



Innovative Applications of O.R.

Computing balanced solutions for large international kidney exchange schemes when cycle length is unbounded ^{*}



Márton Benedek ^{a,b}, Péter Biró ^{a,b,*}, Gergely Csáji ^{b,c}, Matthew Johnson ^d,
Daniël Paulusma ^d, Xin Ye ^d

^a Corvinus University of Budapest, Budapest, Hungary

^b Institute of Economics, ELTE KRTK, Budapest, Hungary

^c Eötvös Loránd University, Budapest, Hungary

^d Department of Computer Science, Durham University, Durham, UK

ARTICLE INFO

Keywords:

Computational complexity
Cooperative game theory
Partitioned permutation game
International kidney exchange

ABSTRACT

In kidney exchange programmes, patients with incompatible donors obtain kidneys via cycles of transplants. Countries may merge their national patient-donor pools to form international programmes. To ensure fairness, a credit-based system is used: a cooperative game-theoretic solution concept prescribes a “fair” initial allocation, which is adjusted with accumulated credits to form a target allocation. The objective is to maximize the number of transplants while staying close to the target allocation. When only 2-cycles are permitted, a solution that lexicographically minimizes deviations from the target can be found in polynomial time. However, even the problem of maximizing the number of transplants is NP-hard for larger upper bounds on cycle length. This latter problem is tractable when cycle lengths are not bounded. We formalize this setting via a new class of cooperative games called *partitioned permutation games*, and prove that computing an optimal solution that is lexicographically closest to the target allocation is NP-hard. We give a randomized XP-time algorithm for solve this problem exactly. We present an experimental study, simulating programmes with up to 10 countries. Allowing unbounded cycle lengths increases the number of transplants by up to 46% compared to 2-cycles. Using credits and selecting lexicographically closest solutions yields low total relative deviation (below 2% for all fairness notions). Among the seven fairness notions tested, a modified Banzhaf value performs best in balancing fairness and efficiency, achieving average deviations below 0.65%. Lexicographic minimization from the target allocation leads to significantly (36 – 56%) smaller average deviations than minimizing maximum difference only.

1. Introduction

We introduce a new class of cooperative games called *partitioned permutation games*, which are closely related to the known classes of permutation games (Tijs et al., 1984) and partitioned matching games (Benedek et al., 2025). Partitioned matching games and partitioned permutation games have immediate applications in international kidney exchange. Before defining these games, we first explain this application area.

Kidney Exchange. The most effective treatment for kidney failure is transplanting a kidney from a deceased or living donor, with better long-term outcomes in the latter case. However, a kidney from a family member or friend might be medically incompatible and could easily be rejected by the patient’s body. Therefore, many countries run national

Kidney Exchange Programmes (KEPs); see Biró et al. (2019, 2021) and the recent survey (Barkel et al., 2025). In a KEP, all patient-donor pairs are placed together in one pool. If for two patient-donor pairs (p, d) and (p', d') , it holds that d and p are incompatible as well as d' with p' , but d and p' are compatible as well as d' with p , then d could donate a kidney to p' , and d' could donate a kidney to p . This is a *2-way exchange*.

We can generalize a 2-way exchange as follows. We model a pool of patient-donor pairs as a directed graph $G = (V, A)$ (the *compatibility graph*) in which each vertex of V is a patient-donor pair, and A consists of every arc (u, v) such that the donor of u is compatible with the patient of v . In a directed cycle $C = u_1 u_2 \dots u_k u_1$, for some $k \geq 2$, the kidney of the donor of u_i could be given to the patient of u_{i+1} for every $i \in \{1, \dots, k\}$, with $u_{k+1} := u_1$. This is a *k-way exchange* using the *exchange cycle C*.

^{*} A 3-page extended abstract of this paper appeared in the proceedings of AAMAS 2024 (Benedek et al., 2024a).

^{*} Corresponding author.

E-mail addresses: benedek.marton@rtk.hun-ren.hu (M. Benedek), biro.peter@rtk.mta.hu (P. Biró), csaji.gergely@gmail.com (G. Csáji), matthew.johnson2@durham.ac.uk (M. Johnson), daniel.paulusma@durham.ac.uk (D. Paulusma), xin.ye@durham.ac.uk (X. Ye).

