved in [23, Theorem 3.2] that a best response y_i can be obtained by a restricted best response moving a packet away from f. Formally, if player i can potentially decrease her cost, then there exists a resource $e \in E$ such that

$$(x_i)_{f \to e} \in \arg \min_{x_i' \in \mathcal{S}_i(d_i,k)} \{\pi_i(x_i', x_{-i})\}$$

Thus, only one packet is moved throughout, preserving the invariant that only players using a resource to which the packet is moved may have an incentive to profitably deviate.

Algorithm 1: Subroutine ADD($x, j, \mathcal{G}_k((d_i)_{i\in N})$)

Input: equilibrium x_k for $\mathcal{G}_k((d_i)_{i\in N})$, player j
Output: equilibrium x_k' for $\mathcal{G}_k((d_i')_{i\in N})$, where $d_j' \leftarrow d_j + k$; $d_i' \leftarrow d_i$ for all $i \in N \setminus \{j\}$

```
x ← x_k; d'_j ← d_j + k; S'_j ← S_j(d'_j, k); d'_i ← d_i for all i ∈ N \ {j}
Choose f ∈ arg min{μ^{+k}_{j,e}(x)};
x_{j,f} ← x_{j,f} + k;
while ∃i ∈ N who can improve in G_k do
    Compute a restricted best response y_i ∈ S'_i;
    x_i ← y_i;
end
x'_k ← x;
return x'_k
```

5.2 RESTORE

The second subroutine, RESTORE, takes as input an equilibrium x_{2k} for packet size $2k$ and game $\mathcal{G}_k((d_i)_{i\in N})$, and constructs an equilibrium for packet size k. In this algorithm, we compute *backward/forward paths of restricted best responses*. In a backward path (Algorithm 2), we are given a resource e_1^- and a strategy profile b_1. In iteration q, we decide if there exists a player i that has a restricted best response from some e_{q+1}^- to e_q^-, and if so, we define $b_{q+1} \leftarrow (b_q)_{i:e_{q+1}^- \to e_q^-}$. If no player has a restricted best response to resource e_q^-, and if $(b_q)_{e_q^-} > (x_{2k})_{e_q^-} - 2mk$, we end our backward path. Else, we look for a player that has a restricted improving move in which she shifts one packet from some e_{q+1}^- to e_q^-, and then continue the backward path. Note that in each step we preserve the invariant that $H(b_1, b_q) \in \{0, 2k\}$.

A forward path is very similar to a backward path, but we change the perspective. Thus, given a resource e_q^+ and a strategy profile f_q, we check in iteration q if there exists a player that has a restricted best response from e_q^+ to some e_{q+1}^+.

We are now ready to define subroutine RESTORE. The vector x is initialized by an equilibrium x_{2k} and, while x is not an equilibrium for $\mathcal{G}_k$, we iterate the following: Among players who can improve, we find the player j that benefits most from a restricted best response. We carry out a restricted best response for player j and move a packet from some resource e_1^- to some e_1^+. Then we compute a backward path, starting in resource e_1^-. If the resulting strategy profile has Hamming distance zero

Algorithm 2: BP(x_{2k}, b_1, e_1^-) : A backward path of restricted best responses.

Input: equilibrium x_{2k} for game $\mathcal{G}_{2k}$, strategy profile b_1 for game $\mathcal{G}_k$ and a resource e_1^-.
Output: Strategy profile b_q for game $\mathcal{G}_k$ and resource e_q^-.

Initialize $q \leftarrow 1$;
while $\{i \in N \mid e_q^- \in \arg\min\{\mu_{i,e}^{+k}(b_q)\}$ *and* $\mu_{i,e_q^-}^{+k}(b_q) < \max_{e \in E}\{\mu_{i,e}^{-k}(b_q)\}\} \neq \emptyset \vee ((b_q)_{e_q^-} \leq (x_{2k})_{e_q^-} - 2mk)$ **do**
  if $\{i \in N \mid e_q^- \in \arg\min\{\mu_{i,e}^{+k}(b_q)\}$ *and* $\mu_{i,e_q^-}^{+k}(b_q) < \max_{e \in E}\{\mu_{i,e}^{-k}(b_q)\}\} \neq \emptyset$ **then**
    Choose $i \in \{i \in N \mid e_q^- \in \arg\min\{\mu_{i,e}^{+k}(b_q)\}$ and $\mu_{i,e_q^-}^{+k}(b_q) < \max_{e \in E}\{\mu_{i,e}^{-k}(b_q)\}\}$
  else
    Choose $i \in \{i \in N \mid \mu_{i,e_q^-}^{+k}(b_q) < \max_{e \in E}\{\mu_{i,e}^{-k}(b_q)\}\}$
  end
  Choose $e_{q+1}^- \in \arg\max\{\mu_{i,e}^{-k}(b_q)\}$;
  $b_{q+1} \leftarrow (b_q)_{i:e_{q+1}^- \to e_q^-}$;
  $q \leftarrow q + 1$;
end
return (b_q, e_q^-)

Algorithm 3: FP(x_{2k}, f_1, e_1^+): A forward path of restricted best responses.

Input: equilibrium x_{2k} for game $\mathcal{G}_{2k}$, strategy profile f_1 for game $\mathcal{G}_k$ and a resource e_1^+.
Output: Strategy profile f_q for game $\mathcal{G}_k$ and resource e_q^-.

Initialize $q \leftarrow 1$;
while $(\exists i \in N$ *with* $e_q^+ \in \arg\max\{\mu_{i,e}^{-k}(f_q)\}$ *and* $\mu_{i,e_q^+}^{-k}(f_q) < \min_{e \in E}\{\mu_{i,e}^{+k}(f_q)\}) \vee ((f_q)_{e_q^+} \geq (x_{2k})_{e_q^+} + 2mk)$ **do**
  if $\{i \in N \mid e_q^+ \in \arg\max\{\mu_{i,e}^{-k}(b_q)\}$ *and* $\mu_{i,e_q^+}^{-k}(b_q) > \min_{e \in E}\{\mu_{i,e}^{+k}(b_q)\}\} \neq \emptyset$ **then**
    Choose $i \in \{i \in N \mid e_q^+ \in \arg\max\{\mu_{i,e}^{-k}(b_q)\}$ and $\mu_{i,e_q^+}^{-k}(b_q) > \min_{e \in E}\{\mu_{i,e}^{+k}(b_q)\}\}$
  else
    Choose $i \in \{i \in N \mid \mu_{i,e_q^-}^{+k}(b_q) < \max_{e \in E}\{\mu_{i,e}^{-k}(b_q)\}\}$
  end
  Choose $e_{q+1}^+ \in \arg\min\{\mu_{i,e}^{+k}(f_q)\}$;
  $f_{q+1} \leftarrow (f_q)_{i:e_q^+ \to e_{q+1}^+}$;
  $q \leftarrow q + 1$;
end
return (f_q, e_q^-)

with x, we stop this iteration and overwrite x by the resulting strategy profile. Else, we compute a forward path, starting in e_1^+ and overwrite x by the resulting strategy profile. The pseudo-code of subroutine RESTORE can be found in Algorithm 4.

Algorithm 4: Subroutine RESTORE($x_{2k}, \mathcal{G}_k((d'_i)_{i\in N})$).

Input: equilibrium x_{2k} for $\mathcal{G}_{2k}((d_i)_{i\in N})$
Output: equilibrium x_k for $\mathcal{G}_k((d_i)_{i\in N})$

$x \leftarrow x_{2k}$;
while *x not an equilibrium for $\mathcal{G}_k((d_i)_{i\in N})$* **do**
  Choose $j \in \arg\min_{i\in N}\{\mu^{+k}_{i,\min}(x) - \mu^{-k}_{i,\max}(x)\}$;
  Choose $e_1^- \in \arg\max\{\mu^{-k}_{j,e}(x)\}$ and $e_1^+ \in \arg\min\{\mu^{+k}_{j,e}(x)\}$;
  $b_1 \leftarrow x_{j:e_1^- \to e_1^+}$;
  $(b_r, e_r^-) \leftarrow$ BP(x_{2k}, b_1, e_1^-);
  if $e_1^+ \neq e_r^-$ **then**
    $f_1 \leftarrow b_r$;
    $(f_s, e_s^+) \leftarrow$ FP(x_{2k}, f_1, e_1^+);
    $x' \leftarrow f_s$;
  else
    $x' \leftarrow b_r$;
  end
  $x \leftarrow x'$;
end
$x_k \leftarrow x$;
return x_k;

5.3 PACKETHALVER

Using the subroutines ADD and RESTORE, we develop algorithm PACKETHALVER, which computes an equilibrium x_{k_0} for the k_0-splittable game $\mathcal{G}_{k_0}((d_i)_{i\in N})$). In this algorithm, we start with an equilibrium x_k for $\mathcal{G}_k((d'_i)_{i\in N})$), where $d'_i = 0$ for all $i \in N$, $k = 2^{q_1}k_0$ and $q_1 = \arg\min_{q\in\mathbb{N}}\{2^q k_0 > \max_{i\in N} d_i\}$. Note that this game has a trivial equilibrium, where $(x_k)_{i,e} = 0$ for all $i \in N$ and $e \in E$. We repeat the following two steps:

1. Given an equilibrium x_k for $\mathcal{G}_k((d'_i)_{i\in N})$, we construct an equilibrium for $\mathcal{G}_{k/2}((d'_i)_{i\in N})$ using subroutine RESTORE and set k to $k/2$.
2. For each player $i \in N$ we check if $d_i - d'_i \geq k$. If so, we increase d'_i by k and recompute equilibrium x_k using subroutine ADD.

After q_1 iterations PACKETHALVER returns an equilibrium x_{k_0} for $\mathcal{G}_{k_0}((d_i)_{i\in N})$). The pseudo-code of PACKETHALVER can be found in Algorithm 5.

6 Correctness

In this section, we prove that PACKETHALVER indeed returns an equilibrium for game $\mathcal{G}_{k_0}((d_i)_{i\in N})$. In order to do so, we first need to verify that the two subroutines ADD and RESTORE are correct. Harks, Klimm and Peis [23, Thm. 5.1] proved that subroutine ADD indeed returns an equilibrium strategy for the new game with increased demand. It is left to verify correctness of RESTORE and PACKETHALVER.

Algorithm 5: Algorithm PACKETHALVER($\mathcal{G}_{k_0}((d_i)_{i \in N})$)

Input: Integral splittable congestion game $\mathcal{G}_{k_0} = (N, E, (d_i)_{i \in N}, (c_{i,e})_{i \in N, e \in E})$.
Output: An equilibrium x_{k_0} for $\mathcal{G}_{k_0}$.

Initialize $q_1 = \arg\min_{q \in \mathbb{N}}\{2^q k_0 > \max_{i \in N} d_i\}$; $k \leftarrow 2^{q_1} k_0$; $d_i' \leftarrow 0$; $x_k \leftarrow (0)_{e \in E, i \in N}$;
for $1, \ldots, q_1 - 1$ **do**
    $k \leftarrow k/2$;
    $x_k \leftarrow$ RESTORE($x_{2k}, \mathcal{G}_k((d_i')_{i \in N})$);
    for $i \in N$ **do**
        if $d_i - d_i' > k$ **then**
            $x_k \leftarrow$ ADD($x_k, i, \mathcal{G}_k((d_i')_{i \in N})$);
            $d_i' \leftarrow d_i' + k$;
        end
    end
end
return x_k;

6.1 Correctness RESTORE

In RESTORE, the packet size k does not change. Hence, we shorten the notation of the marginal cost $\mu_{i,e}^{+k}(x)$ and $\mu_{i,e}^{-k}(x)$ to $\mu_{i,e}^{+}(x)$ and $\mu_{i,e}^{-}(x)$. To verify that RESTORE is correct, we need to show that the following three properties hold:

1. In the algorithm that constructs the backward path, we need for each $q \in \{1, \ldots, r\}$, that whenever $(b_q)_{e_q^-} \leq (x_{2k})_{e_q^-} - 2mk$, there exists a player that has a restricted improving move to resource e_q^-.
2. In the algorithm that constructs the forward path, we need for each $q \in \{1, \ldots, s\}$, that whenever $(f_q)_{e_q^+} \geq (x_{2k})_{e_q^+} + 2mk$, there exists a player that has a restricted improving move from resource e_q^+.
3. RESTORE terminates.

The first two properties follow directly from Corollary 3. Note that the corollary already implies the existence of a restricted improving move for a load difference of order $\frac{3}{2}mk$, but for the sake of a cleaner analysis of the running time of the algorithm, we work with a load imbalance of order $2mk$. In order to prove that RESTORE terminates, we need to check that we enter the while loop in line 2 only a finite number of times, and, that within the while loop, strategies do not cycle. We define:

$$\Delta_{i,e}(x) := \mu_{i,\min}^{+}(x) - \mu_{i,e}^{-}(x)), \ i \in N, e \in E.$$

Let $\Delta_{\min}(x)$ be the minimum value in $\Delta(x) := (\Delta_{i,e}(x))_{i \in N, e \in E}$. When all elements in $\Delta(x)$ are non-negative, or, equivalently, when $\Delta_{\min}(x)$ is non-negative, x is an equilibrium. In general, under the hypothesis that the backward and forward paths end after a finite number of steps, we obtain the following (finite) sequence of strategy profiles within a while-loop:

$$x \to b_1 \to b_2 \to \cdots \to b_r = f_1 \to f_2 \to \cdots \to f_s = x'. \tag{10}$$

We aim to prove that after each iteration in the while loop of RESTORE, either $\Delta_{\min}(x)$ increased, or the value $\Delta_{\min}(x)$ occurs less often. We define

$$\#(\Delta(x)) := |\{(i,e) \in N \times E \mid \Delta_{i,e}(x) = \Delta_{\min}\}|,$$

as the number of times the value $\Delta_{\min}(x)$ occurs in $\Delta(x)$, and:

$$\Phi(\Delta(x)) := (\Delta_{\min}(x), \#(\Delta(x))).$$

For two strategies, $y, y' \in \prod_{i \in N} \mathcal{S}_i(d_i, k)$, we write that $\Phi(\Delta(y)) <_{\text{lex}} \Phi(\Delta(y'))$ if either: (I) $\Delta_{\min}(y) < \Delta_{\min}(y')$ or (II) $\Delta_{\min}(y) = \Delta_{\min}(y')$ and $\#(\Delta(y)) > \#(\Delta(y'))$.

We aim to prove the following theorem.

Theorem 5 *Let x and x' be defined as in the while-loop (lines 2–15) of* RESTORE, *then* $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\Delta(x'))$.

As the strategy space is finite, Theorem 5 implies that we can only enter the while loop in line 2 of RESTORE a finite number of times. In order to prove Theorem 5, we need to keep track of the single packet exchanges described in Eq. (10). We remark that there is no monotonicity of the form:

$$\Phi(\Delta(x)) \leq_{\text{lex}} \Phi(\Delta(b_1)) \leq_{\text{lex}} \Phi(\Delta(b_2)) \leq_{\text{lex}} \cdots \leq_{\text{lex}} \Phi(\Delta(f_s)) = \Phi(\Delta(x')).$$

For example, a single packet exchange for some player i as described in lines 3-5 of RESTORE can increase $\min_{e \in E}\{(\Delta(x))_{i,e}\}$ but at the same time decrease $\min_{e \in E}\{(\Delta(x))_{j,e}\}$ for another player j, which can cause $\Delta_{\min}(x) > \Delta_{\min}(b_1)$ overall.

Therefore, we introduce slightly different vectors $\tilde{\Delta}(b_1), \ldots, \tilde{\Delta}(f_s)$, where we add a *correcting term* to some entries of $\Delta(b_1), \ldots, \Delta(f_s)$. Let us discuss this corrective term and $\tilde{\Delta}(\cdot)$ in more detail. In strategy b_q, we correct the marginal costs $\mu^-_{i,e}(b_q)$ and $\mu^+_{\min}(b_q)$, as we know that the total load on resource e^-_q will increase by k after the next single packet exchange. On resource e^+_1, the total load will decrease by k once we start with the forward path. We call these corrected marginal costs: $\tilde{\mu}^-_{i,e}(b_q)$ and $\tilde{\mu}^+_{i,e}(b_q)$. Thus, for all $q \in \{1, \ldots r\}$ we define:

$$\tilde{\mu}^-_{i,e}(b_q) := \begin{cases} \mu^-_{i,e}(b_q) + k^2 a_{i,e}, & \text{if } e^-_q \neq e^+_1 \text{ and } e = e^-_q, \\ \mu^-_{i,e}(b_q) - k^2 a_{i,e}, & \text{if } e^-_q \neq e^+_1 \text{ and } e = e^+_1, \\ \mu^-_{i,e}(b_q), & \text{otherwise.} \end{cases} \tag{11}$$

Similarly, we define $\tilde{\mu}^+_{i,e}(b_q)$ and $\tilde{\mu}^+_{i,\min}(b_q) := \min_{e \in E}\{\tilde{\mu}^+_{i,\min}(b_q)\}$ for all $q \in \{1, \ldots r\}$. Then, we define

$$\tilde{\Delta}(b_q) := (\tilde{\mu}^+_{i,\min}(b_q) - \tilde{\mu}^-_{i,e}(b_q))_{i \in N, e \in E}.$$

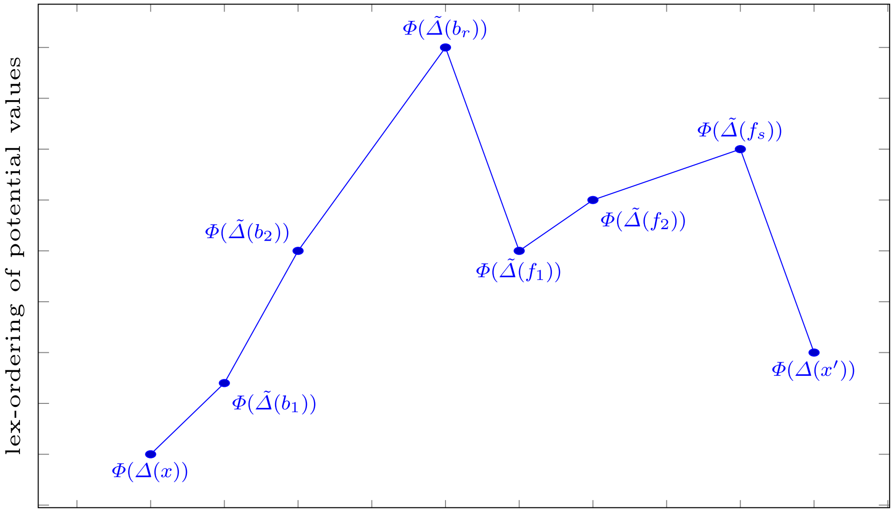

Fig. 1 In this figure, we visualize the lexicographical ordering of potential values $\Phi(\Delta(x)), \Phi(\tilde{\Delta}(b_1)), \ldots, \Phi(\tilde{\Delta}(b_r)), \Phi(\tilde{\Delta}(f_1)), \ldots, \Phi(\tilde{\Delta}(f_s)), \Phi(\Delta(x'))$. This potential vector is lexicographically increasing, except from b_r to f_1 and from f_s to x'. Still, we can show that $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\Delta(x'))$