<https://doi.org/10.1016/j.ejor.2025.12.046>

Received 12 August 2024; Accepted 31 December 2025

Available online 9 January 2026

0377-2217/© 2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

To prevent exchange cycles from breaking (and a patient from losing their willing donor), hospitals perform the k transplants in a k -way exchange simultaneously. Hence, KEPs impose a bound ℓ , called the *exchange bound*, on the maximum length (number of arcs) of an exchange cycle, typically $2 \leq \ell \leq 5$. An ℓ -cycle packing of G is a set C of directed cycles, each of length at most ℓ , that are pairwise vertex-disjoint; if $\ell = \infty$, we also say that C is a *cycle packing*. The size of C is the sum of the lengths of the cycles of C .

KEPs operate in rounds. A solution for round r is an ℓ -cycle packing in the corresponding compatibility graph G^r . The goal is to help as many patients as possible in each round. Hence, to maximize the number of transplants in round r , we seek an *optimal* solution, that is, a *maximum* ℓ -cycle packing of G^r , i.e., one that has maximum size. After round r , some patients will have received a kidney or died (or have left the pool for some other reason), while other patient-donor pairs may have arrived. This results in a new compatibility graph G^{r+1} for round $r + 1$. We have now finished the description of the basic KEP model, as used in this paper. As discussed by Biró et al. (2019), there exist many variants and extensions, some of which we will discuss in more detail later.

An important computational issue for KEPs is how to find an optimal solution in each round. If $\ell = 2$, we can apply a well-known transformation (see e.g. Roth et al., 2005) that only keeps the “double” arcs of a compatibility graph G . Namely we transform G into an undirected graph $\hat{G} = (V, E)$ as follows. For every $u, v \in V$, we have $uv \in E$ if and only if $(u, v) \in A$ and $(v, u) \in A$. It then remains to compute a maximum matching in \hat{G} , which can be done in polynomial time (Edmonds, 1965). If we set $\ell = \infty$, another well-known trick works (see e.g. Abraham et al., 2007). We transform G into a bipartite graph H with partition classes V and V' , where V' is a copy of V . For each $u \in V$ and its copy $u' \in V'$, we add the edge uu' with weight 0. For each $(u, v) \in A$, we add the edge uv with weight 1. Now it remains to find in polynomial time a maximum weight perfect matching in H . However, for any constant $\ell \geq 3$, the complexity changes, as shown by Abraham, Blum and Sandholm (Abraham et al., 2007).

Theorem 1 (Abraham et al., 2007). *If $\ell = 2$ or $\ell = \infty$, we can find an optimal solution for a KEP round in polynomial time; else this is NP-hard.*

International Kidney Exchange. As merging pools of national KEPs leads to better outcomes, many countries today work together with the aim of forming an *international* KEP (IKEP), e.g. Austria and the Czech Republic (Böhmič et al., 2017); Denmark, Norway and Sweden; and Italy, Portugal and Spain (Valentín et al., 2019). Apart from ethical, legal and logistical issues (all beyond our scope), there is now a new and highly non-trivial issue that needs to be addressed: *How can we ensure long-term stability of an IKEP?* If countries are not treated *fairly*, they may leave the IKEP. In a worst-case scenario it could even happen that in the end each country returns to their own national KEP again.

Example 1. Let G be the compatibility graph from Fig. 1. Then a total of five kidney transplants is possible (only) if we choose the solution of size 5 that consists of the single exchange cycle $C = abdeca$. We have that patient-donor pairs a, b, c belong to country 1, whereas patient-donor pairs d and e belong to country 2, and patient-donor pair f belongs to country 3. So using this solution, countries 1, 2 and 3 receive three, two and zero kidney transplants, respectively. Note that without cooperation only country 2 can receive kidney transplants in this example.

Cooperative Game Theory considers *fair* distributions of joint profit if all parties involved collaborate (the notion of fairness depends on context). Before describing its role in our setting, we first give some basic terminology on cooperative game theory; we refer to Peters (2008) for more details.