In strategy f_q, we only correct marginal costs $\mu^-_{i,e}(f_q)$ and $\mu^+_{\min}(f_q)$ for the fact that we know that the total load on resource e^+_q will decrease by k after the next single packet exchange. Thus, for all $q \in \{1, \ldots s\}$ we define:

$$\tilde{\mu}^-_{i,e}(f_q) := \begin{cases} \mu^-_{i,e}(f_q) - k^2 a_{i,e}, & \text{if } e = e^+_q, \\ \mu^-_{i,e}(f_q), & \text{otherwise.} \end{cases} \tag{12}$$

Similarly, we define $\tilde{\mu}^+_{i,e}(f_q)$ and $\tilde{\mu}^+_{i,\min}(f_q) := \min_{e \in E}\{\tilde{\mu}^+_{i,\min}(f_q)\}$ for all $q \in \{1, \ldots s\}$. Then, we define

$$\tilde{\Delta}(f_q) := (\tilde{\mu}^+_{i,\min}(f_q) - \tilde{\mu}^-_{i,e}(f_q))_{i \in N, e \in E}.$$

Under these corrected vectors $\tilde{\Delta}(b_q)$, $\tilde{\Delta}(f_q)$, values only change for the player that executed the restricted best response in that iteration. It can still happen that $\Phi(\tilde{\Delta}(b_r)) \geq_{\text{lex}} \Phi(\tilde{\Delta}(f_1))$ and $\Phi(\tilde{\Delta}(f_s)) \geq_{\text{lex}} \Phi(x')$. Though, besides these two exceptions, the potential value is lexicographically increasing (see Fig. 1 for a visualization of the lexicographical ordering of the potential values). In order to prove Theorem 5, we first prove the following three statements:

i) $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(b_1))$ (Lemma 3).
ii) $\Phi(\tilde{\Delta}(b_q)) \leq_{\text{lex}} \Phi(\tilde{\Delta}(b_{q+1}))$ for all $q \in \{1, \ldots, r-1\}$ (Lemma 4).
iii) $\Phi(\tilde{\Delta}(f_q)) \leq_{\text{lex}} \Phi(\tilde{\Delta}(f_{q+1}))$ for all $q \in \{1, \ldots, s-1\}$ (Lemma 5).

Lemma 3 *Let x and b_1 be defined as in the while-loop (lines 2–15) of* Restore*, then* $\Phi(\Delta(x)) <_{lex} \Phi(\tilde{\Delta}(b_1))$.

Proof Assume we obtain b_1 from strategy x as player j is moving a single packet from resource e_1^- to e_1^+. Using Eq. (11), we know that for all players $i \in N\backslash\{j\}$ and all resources $e \in E$: $\tilde{\Delta}(b_1)_{i,e} = \Delta(x)_{i,e}$. Furthermore, as

- $\mu^+_{j,e_1^+}(x) = \mu^+_{j,e_1^+}(b_1) - 2k^2 a_{j,e_1^+} < \tilde{\mu}^+_{j,e_1^+}(b_1)$,
- $\mu^+_{j,e_1^+}(x) < \mu^-_{j,e_1^-}(x) = \mu^+_{j,e_1^-}(b_1) < \tilde{\mu}^+_{j,e_1^-}(b_1)$ and,
- $\mu^+_{j,e_1^+}(x) \leq \mu^+_{j,e}(x) = \tilde{\mu}^+_{j,e}(b_1)$ for all $e \in E\backslash\{e_1^-, e_1^+\}$,

we obtain:

$$\mu^+_{j,e_1^+}(x) \leq \tilde{\mu}^+_{j,\min}(b_1). \tag{13}$$

This implies

$$\Delta_{\min}(x) = \mu^+_{j,e_1^+}(x) - \mu^-_{j,e_1^-}(x) < \tilde{\mu}^+_{j,\min}(b_1) - \tilde{\mu}^-_{j,e_1^-}(b_1). \tag{14}$$

Hence, at least one value in $\Delta(x)$ that was equal to $\Delta_{\min}(x)$ increased in $\tilde{\Delta}(b_1)$. It is left to show that if there exists a resource e for player j, where $\Delta_{j,e}(x)$ decreased, it does not decrease to a value equal to or lower than $\Delta_{\min}(x)$. Thus, assume there exists an $e \in E\backslash\{e_1^-\}$ such that $\Delta_{j,e}(x) > \tilde{\Delta}_{j,e}(b_1)$. By Eq. (13), we get that $\tilde{\mu}^-_{j,e}(b_1) > \mu^-_{j,e}(x)$, which is only possible if $e = e_1^+$. We obtain:

$$\begin{aligned}
&\tilde{\mu}^+_{j,\min}(b_1) - \tilde{\mu}^-_{j,e_1^+}(b_1) \\
&\geq \mu^+_{j,e_1^+}(x) - \tilde{\mu}^-_{j,e_1^+}(b_1) && \text{(By Eq. (13))} \\
&= (\mu^+_{j,e_1^+}(b_1) - 2k^2 a_{j,e_1^+}) \\
&\quad -(\mu^-_{j,e_1^+}(b_1) - k^2 a_{j,e_1^+}) && \text{(By construction and Eq. (11))} \\
&= \mu^+_{j,e_1^+}(b_1) - \mu^-_{j,e_1^+}(b_1) - k^2 a_{j,e_1^+} && \text{(Simplification)} \\
&= k^2 a_{j,e_1^+} > 0 && \text{(By definition of } \mu^+_{j,e}(\cdot) \text{ and } \mu^-_{j,e}(\cdot))
\end{aligned}$$

Hence, for at least one resource with $\Delta_{j,e}(x) = \Delta_{\min}(x)$ we have $\Delta_{j,e}(x) < \tilde{\Delta}_{j,e}(b_1)$ and for any resource for which $\Delta_{j,e}(x) > \tilde{\Delta}_{j,e}(b_1)$, we have $\tilde{\Delta}_{j,e}(b_1) > \Delta_{\min}(x)$. Hence, with $\Delta_{\min}(x) \leq 0$ this completes the proof. □

Next, we show that $\Phi(\tilde{\Delta}(b_q)) \leq_{\text{lex}} \Phi(\tilde{\Delta}(b_{q+1}))$ for all $q \in \{1, \ldots, r-1\}$ by showing that $\tilde{\Delta}(x)$ is *sorted lexicographically increasing*. Given two vectors $u, v \in \mathbb{R}^n$, we say that v is *sorted lexicographically larger* than u, if there is an $k \in \{1, \ldots, n\}$ such that $u_{\phi(i)} = v_{\psi(i)}$ for all $i < k$ and $u_{\phi(k)} < v_{\psi(k)}$, where ϕ and ψ are permutations that sort u and v non-decreasingly. We write $u <_{\text{slex}} v$. If $u_{\phi(i)} = v_{\psi(i)}$ for all $i \in \{1, \ldots, n\}$, we write $u =_{\text{slex}} v$.

Lemma 4 *Let x and b_q be defined as in the while-loop (lines 2-15) of* Restore *with* $q \in \{1, \ldots, r-1\}$. *Then* $\Phi(\tilde{\Delta}(b_q)) \leq_{lex} \Phi(\tilde{\Delta}(b_{q+1}))$.

Proof Note that it is sufficient to prove that $\tilde{\Delta}(b_q) <_{\text{slex}} \tilde{\Delta}(b_{q+1})$ for all $q \in \{1, \ldots, r-1\}$. In Lemma 3 we knew that the single packet exchange was a restricted best response, but here we only know it is a restricted improving move. Still, similar to the proof of Lemma 3, we know that for all players $j \in N\backslash\{i\}$ and all resources $e \in E$: $\tilde{\Delta}_{j,e}(b_{q+1}) = \tilde{\Delta}_{j,e}(b_q)$.

Note that as

$$\tilde{\mu}^+_{i,e}(b_{q+1}) \geq \tilde{\mu}^+_{i,e}(b_q)$$

for all $e \in E\backslash\{e^+_{q+1}\}$, and

$$\tilde{\mu}^+_{i,e^+_{q+1}}(b_{q+1}) = \tilde{\mu}^-_{i,e^+_{q+1}}(b_q) > \tilde{\mu}^+_{i,e^+_q}(b_q),$$

we obtain that:

$$\tilde{\mu}^+_{i,e^+_{q+1}}(b_q) \leq \tilde{\mu}^+_{i,\min}(b_{q+1}).$$
$$\tilde{\mu}^+_{i,\min}(b_q) \leq \tilde{\mu}^+_{i,\min}(b_{q+1}).$$

Now, for all $e \neq e^+_q, e^+_{q+1}$ we have $\tilde{\mu}^-_{i,e}(b_q) = \tilde{\mu}^-_{i,e}(b_{q+1})$ and thus $\tilde{\Delta}_{i,e}(b_q) \leq \tilde{\Delta}_{i,e}(b_{q+1})$. For e^+_q, we get $\tilde{\mu}^-_{i,e^+_q}(b_q) > \tilde{\mu}^-_{i,e^+_q}(b_{q+1})$ and hence $\tilde{\Delta}_{i,e^+_q}(b_q) < \tilde{\Delta}_{i,e^+_q}(b_{q+1})$. Finally, for e^+_{q+1} we have $\tilde{\mu}^-_{i,e^+_{q+1}}(b_q) < \tilde{\mu}^-_{i,e^+_{q+1}}(b_{q+1})$, but still $\tilde{\Delta}_{i,e^+_{q+1}}(b_{q+1}) > \tilde{\Delta}_{i,e^+_q}(b_q)$. Thus, we get that $\tilde{\Delta}(b_q) <_{\text{slex}} \tilde{\Delta}(b_{q+1})$ and altogether we have $\Phi(\tilde{\Delta}(b_q)) \leq_{\text{lex}} \Phi(\tilde{\Delta}(b_{q+1}))$ for all $q \in \{1, \ldots, r-1\}$. □

As we only use the fact that player i executes a restricted improving move from b_q to b_{q+1}, we obtain a similar statement during the execution of the forward path:

Lemma 5 *Let x and b_q be defined as in the while-loop (lines 2–15) of* Restore *with $q \in \{1, \ldots, s-1\}$. Then $\Phi(\tilde{\Delta}(f_q)) \leq_{lex} \Phi(\tilde{\Delta}(f_{q+1}))$.*

Finally, we can prove Theorem 5.

Proof (Theorem 5) By Lemma 3 we know that $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(b_1))$ and Lemma 4 implies that $\Phi(\tilde{\Delta}(b_q)) \leq_{\text{lex}} \Phi(\tilde{\Delta}(b_{q+1}))$ for all $q \in \{1, \ldots, r-1\}$. Hence, $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(b_r))$. Next we claim that $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(f_1))$. If we have

$$\tilde{\mu}^+_{i,\min}(b_r) - \tilde{\mu}^-_{i,e}(b_r) \leq \tilde{\mu}^+_{i,\min}(f_1) - \tilde{\mu}^-_{i,e}(f_1),$$

for all $i \in N$ and $e \in E$, then $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(f_1))$ and our claim follows. Thus, assume that for some $i \in N$ and $e \in E$ it holds that:

$$\tilde{\mu}^+_{i,\min}(b_r) - \tilde{\mu}^-_{i,e}(b_r) > \tilde{\mu}^+_{i,\min}(f_1) - \tilde{\mu}^-_{i,e}(f_1). \tag{15}$$

This implies that $e_r^- \neq e_1^+$. Using Eqs. (11) and (12) we obtain:

$$\tilde{\mu}_{i,e}^-(f_1) = \begin{cases} \tilde{\mu}_{i,e}^-(b_r) - a_{i,e}k, & \text{if } e = e_r^-, \\ \tilde{\mu}_{i,e}^-(b_r), & \text{otherwise.} \end{cases} \tag{16}$$

And a similar statement as (16) holds for $\tilde{\mu}_{i,e}^+(f_1)$ and $\tilde{\mu}_{i,e}^+(b_r)$. Therefore, we get $e_r^- \in \arg\min_{e \in E} \tilde{\mu}_{i,e}^+(b_r)$ and, hence, $e_r^- \in \arg\min_{e \in E} \mu_{i,e}^+(b_r)$, which in turn implies $e_r^- \in \arg\min_{e \in E} \tilde{\mu}_{i,e}^+(f_1)$. We obtain:

$$\tilde{\mu}_{i,\min}^+(b_r) - \tilde{\mu}_{i,e}^-(b_r) > \tilde{\mu}_{i,e_r^-}^+(f_1) - \tilde{\mu}_{i,e}^-(f_1) \geq \mu_{i,e_r^-}^+(f_1) - \mu_{i,e}^-(b_r) \geq 0.$$

Here, the first inequality holds true as $e_r^- \in \arg\min \mu_{i,e}^+(b_r)$, the second inequality as $e_r^- \neq e_1^+$ and the third as e_r^- is the last resource of the backward path. Thus, for all resources $e \in E$ for which Eq. (15) is true (thus $\tilde{\Delta}_{i,e}(b_r) > \tilde{\Delta}_{i,e}(f_1)$), we have $\tilde{\Delta}_{i,e}(f_1) \geq 0$. As x is not an equilibrium, $\Delta_{\min}(x) < 0$. Thus, as $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(b_r))$, also $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(f_1))$.

Using Corollary 5 we know that $\Phi(\tilde{\Delta}(f_q)) \leq_{\text{lex}} \Phi(\tilde{\Delta}(f_{q+1}))$ for all $q \in \{1, \ldots, s-1\}$ and thus $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(f_s))$.

Lastly, we claim that $\Delta(x) <_{\text{lex}} \tilde{\Delta}(f_s) \leq_{\text{lex}} \Delta(x')$. If for all $i \in N$ and $e \in E$ we have:

$$\tilde{\mu}_{i,\min}^+(f_1) - \tilde{\mu}_{i,e}^-(f_s) \leq \mu_{i,\min}^+(f_s) - \mu_{i,e}^-(f_s),$$

then $\Delta(x) <_{\text{lex}} \Delta(\tilde{f_s}) \leq_{\text{lex}} \Delta(x')$ and the theorem follows. Therefore, assume that for some $i \in N$ and $e \in E$, we have:

$$\tilde{\mu}_{i,\min}^+(f_1) - \tilde{\mu}_{i,e}^-(f_s) > \mu_{i,\min}^+(f_s) - \mu_{i,e}^-(f_s). \tag{17}$$

Equation (12) implies that in this case $e = e_s^+$. As e_s^+ is the end of the backward path, for all $i \in N$, we have either that (I) $e_s^+ \notin \arg\max\{\mu_{i,e}^-(f_s)\}$ or that (II) $\mu_{i,\min}^+(f_s) - \mu_{i,\max}^-(f_s) \geq 0$. In the first case, it holds for any $e' \in \arg\max\{\mu_{i,e}^-(f_s)\}$ that:

$$\begin{aligned} &\mu_{i,\min}^+(f_s) - \mu_{i,e_s^+}^-(f_s) \\ &\quad > \mu_{i,\min}^+(f_s) - \mu_{i,e'}^-(f_s) && (\text{as } e' \in \arg\max\{\mu_{i,e}^-(f_s)\}) \\ &\quad \geq \tilde{\mu}_{i,\min}^+(f_s) - \tilde{\mu}_{i,e'}^-(f_s) && (\text{as } e_s^+ \neq e') \\ &\quad \geq \Delta_{\min}(x). && (\text{as } \Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(f_s))) \end{aligned}$$

In the second case, as $\Delta_{\min}(x) < 0$, we also have $\mu_{i,\min}^+(f_s) - \mu_{i,e_s^+}^-(f_s) \geq 0 > \Delta_{\min}(x)$.

Hence, in both cases, $\mu_{i,\min}^+(f_s) - \mu_{i,e_s^+}^-(f_s) > \Delta_{\min}$. Thus, if $\tilde{\Delta}_{i,e}(f_s) > \Delta_{i,e}(x'))$, then we still have $\Delta_{i,e}(x')) > \Delta_{\min}$. As $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\tilde{\Delta}(f_s))$, we get $\Phi(\Delta(x)) <_{\text{lex}} \Phi(\Delta(x'))$. □

6.2 Correctness PacketHalver

It is left to prove that PacketHalver returns an equilibrium for $\mathcal{G}_{k_0}((d_i)_{i \in N}$.

Theorem 6 *Given a k_0-integral splittable singleton game with affine player-specific cost functions $\mathcal{G}_{k_0} := \big(N, E, (d_i)_{i \in N}, ((c_{i,e})_{e \in E})_{i \in N}\big)$,* PacketHalver *returns an equilibrium for $\mathcal{G}_{k_0}$.*

Proof We initialize $x_{i,e} = 0$ for all $i \in N$ and $e \in E$, which is an equilibrium for the game $\mathcal{G}_{2^{q_1}k_0}((0)_{i \in N})$. Assume that in iteration q we enter the for-loop in PacketHalver with an equilibrium x for game $\mathcal{G}_{2^{q_1-q+1}k_0}$ with demands $d_i' = d_i - (d_i \mod 2^{q_1-q+1}k_0)$. First, Restore computes an equilibrium for demands demands $d_i' = d_i - (d_i \mod 2^{q_1-q}k_0)$ and packet size $2^{q_1-q}k_0$. In lines 5-10 of PacketHalver we check for each player $i \in N$ if her unscheduled load satisfies $d_i - d_i' \geq 2^{q_1-q}k_0$. If so, we schedule one extra packet for player i using subroutine Add. Thus, after the q'th iteration in the for-loop, we obtain an equilibrium for demands $d_i' = d_i - (d_i \mod 2^{q_1-q}k_0)$ and packet size $2^{q_1-q}k_0$. Hence, after the q_1'th iteration, we obtain an equilibrium for the desired packet size $2^0 k_0 = k_0$ and demands $d_i' = d_i - (d_i \mod k_0) = d_i$, which is an equilibrium for game $\mathcal{G}_{k_0}((d_i)_{i \in N})$. □

7 Running time

We prove that the running time of PacketHalver is polynomially bounded in n, m, $\log k$ and $\log \delta$, where δ is the upper bound on player-specific demands d_i. For this, we first need to analyze the running time of the two subroutines Add and Restore.