A (*cooperative*) *game* is a pair (N, v) , where N is a set of n players and $v : 2^N \rightarrow \mathbb{R}$ is a *value function* with $v(\emptyset) = 0$. A subset $S \subseteq N$ is a *coalition*. If for every possible partition (S_1, \dots, S_r) of N it holds that $v(N) \geq v(S_1) + \dots + v(S_r)$, then players will benefit most by forming the *grand coalition* N . The problem is then how to fairly distribute $v(N)$

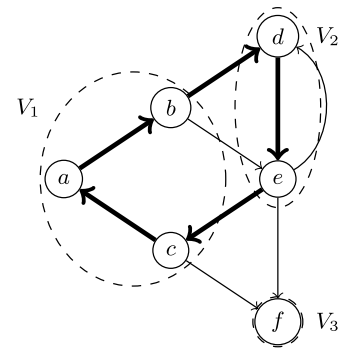


Fig. 1. A partitioned permutation game (N, v) of width $c = 3$ defined on a graph $G = (V, A)$. Note that $N = \{1, 2, 3\}$ and $V = V_1 \cup V_2 \cup V_3$ with $V_1 = \{a, b, c\}$, $V_2 = \{d, e\}$ and $V_3 = \{f\}$, as indicated by the dotted circles. Recall that G has exactly one maximum cycle packing C , which consists of the single exchange cycle $C = abdeca$, as indicated by the thick edges. Selecting C yields the allocation $x = (3, 2, 0)$. We note that $v(N) = v(\{1, 2\}) = 5$, while $v(\{2\}) = 2$ and $v(\{1\}) = v(\{3\}) = v(\{1, 3\}) = v(\emptyset) = 0$ and $v(\{2, 3\}) = 2$. It can be readily checked that for all $S \subseteq N$, $x(S) \geq v(S)$. Hence, x belongs to the core of (N, v) .

amongst the players of N . An *allocation* is a vector $x \in \mathbb{R}^N$ with $x(N) = v(N)$ (we write $x(S) = \sum_{p \in S} x_p$ for $S \subseteq N$). In this context, the notion of fairness is defined by a *solution concept*, which prescribes a set of fair allocations for a game (N, v) . Each solution concept has its own fairness properties; we discuss these properties in detail in Section 4.2.

The solution concepts considered in this paper are the Shapley value, Banzhaf value (two variants) nucleolus, tau value, benefit value, contribution value (all of which are defined in Section 4.2) and the core. They all prescribe a unique allocation except for the *core*, which consists of all allocations $x \in \mathbb{R}^N$ with $x(S) \geq v(S)$ for every $S \subseteq N$. Core allocations ensure N is stable in the sense that no subset S will benefit from forming their own coalition. However, the core of a cooperative game may be empty.

We now define some relevant games. For a directed graph $G = (V, A)$ and subset $S \subseteq V$, we let $G[S] = (S, \{(u, v) \in A \mid u, v \in S\})$ be the subgraph of G induced by S . An ℓ -permutation game on a directed graph $G = (V, A)$ is the game (N, v) , where $N = V$ and for $S \subseteq N$, the value $v(S)$ is the maximum size of an ℓ -cycle packing of $G[S]$. Two special cases are well studied. We obtain a *matching game* if $\ell = 2$, which may have an empty core (e.g. when G is a triangle on vertices a, b, c with arcs (a, b) , (b, a) , (a, c) , (c, a) , (b, c) , (c, b) , see also Fig. 2) and a *permutation game* if $\ell = \infty$, whose core is always nonempty (Tijs et al., 1984).

In the remainder, we differentiate between the sets N (of players in the game) and V (of vertices in the underlying graph G). That is, we associate each player $i \in N$ with a distinct subset V_i of V . A *partitioned* ℓ -permutation game on a directed graph $G = (V, A)$ with a partition (V_1, \dots, V_n) of V is the game (N, v) , where $N = \{1, \dots, n\}$, and for $S \subseteq N$, the value $v(S)$ is the maximum size of an ℓ -cycle packing of $G[\bigcup_{i \in S} V_i]$. We obtain a *partitioned matching game* (Benedek et al., 2023, 2025) if $\ell = 2$, and a *partitioned permutation game* if $\ell = \infty$. The *width* of (N, v) is the *width* of (V_1, \dots, V_n) , which is defined as $c = \max\{|V_i| \mid 1 \leq i \leq n\}$. We refer again to Fig. 1 for an example.