7.1 Running time Add

In [23, Corollary 5.2] Harks et al. proved that it takes time $nm(\delta/k)^2$ to execute Add. If their algorithm is applied to games with singleton strategy spaces and player-specific affine cost functions, we show next that the running time reduces to $O(nm^4)$. The main reason for this is that equilibria are not very sensitive under small changes in demands.

Lemma 6 *Let x_k be an equilibrium for game $\mathcal{G}_k((d_i)_{i \in N})$ and let x_q be the strategy profile after the q'th iteration of the while-loop described in lines 4–7 of subroutine* Add. *Then $|(x_k)_{i,e} - (x_q)_{i,e}| < 2mk$ for all $i \in N$ and $e \in E$.*

Proof On the contrary, assume q is the first iteration where $|(x_q)_{i,e} - (x_k)_{i,e}| = 2mk$ for some $i \in N$ and $e \in E$. There are two cases: either (I)$(x_q)_{i,e} - (x_k)_{i,e} = 2mk$ or (II)$(x_k)_{i,e} - (x_q)_{i,e} = 2mk$. We prove that the first case leads to a contradiction. For the second case a contradiction can be obtained in a similar manner.

Harks, Klimm and Peis [23] proved that only the players using a resource whose load increased in the previous iteration may have an improving move, and if so, a best response consists in moving one packet from this resource to another one. This implies that $(x_k)_e \leq (x_q)_e \leq (x_k)_e + k$ for all $e \in E$. Thus, when assuming

$(x_q)_{i,e} = (x_k)_{i,e} + 2mk$, we obtain:

$$(x_q)_e + (x_q)_{i,e} \geq (x_k)_e + (x_k)_{i,e} + 2mk. \tag{18}$$

Remember that the total load distributed in x_q by player i exceeds the total load distributed in x_k by at most k, and hence $\sum_{f \in E} (x_q)_{i,e} \leq k + \sum_{f \in E} (x_k)_{i,e}$. We obtain:

$$\sum_{f \neq e} (x_q)_{i,e} \leq \sum_{f \neq e} (x_k)_{i,e} + (1 - 2m)k < \sum_{f \neq e} (x_k)_{i,e} - 2(m - 1)k.$$

The pigeonhole principle implies there exists an $f \in E$ such that $(x_q)_{i,f} < (x_k)_{i,f} - 2k$ and thus $(x_q)_{i,f} \leq (x_k)_{i,f} - 3k$. Combined with the fact that $(x_q)_f \leq (x_k)_f + k$, this implies:

$$(x_q)_{i,f} + (x_q)_f \leq (x_k)_{i,f} + (x_k)_f - 2k. \tag{19}$$

As q is the first iteration in which $(x_q)_{i,e} - (x_k)_{i,e} = 2mk$, we have that $x_q = (x_{q-1})_{i:e' \to e}$ for some $e' \in E$. Using inequalities (18, 19), $m > 1$ and the fact that x_k is an equilibrium for packet size k, we obtain:

$$\mu_{i,e}^{-k}(x_q) > \mu_{i,e}^{+k}(x_k) \geq \mu_{i,f}^{-k}(x_k) \geq \mu_{i,f}^{+k}(x_q).$$

This, combined with the fact that $(x_q)_{i,e} > (x_k)_{i,e} \geq 0$ and that $(x_k)_{i,f} \geq (x_q)_{i,e} + 3k > 0$, implies player i can decrease her cost by moving a packet from e to f. This contradicts the fact that in strategy profile x_{q-1} moving a packet to e is a restricted best response for player i. □

Lemma 7 *Algorithm* ADD *has running time* $O(nm^4)$.

Proof Let x_q be the strategy profile after line 5 of the algorithm has been executed for the q'th time, where we use the convention that x_0 denotes the preliminary strategy profile when entering the while-loop. Note that there is a unique resource e_0 such that $(x_0)_{e_0} = x_{e_0} + k$ and $(x_0)_e = x_e$ for all $e \in E \backslash \{e_0\}$. Furthermore, because we choose in Line 5 a restricted best response, a simple inductive argument shows that after each iteration q of the while-loop, there is a unique resource $e_q \in E$ such that $(x_0)_{e_q} = x_{e_q} + k$ and $(x_0)_e = x_e$ for all $e \in E \backslash \{e_q\}$.

We assume that players move packets according to a *Last In First Out (LIFO)* principle. Thus, whenever player i removes packet i_j from e_q, she moves the packet that was placed on this resource last. We keep track of the marginal cost of a packet i_j at the moment it is moved. Assume that packet i_j is moved in p iterations $q_1, \ldots q_p$. Then:

$$\mu_{i,e_{q_1}}^{-k}(x_{q_1}) > \mu_{i,e_{q_1+1}}^{+k}(x_{q_1}) = \mu_{i,e_{q_1+1}}^{-k}(x_{q_1+1}) = \mu_{i,e_{q_2}}^{-k}(x_{q_2}).$$

Here, the first equality is true as moving packet i_j is an improving move for player i, the second by construction of x_{q_1+1} and the third as $e_{q_2} = e_{q_1+1}$ and by LIFO principle $(x_{q_2})_{i,e_{q_2}} = (x_{q_1+1})_{i,e_{q_2}}$.

Via similar arguments, we obtain: $\mu^{-k}_{i,e_{q_1}}(x_{q_1}) > \mu^{-k}_{i,e_{q_2}}(x_{q_2}) > \cdots > \mu^{-k}_{i,e_{q_p}}(x_{q_p})$. Note that in iterations $q_1, \ldots q_p$, marginal cost value $\mu^{-k}_{i,e_{q_\ell}}(x_{q_\ell})$ does not depend on the aggregated load $(x_{q_\ell})_{e_{q_\ell}}$, as $(x_{q_\ell})_{e_{q_\ell}} = (x_q)_{e_{q_\ell}} + k$ for each $\ell \in \{q_1, \ldots, q_p\}$. Instead it only depends on the player-specific load $(x_{q_\ell})_{i,e_{q_\ell}}$. Lemma 6 implies that each player $i \in N$ will move at most $2m$ packets from each resource and hence there will occur at most $4m$ different values of $(x_{q_\ell})_{i,e_{q_\ell}}$. Thus, each packet visits each resource at most 4 times. As each player i moves at most $2m^2$ packets, and each packet visits each resource (m resources) at most $4m$ times, the running time of Add is bounded by $O(nm^4)$. □

7.2 Running time Restore

We analyze the running time of Restore. The crucial idea is that for each strategy profile y (for a game with packet size k) obtained during the execution of Restore, we have both $|(y_e - (x_{2k})_e| \leq 2mk$ and $|y_{i,e} - (x_{2k})_{i,e}| < 2m^2k$ for all $i \in N$ and $e \in E$. Hence, for each player at most $2m^2$ packets are moved.

Lemma 8 *Let x_{2k} be an equilibrium for game $\mathcal{G}_{2k}$. Then, for any strategy y obtained in any step of algorithm* Restore, *we have that $|y_e - (x_{2k})_e| \leq 2mk$ for all $e \in E$.*

Proof In Restore we iterate between a restricted best response, a backward path of restricted best responses and possibly a forward path of restricted best responses. Within one iteration, we obtain a sequence of strategies as described in Eq. (10)

$$x \to b_1 \to b_2 \to \cdots \to b_r = f_1 \to f_2 \to \cdots \to f_s = x'.$$

We first show that if $|x_e - (x_{2k})_e| < 2mk$, then $|x'_e - (x_{2k})_e| < 2mk$.

If $x'_e = (x_{2k})_e$ for all $e \in E$ the lemma follows trivially. Thus, assume that there exists an e such $x'_e \neq (x_{2k})_e$. By construction of x', we have that $x'_{e_r^-} = x_{e_r^-} - k$, $x'_{e_s^+} = x_{e_s^+} + k$ and $x'_e = (x_{2k})_e$ for all $e \in E \backslash \{e_r^-, e_s^+\}$. Thus, it remains to check that (I) $x'_{e_r^-} > (x_{2k})_{e_r^-} - 2mk$ and (II) $x'_{e_s^+} < (x_{2k})_{e_s^+} + 2mk$. For the first case we note that $x'_{e_r^-} = (b_r)_{e_r^-}$. Any strategy b_r that is returned by Algorithm 2 has the property that $(b_r)_{e_r^-} > (x_{2k})_{e_r^-} - 2mk$ and, hence, $x'_{e_r^-} > (x_{2k})_{e_r^-} - 2mk$. For the second case we note that $x'_{e_r^-} = (f_s)_{e_s^+}$. For any strategy f_s that is returned by Algorithm 3 it holds that $(f_s)_{e_s^+} < (x_{2k})_{e_s^+} + 2mk$. Hence, $x'_{e_s^+} < (x_{2k})_{e_s^+} + 2mk$. Thus, if $|x_e - (x_{2k})_e| < 2mk$ for all $e \in E$, then $|x'_e - (x_{2k})_e| < 2mk$.

In each strategy that is obtained during one iteration in the while loop, there is at most one resource where the total load increase by k and at most one resource where the total load decreased by k. Hence, we obtain that $H(x, b_q) \in \{0, 2k\}$ for all $q \in \{1, \ldots, r\}$ and $H(x, f_q) \in \{0, 2k\}$ for all $q \in \{1, \ldots, s\}$. We combine this observation with the fact that at the beginning of the while-loop of Restore $|x_e - (x_{2k})_e| < 2mk$ and the lemma follows. □

Lemma 9 *Let x_{2k} be an equilibrium for game $\mathcal{G}_{2k}$. Then, for any strategy y obtained in any step of algorithm* Restore*, we have that $|y_{i,e} - (x_{2k})_{i,e}| < 2m^2k$ for all $i \in N$ and $e \in E$.*

Proof By Lemma 8 we know that for any strategy y obtained in any step of algorithm Restore, we have that $|y_e - (x_{2k})_e| \leq 2mk$ for all $e \in E$. Basically, in Restore we find a sequence of single packet exchanges that transforms an equilibrium for packet size $2k$ in an equilibrium for packet size k. Clearly, after zero single packet exchanges, $y = x_{2k}$ and hence $|y_{i,e} - (x_{2k})_{i,e}| < 2m^2k$ for all $i \in N$ and $e \in E$.

Assume that the lemma does not hold and let $y' = y_{e_1 \to e_2}$ be the first single packet exchange such that either

$$\text{(I) } (x_{2k})_{i,e_1} - y'_{i,e_1} = 2m^2k \quad \text{or} \quad \text{(II) } y'_{i,e_2} - (x_{2k})_{i,e_2} = 2m^2k,$$

(or both). Then, by definition of Restore, if (I) holds, we have $e_1 \in \arg\max \mu^-_{i,e}(y)$. Furthermore, Theorem 2 implies that there exist a resource $e_2 \in E$ with

$$\mu^{-k}_{i,e_2}(y') > \mu^{+k}_{i,e_1}(y') = \mu^{-k}_{i,e_1}(y).$$

As $\mu^{-k}_{i,e}(y) \geq \mu^{-k}_{i,e}(y')$ for all $e \in E \backslash \{e_1\}$, we obtain that $\mu^{-k}_{i,e_2}(y) > \mu^{-k}_{i,e_1}(y)$, which contradicts the fact that $e_1 \in \arg\max \mu^-_{i,e}(y)$. Similarly, if (II) holds, $e_2 \in \arg\min \mu^+_{i,e}(y)$. From the proof of Theorem 1 for $r = 2$, it follows that there exist a resource $e_1 \in E$ with

$$\mu^{+k}_{i,e_1}(y') < \mu^{-k}_{i,e_2}(y') = \mu^{+k}_{i,e_2}(y).$$

As $\mu^{+k}_{i,e}(y) \leq \mu^{+k}_{i,e}(y')$ for all $e \in E \backslash \{e_2\}$, this contradicts the fact that $e_2 \in \arg\min \mu^+_{i,e}(y)$. Hence, neither (I) or (II) can be true, and $|y_{i,e} - (x_{2k})_{i,e}| < 2m^2k$ for all $i \in N$ and $e \in E$. □

The bounds on the total and player-specific load enable us to prove that Algorithm 2 runs in polynomial time.

Lemma 10 *Let x_{2k} be an equilibrium for packet size $2k$. And let x be a strategy profile for packet size k such that $|(x_{2k})_e - x_e| < 2mk$ for all $e \in E$. Then Algorithm 2 has a running time of $O(nm^6)$.*

Proof Lemmas 8 and 9 imply that for each player $i \in N$ and each resource $e \in E$, at most $4m^2$ different values $x_{i,e}$ can occur whenever a packet i_j of player i is moved within a path of restricted best responses. Using the same argumentation as in Lemma 7, and assuming packets are moved according to LIFO, the marginal cost of packet i_j decreases each time it is moved. This implies that each unit of demand for player i ($m \cdot 2m^2$ units) visits each resource (m resources) at most $4m^2$ times. Therefore the running time of finding a backward or forward path of restricted best responses is bounded by $O(nm^6)$. □

We combine all previous results to prove Lemma 11.

Lemma 11 RESTORE *has running time* $O(n^2m^{14})$.

Proof The running time of RESTORE is dominated by the number of times we enter the while-loop, and the running time of computing a forward and backward path of restricted best responses, which can both be found in time $O(nm^6)$. Hence, the running time of a complete iteration in the while-loop is $O(nm^6)$.

Note that Lemmas 8 and 9 imply that on each resource e at most $O(m^3)$ different values $\mu_{i,e}^{-k}(\cdot)$ can occur and $O(m^4)$ different values $\mu_{i,\min}^{+k}(\cdot)$. Thus, for each player at most $O(m^7)$ different values $\mu_{i,\min}^{+k}(\cdot)-\mu_{i,e}^{-k}(\cdot)$ can appear on a resource, thus $O(nm^8)$ different values in total. In Lemma 5, we proved that $\Phi(\Delta(x))$ lexicographically increases after each iteration in the while-loop. Hence, we enter the while-loop at most $O(nm^8)$ times. As we enter the while-loop at most $O(nm^8)$ times, and each iteration runs in $O(nm^6)$, PACKETHALVER runs in $O(n^2m^{14})$. □

7.3 Running time PACKETHALVER

Finally, we prove the following theorem.

Theorem 7 PACKETHALVER *runs in time* $O(n^2m^{14}\log(\delta/k_0))$.

Proof Note that we picked $q_1 \in \mathbb{N}$ to be the smallest number such that $2^{q_1}k_0 > d_i$ for all player-specific demands d_i. This implies that q_1 is bounded in $O(\log(\delta/k_0))$, where δ is an upper bound on the player-specific demands. Thus, we execute lines 3-10 $O(\log(\delta/k_0))$ times. In line 4 we call RESTORE, which runs in $O(n^2m^{14})$. In line $5--9$ we execute ADD (which runs in $O(nm^6)$) at most n times. Thus, the computation time of lines $5--10$ is $O(n^2m^6)$. This implies that it takes time $O(n^2m^{14})$ to go through a complete iteration in the for loop. Thus, PACKETHALVER runs in time $O(n^2m^{14}\log(\delta/k_0))$. □

It is left to show that in an atomic splittable game $\mathcal{G}$, $\log(1/k_0)$ is polynomially bounded in the input.

$$
\begin{aligned}
&O\left(\log\left(2m^2(2a_{\max})^{nm}(nm)^{nm/2})/(a_{\text{gcd}}^{nm})\right)\right)\\
&\quad= O\left(\log m + \log\left(\det(A)\right) + \log\left(1/(a_{\text{gcd}}^{nm})\right)\right)\\
&\quad= O\left(\log m + \log((2a_{\max}\sqrt{nm})^{nm}) + \log\left(\textstyle\prod_{i\in N,e\in E_i}\overline{d_i}\cdot\overline{a_{i,e}}\cdot\overline{b_{i,e}}\right)\right)\\
&\quad= O\left(nm\log(nma_{\max}) + \textstyle\sum_{i\in N,e\in E_i}\left(\log(\overline{d_i})+\log(\overline{a_{i,e}})+\log(\overline{b_{i,e}})\right)\right).
\end{aligned}
$$

Which is indeed polynomial in the size of the input. Remember that if we are computing an atomic splittable equilibrium, we first compute the k_0 splittable equilibrium using the algorithm above. Second, we compute the exact equilibrium in time $O((nm)^3)$.

Theorem 8 *Given game* $\mathcal{G}$*, we can compute an atomic splittable equilibrium for* $\mathcal{G}$ *in running time:* $O\left((nm)^3 + n^2m^{14}\log\left(\delta/k_0\right)\right)$.

8 Multimarket cournot oligopoly

In this section, we derive a strong connection between atomic splittable singleton congestion games with affine cost functions and multimarket Cournot oligopolies with affine price functions and quadratic costs. Such a game is compactly represented by the tuple $\mathcal{M} = (N, E, (E_i)_{i\in N}, (p_{i,e})_{i\in N, e\in E_i}, (C_i)_{i\in N})$, where N is a set of n firms and E a set of m markets. Each firm i only has access to a subset $E_i \subseteq E$ of the markets. Each market e is endowed with firm-specific, non-increasing, affine price functions $p_{i,e}(t) = s_{i,e} - r_{i,e}t, i \in N$. In a strategy profile, a firm chooses a non-negative production quantity $x_{i,e} \in \mathbb{R}_{\geq 0}$ for each market $e \in E_i$. We denote a strategy profile for a firm by $x_i = (x_{i,e})_{e\in E_i}$, and a joint strategy profile by $x = (x_i)_{i\in N}$. The production costs of a firm are of the form $C_i(t) = c_i t^2$ for some $c_i \geq 0$. The goal of each firm $i \in N$ is to maximize its utility, which is given by $u_i(x) = \sum_{e\in E_i} p_{i,e}(x_e)x_{i,e} - C_i\Big(\sum_{e\in E_i} x_{i,e}\Big)$, where $x_e := \sum_{i\in N} x_{i,e}$. In the rest of this section we prove that several results that hold for atomic splittable equilibria and k-splittable equilibria carry over to multimarket oligopolies.

A strategic game $\mathcal{G} = (N, (X)_{i\in N}, (u_i)_i \in N)$ is defined by a set of players N, a set of bigfeasible strategies X_i for each player $i \in N$ and a pay-off function $u_i(x)$ for each $i \in N$, where $x \in \times_{i\in N} X_i$.