The Model. For a round of an IKEP with exchange bound ℓ , let (N, v) be the partitioned ℓ -permutation game defined on the compatibility graph $G = (V, A)$, where $N = \{1, \dots, n\}$ is the set of countries in the IKEP, and V is partitioned into subsets V_1, \dots, V_n such that for every $p \in N$, V_p consists of the patient-donor pairs of country p . We say that (N, v) is the *associated game* for G . We can now make use of a solution concept S for (N, v) to obtain a fair *initial allocation* y , where y_p prescribes the initial number of kidney transplants country p should receive in this round (possibly, y_p is not an integer, but as we shall see this is not relevant).

To ensure IKEP stability, we use the model of Klimentova et al. (2021), which is a *credit-based* system. For round $r \geq 1$, let G^r be the

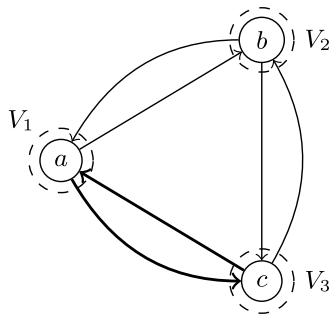


Fig. 2. A partitioned matching game (N, v) of width $c = 1$ on the graph $G = (V, A)$ with $N = \{1, 2, 3\}$ and $V = V_1 \cup V_2 \cup V_3$ for $V_1 = \{a\}$, $V_2 = \{b\}$ and $V_3 = \{c\}$. Note that $v(\{1\}) = v(\{2\}) = v(\{3\}) = v(\emptyset) = 0$, whereas $v(N) = v(\{1, 2, 3\}) = v(\{1, 2\}) = v(\{1, 3\}) = v(\{2, 3\}) = 2$, implying the core of (N, v) is empty.

compatibility graph with associated game (N, v^r) ; let y^r be the initial allocation (as prescribed by some solution concept S); and let $c^r : N \rightarrow \mathbb{R}$ be a *credit function*, which satisfies $\sum_{p \in N} c_p^r = 0$; if $r = 1$, we set $c^r \equiv 0$. For $p \in N$, we set $x_p^r := y_p^r + c_p^r$ to obtain the *target allocation* x^r for round r ; note that x^r is indeed an allocation, as y^r is an allocation and $\sum_{p \in N} c_p^r = 0$. We choose some maximum ℓ -cycle packing C of G^r as optimal solution for round r (out of possibly exponentially many optimal solutions). For round r , let $s_p(C)^r$ be the number of kidney transplants for patients in country p (with donors both from p and other countries). For $p \in N$, we set $c_p^{r+1} := x_p^r - s_p(C)^r$ to get the credit function c^{r+1} for round $r + 1$ (note that $\sum_{p \in N} c_p^{r+1} = 0$). For round $r + 1$, a new initial allocation y^{r+1} is prescribed by S for the associated game (N, v^{r+1}) . For every $p \in N$, we set $x_p^{r+1} := y_p^{r+1} + c_p^{r+1}$, and we repeat the process.

Apart from specifying the solution concept S , we must also determine how to choose in each round a maximum ℓ -cycle packing C (optimal solution) of the corresponding compatibility graph G . We will choose C , such that the vector $s(C)$, with number of kidney transplants entries $s_p(C)$, is *closest* to the target allocation x for the round under consideration. To ensure (*long-term*) *stability* of an IKEP, we aim to make an IKEP *balanced*, that is, in each round the goal is to find optimal cycle packings that are close to the target allocations, yielding consistently low (ideally zero) credit values.