Definition 2 Let $\mathcal{G} = (N, (X_i)_{i\in N}, (u_i)_i \in N), \mathcal{H} = (N, (Y_i)_{i\in N}, (v_i)_i \in N)$ be two strategic games with identical player set N. Then, $\mathcal{G}$ and $\mathcal{H}$ are called *isomorphic*, if for all $i \in N$ there exists a bijective function $\phi_i : X_i \to Y_i$ and $A_i \in \mathbb{R}$ such that: $u_i(x_1, \ldots x_n) = v_i(\phi_1(x_1), \ldots, \phi_n(x_n)) + A_i$.

Let $\mathcal{G} = (N, (X_i)_{i\in N}, (u_i)_i \in N)$ and $\mathcal{H} = (N, (Y_i)_{i\in N}, (v_i)_i \in N)$ be isomorphic games. Then, $(x_i)_{i\in N}$ is an equilibrium of game $\mathcal{G}$ if and only if $(\phi_i(x_i))_{i\in N}$ is an equilibrium of game $\mathcal{H}$. This implies that $(x_i)_{i\in N}$ is the unique equilibrium of game $\mathcal{G}$ if and only if $(\phi_i(x_i))_{i\in N}$ is the unique equilibrium of game $\mathcal{H}$.

We prove that for each multimarket oligopoly, there exists an isomorphic atomic splittable game. Moreover, we can construct the isomorphism in polynomial time.

Theorem 9 *Given a multimarket oligopoly $\mathcal{M}$, there exists an atomic splittable game $\mathcal{G}$ that is isomorphic to $\mathcal{M}$.*

Proof Given multimarket oligopoly $\mathcal{M}$, we construct an atomic splittable singleton game $\mathcal{G}$. For every firm $i \in N$ we create a player i and we define her demand d_i as an upper bound on the maximal quantity that firm i will produce, that is, $d_i := \sum_{e\in E_i} \max\{t \mid p_{i,e}(t) = 0\}$. Note that if we limit the strategy space for each player $i \in N$ in game $\mathcal{M}$ to strategies x satisfying $\sum_{e\in E_i} x_{i,e} \leq d_i$, all equilibria are preserved. Then, for every player i we introduce a special resource e_i, and define the set of allowable resources for this player as: $\tilde{E}_i = E_i \cup \{e_i\}$ with $e_i \neq e_j$ for $i \neq j$. The cost functions of special resources e_i are defined as $c_{i,e_i}(t) := c_i(t - 2d_i)$ for all $i \in N$ and the cost functions of resources $e \in E_i$ as: $c_{i,e}(t) := -p_{i,e}(t) = r_{i,e}t - s_{i,e}$ for all $i \in N$.

In order to guarantee that the affine cost functions are non-negative, one can add a sufficiently large positive constant $c_{\max}$ to every cost function on each resource. We define: $c_{\max} = \max\left\{\{s_{i,e} \mid \text{ for all } i \in N, e \in E_i\} \cup \{2c_i d_i \mid \text{ for all } i \in N\}\right\}$. Note that adding $c_{\max}$ to every cost function does not change the equilibrium, it only adds $d_i c_{\max}$ to the total cost of each player. The total cost of a strategy x for player i in game $\mathcal{G}$ is: $\pi_i(x') = \sum_{e\in\tilde{E}_i} c_{i,e}(x'_e)x'_{i,e}$, which is equal to

$$\pi_i(x') = \sum_{e\in E_i} -p_{i,e}(x'_e)x'_{i,e} + x'_{i,e_i} c_i(x'_{i,e_i} - 2d_i). \tag{20}$$

As maximizing pay-off equals minimizing costs, the payoff function of player i in x' is defined by: $v_i(x') = -\pi_i(x')$. It is left to prove that game $\mathcal{G}$ is isomorphic to game $\mathcal{M}$. Let x be a feasible strategy in $\mathcal{M}$. For each player $i \in N$, we define bijection $\phi_i : E_i \to \tilde{E}$ as: $\phi_i(x_{i,1}, \ldots, x_{i,m}) = (x_{i,1}, \ldots, x_{i,m}, d_i - \sum_{e\in E_i} x_{i,e}) =: (x'_{i,1}, \ldots, x'_{i,m}, x'_{i,m+1})$. As we limited the strategy space for each $i \in N$ in game $\mathcal{M}$ to strategies x where $\sum_{e\in E_i} x_{i,e} \le d_i$, $x' := \phi(x)$ is a feasible strategy in $\mathcal{G}$. For each feasible strategy x for game $\mathcal{M}$, and for each $i \in N$, we have:

$$\begin{aligned}
u_i(x) &= \sum_{e\in E_i} p_{i,e}(x_e)x_{i,e} - C_i\Big(\sum_{e\in E_i} x_{i,e}\Big)\\
&= \sum_{e\in E_i} p_{i,e}(x_e)x_{i,e} - c_i\Big(d_i - \sum_{e\in E_i} x_{i,e}\Big)\Big(-d_i - \sum_{e\in E_i} x_{i,e}\Big) - c_i d_i^2\\
&= \sum_{e\in E_i} p_{i,e}(x_e)x_{i,e} - c_i\Big(d_i - \sum_{e\in E_i} x_{i,e}\Big)\Big(d_i - \sum_{e\in E_i} x_{i,e} - 2d_i\Big) - c_i d_i^2\\
&= v_i(\phi_1(x_1), \ldots, \phi_1(x_n)) - c_i d_i^2.
\end{aligned}$$

Thus, games $\mathcal{M}$ and $\mathcal{G}$ are isomorphic. □

One of our main results is our polynomial time algorithm that finds the unique equilibrium for atomic splittable singleton congestion games within polynomial time. As for each multimarket oligopoly there exists an atomic splittable game isomorphic to it, we can to construct this unique equilibrium within polynomial time.

Theorem 10 *Given a multimarket oligopoly $\mathcal{M}$, an equilibrium can be computed within running time: $O\left(n^{16}m^{14}\log(\delta/k_0)\right)$.*

Proof This theorem follows directly from the fact that we can construct an atomic splittable singleton game $\mathcal{G}$ isomorphic to $\mathcal{M}$ (using Theorem 9) and the fact that $x = (x_i)_{i\in N}$ is an equilibrium in $\mathcal{G}$ if and only if $x = (\phi_i(x_i))_{i\in N}$ is an equilibrium in $\mathcal{M}$. Note that if in $\mathcal{M}$, firms compete over m markets, the isomorphic atomic splittable singleton game $\mathcal{G}$ has $m + n$ resources. For such a game, Theorem 8 implies that an equilibrium can be found in $O\left(n^3(m+n)^3 + n^2(m+n)^{14}\log(\delta/k_0)\right)$. □

In an *integral multimarket oligopoly* players sell indivisible goods. Thus, players can only produce and sell integer quantities, i.e., $x_{i,e} \in \mathbb{N}_{\ge 0}$ for each $i \in N$ and $e \in E_i$. For these games, we can construct an isomorphic 1-splittable congestion game.

Theorem 11 *Given an integral multimarket oligopoly $\mathcal{M}$, we can construct a 1-splittable congestion game $\mathcal{G}$ isomorphic to $\mathcal{M}$ within running time $O(nm)$.*

Proof We define $d_i := \sum_{e\in E_i} \lfloor \max\{t \mid p_{i,e}(t) = 0\} \rfloor$. Then, the theorem follows using the same construction as in Theorem 9. □

Theorem 12 *Given an integral multimarket oligopoly $\mathcal{M}$, an integral equilibrium can be computed within $O\left(n^{16}m^{14}\log(\delta/k_0)\right)$.*

Proof Theorem 11 implies that we can construct an atomic splittable singleton game $\mathcal{G}$ isomorphic to $\mathcal{M}$. Note that if in $\mathcal{M}$, n firms compete over m markets, the isomorphic atomic splittable singleton game has $m+n$ resources. For such a game, Theorem 7 implies the desired running time. □

Lastly, we extend a result by Todd [44], where the total and individual production in one market in an integer equilibrium and a real equilibrium are compared.

Theorem 13 *Given a multimarket oligopoly $\mathcal{M}$, with real equilibrium $(x_i)_{i\in N}$. Then, for any integer equilibrium $(y_i)_{i\in N}$ it holds that $|x_e - y_e| \leq m+n$ and $|x_{i,e} - y_{i,e}| \leq (m+n)^2$.*

Proof Assume that in game $\mathcal{M}$, n firms compete over m markets. According to Theorem 9, we can construct an atomic splittable congestion game $\mathcal{G}$ on $m+n$ resources that is isomorphic to $\mathcal{M}$ using bijection ϕ. Let $x = (x_i)_{i\in N}$ be an atomic splittable equilibrium of $\mathcal{M}$ and let $y = (y_i)_{i\in N}$ be a 1-splittable equilibrium of $\mathcal{M}$. Then $x' := (\phi_i(x_i))_{i\in N}$ is an atomic splittable equilibrium of $\mathcal{G}$ and $y' := (\phi_i(y_i))_{i\in N}$ is a 1-splittable equilibrium of $\mathcal{G}$. According to Theorem 1 and 2we know that for any real equilibrium x' and 1-splittable equilibrium y' it holds that $|x'_e - y'_e| < (m+n)$ and $|x'_{i,e} - y'_{i,e}| < (m+n)^2$ for all $i \in N$ and $e \in E_i$. Then, using the bijection ϕ described in (20), we get $|x_e - y_e| < (m+n)$ and $|x_{i,e} - y_{i,e}| < (m+n)^2$. □

Todd [44] showed that the total production in a 1-splittable equilibrium is at most $n/2$ away from that in the real equilibrium, and the individual firm's choice can be more than $(n-1)/4$ away from her choice in the real equilibrium. Our bounds are larger than Todd's, yet, they hold for a more general model – multiple markets and firm-specific price functions. We pose as an open question, whether or not our bounds are tight or can be further improved.

Acknowledgements This work is supported by the COST Action CA16228 "European Network for Game Theory". We thank the two reviewers for their valuable feedback.

Funding Open Access funding enabled and organized by Projekt DEAL.

References

1. Abolhassani, M., Bateni, M.H., Hajiaghayi, M., Mahini, H., Sawant, A.: Network Cournot competition. In: Liu, T.Y., Qi, Q., Ye, Y. (eds.) Web and Internet Economics, pp. 15–29. Springer International Publishing, Cham (2014)

2. Ackermann, H., Röglin, H., Vöcking, B.: On the impact of combinatorial structure on congestion games. J. ACM **55**(6), 1–22 (2008)
3. Ackermann, H., Röglin, H., Vöcking, B.: Pure Nash equilibria in player-specific and weighted congestion games. Theor. Comput. Sci. **410**(17), 1552–1563 (2009)
4. Amir, R.: Cournot oligopoly and the theory of supermodular games. Games. Econ. Behav. **15**(2), 132–148 (1996)
5. Bhaskar, U., Fleischer, L., Hoy, D., Huang, C.C.: Equilibria of atomic flow games are not unique. Math. Op. Res. **40**(3), 634–654 (2015)
6. Bhaskar, U., Lolakapuri, P.R.: Equilibrium computation in atomic splittable routing games. In: 26th Annual European Symposium on Algorithms, ESA 2018, August 20-22, 2018, Helsinki, Finland, pp. 58:1–58:14 (2018)
7. Bimpikis, K., Ehsani, S., Ilkiliç, R.: Cournot competition in networked markets. In: Proceedings of the Fifteenth ACM Conference on Economics and Computation, EC '14, pp. 733–733. ACM, NY, (2014). https://doi.org/10.1145/2600057.2602882
8. Bulow, J., Geanakoplos, J., Klemperer, P.: Multimarket oligopoly: strategic substitutes and complements. J. Polit. Econ. **93**(3), 488–511 (1985)
9. Caragiannis, I., Fanelli, A., Gravin, N., Skopalik, A.: Efficient computation of approximate pure Nash equilibria in congestion games. In: FOCS 2011, Palm Springs, CA, USA, 2011, pp. 532–541 (2011)
10. Caragiannis, I., Fanelli, A., Gravin, N., Skopalik, A.: Approximate pure Nash equilibria in weighted congestion games: existence, efficient computation, and structure. ACM Trans. Econ. Comput. **3**(1), 2 (2015)
11. Chen, X., Deng, X., Teng, S.H.: Settling the complexity of computing two-player Nash equilibria. J. ACM **56**(3), 14:1–14:55 (2009)
12. Chien, S., Sinclair, A.: Convergence to approximate Nash equilibria in congestion games. Games Econ. Behav. **71**(2), 315–327 (2011)
13. Cominetti, R., Correa, J.R., Stier-Moses, N.E.: The impact of oligopolistic competition in networks. Op. Res. **57**(6), 1421–1437 (2009). https://doi.org/10.1287/opre.1080.0653
14. Cournot, A.A.: Recherches sur les principes mathématiques de la théorie des richesses Goldsmiths'. L. Hachette, New York Public Library, p. 198 (1838)
15. Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The complexity of computing a Nash equilibrium. SIAM J. Comput. **39**(1), 195–259 (2009)
16. Del Pia, A., Ferris, M., Michini, C.: Totally unimodular congestion games. In: Proceeding 28th Annual ACM-SIAM Symposium on Discrete Algorithms (2017)
17. Deligkas, A., Fearnley, J., Spirakis, P.G.: Lipschitz continuity and approximate equilibria. SAGT, UK (2016)
18. Fabrikant, A., Papadimitriou, C., Talwar, K.: The complexity of pure Nash equilibria. Association for Computing Machinery, New York, NY. https://doi.org/10.1145/1007352.1007445 (2004)
19. Gairing, M., Monien, B., Tiemann, K.: Routing (un-)splittable flow in games with player-specific linear latency functions. ACM Trans. Algorithm. **7**(3), 1–31 (2011)
20. Groenevelt, H.: Two algorithms for maximizing a separable concave function over a polymatroid feasible region. Eur. J. Op. Res. **54**(2), 227–236 (1991)
21. Harks, T.: Stackelberg strategies and collusion in network games with splittable flow. Theory Comput. Syst. **48**, 781–802 (2011)
22. Harks, T., Klimm, M.: Multimarket oligopolies with restricted market access. In: Lavi, R. (ed.) Algorithmic Game Theory, pp. 182–193. Springer, Berlin, Heidelberg (2014)
23. Harks, T., Klimm, M., Peis, B.: Sensitivity analysis for convex separable optimization over integral polymatroids. SIAM J. Optim. **28**(3), 2222–2245 (2018). https://doi.org/10.1137/16M1107450
24. Haurie, A., Marcotte, P.: On the relationship between Nash-Cournot and Wardrop equilibria. Networks **15**, 295–308 (1985)
25. Huang, C.C.: Collusion in atomic splittable routing games. Theory Comput. Syst. **52**(4), 763–801 (2013)
26. Klimm, M., Warode, P.: Parametrized nash equilibria in atomic splittable congestion games via weighted block laplacians. CoRR arXiv:1811.08354 (2018)
27. Korilis, Y., Lazar, A., Orda, A.: Capacity allocation under noncooperative routing. IEEE Trans. on Aut. Contr. **42**(3), 309–325 (1997)
28. Korilis, Y.A., Lazar, A.A., Orda, A.: Architecting noncooperative networks. IEEE J. Sel. Areas Commun. **13**(7), 1241–1251 (1995)

29. Kukushkin, N.: A fixed-point theorem for decreasing mappings. Econ. Lett. **46**, 23–26 (1994)
30. Lipton, R.J., Markakis, E., Mehta, A.: Playing large games using simple strategies. In: Proceedings 4th ACM Conference on Electronic Commerce (EC-2003), San Diego, California, USA, June 9-12, 2003, pp. 36–41 (2003)
31. Marcotte, P.: Algorithms for the network oligopoly problem. J. Op. Res. Soc. **38**(11), 1051–1065 (1987)
32. Meunier, F., Pradeau, T.: A Lemke-like algorithm for the multiclass network equilibrium problem. WINE **2013**, 363–376 (2013)
33. Milgrom, P., Roberts, J.: Rationalizability, learning, and equilibrium in games with strategic complementarities. Econometrica **58**, 1255–1277 (1990)
34. Milgrom, P., Shannon, C.: Monotone comparative statics. Econometrica **62**(1), 157–80 (1994)
35. Moulin, H., Ray, I., Gupta, S.S.: Improving Nash by coarse correlation. J. Econ. Theory **150**, 852–865 (2014)
36. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. Wiley, New York (1988)
37. Novshek, W.: On the existence of Cournot equilibrium. Rev. Econ. Stud. **52**(1), 85–98 (1985)
38. Orda, A., Rom, R., Shimkin, N.: Competitive routing in multi-user communication networks. IEEE/ACM Trans. Netw **1**, 510–521 (1993)
39. Papadimitriou, C.H.: On the complexity of the parity argument and other inefficient proofs of existence. J. Comput. Syst. Sci. **48**(3), 498–532 (1994)
40. Richman, O., Shimkin, N.: Topological uniqueness of the Nash equilibrium for selfish routing with atomic users. Math. Oper. Res. **32**(1), 215–232 (2007)
41. Rosenthal, R.: A class of games possessing pure-strategy Nash equilibria. Internat. J. Game Theory **2**(1), 65–67 (1973)
42. Roughgarden, T., Schoppmann, F.: Local smoothness and the price of anarchy in splittable congestion games. J. Econom. Theory **156**, 317 – 342 (2015). https://doi.org/10.1016/j.jet.2014.04.005. http://www.sciencedirect.com/science/article/pii/S0022053114000593. Computer Science and Economic Theory
43. Skopalik, A., Vöcking, B.: Inapproximability of pure Nash equilibria. In: Proceeding 40th Annual ACM Symposium. Theory Comput., pp. 355–364 (2008)
44. Todd, M.J.: Computation, multiplicity, and comparative statics of Cournot equilibria in integers. Math. Op. Res. **41**(3), 1125–1134 (2016). https://doi.org/10.1287/moor.2015.0771
45. Topkis, D.: Equilibrium points in nonzero n-person submodular games. SIAM J. Control Optim. **17**, 773–787 (1979)
46. Topkis, D.: Supermodularity and Complementarity. Princeton University Press, Princeton (1998)
47. Tran-Thanh, L., Polukarov, M., Chapman, A., Rogers, A., Jennings, N.: On the existence of pure strategy Nash equilibria in integer-splittable weighted congestion games. In: SAGT (2011)
48. Vives, X.: Nash equilibrium with strategic complementarities. J. Math. Econ. **19**(3), 305–321 (1990)
49. Vives, X.: Games with strategic complementarities: New applications to industrial organization. Int. J. Ind. Organ. **23**(7–8), 625–637 (2005)