To explain our distance measures, let $|x_p - s_p(C)|$ be the *deviation* of country p from its target x_p if C is chosen out of all optimal solutions. We order the deviations $|x_p - s_p(C)|$ non-increasingly as a vector

$$d(C) = (|x_{p_1} - s_{p_1}(C)|, \dots, |x_{p_n} - s_{p_n}(C)|).$$

We say that C is *strongly close* to x if $d(C)$ is lexicographically minimal over all optimal solutions. If we only minimize $d_1(C) = \max_{p \in N} \{|x_p - s_p(C)|\}$ over all optimal solutions, we obtain a *weakly close* optimal solution. If an optimal solution is strongly close, then it is weakly close, but the reverse might not be true. Both measures have been used in the literature, as we discuss below after first giving an example.

Example 2. Let $\ell = 2$ and consider the partitioned matching game (N, v) from Fig. 2, which has width $c = 1$. Note that $N = \{1, 2, 3\}$ and $V = V_1 \cup V_2 \cup V_3$ with $V_1 = \{a\}$, $V_2 = \{b\}$ and $V_3 = \{c\}$. Moreover, $v(N) = v(\{1, 2, 3\}) = v(\{1, 2\}) = v(\{1, 3\}) = v(\{2, 3\}) = 2$, while $v(\{1\}) = v(\{2\}) = v(\{3\}) = v(\emptyset) = 0$. Say we are in round 1 and use the Shapley value (see Section 4), which yields the initial allocation $x^1 = (\frac{2}{3}, \frac{2}{3}, \frac{2}{3})$. As $c^1 = (0, 0, 0)$, we have $y^1 = x^1$ as target allocation. Note that y^1 is not in the core, because the core of (N, v) is empty, as we observed before. There are three optimal solutions, which each correspond to a cycle packing consisting of exactly one 2-vertex cycle. By symmetry, each of them is strongly close to x . Suppose we select $C = \{\{aba\}\}$. Note that $s_1(C) = s_2(C) = 1$ and $s_3(C) = 0$. Hence, $d(C) = (\frac{1}{3}, \frac{1}{3}, \frac{2}{3})$, and the credit vector c^2 for round 2 is $(-\frac{1}{3}, -\frac{1}{3}, \frac{2}{3})$.

Related work

Benedek et al. (2025) proved the following theorem for $\ell = 2$ (the “matching” case).

Theorem 2 (Benedek et al., 2025). *For partitioned matching games, the problem of finding an optimal solution that is strongly close to a given target allocation x is polynomial-time solvable.*

In contrast to Theorem 2, Benedek et al. (2025) also showed that the problem becomes NP-hard for *edge-weighted* graphs, in which the edges $e \in A$ have an associated weight $w(e)$ reflecting the expected utility of the corresponding kidney transplant. Namely, if maximum weight matchings are taken as optimal solutions for edge-weighted graphs, then it is NP-hard to find a weakly close optimal solution even for $n = 2$.

Benedek et al. (2022) used the algorithm of Theorem 2 to perform simulations for up to 15 countries for $\ell = 2$. As initial allocations, they used the Shapley value, nucleolus, benefit value and contribution value, with the Shapley value yielding the best results. Afterwards, Benedek et al. (2024b) extended the experimental results from Benedek et al. (2022) for $\ell = 2$ by also considering two variants of the Banzhaf value and the tau value. It turned out that the Banzhaf value variants behaved slightly better than the Shapley value.

The good performance of the Shapley value in Benedek et al. (2022, 2024b) is in line with the simulation results of Klimentova et al. (2021) and Biró et al. (2020) for $\ell = 3$. Due to Theorem 1, the simulations in Biró et al. (2020), Klimentova et al. (2021) are for up to four countries and use weakly close optimal solutions. We note that the simulations in Biró et al. (2020), Klimentova et al. (2021) model IKEPs that allow for *non-directed* donors. These are altruistic donors that are added to the pool without a corresponding patient. By allowing non-directed donors, optimal solutions for compatibility graphs may include *exchange chains*, which are directed paths that start from a non-directed donor. This may result in more kidney transplants.

For $\ell = 2$, we refer to Sun et al. (2021) for an alternative model based on so-called selection ratios using lower and upper target numbers. IKEPs have also been modelled as non-cooperative games in the *consecutive matching* setting. In this setting, each round consists of two phases: national pools in phase 1 and a merged pool for unmatched patient-donor pairs in phase 2; see Carvalho and Lodi (2023), Carvalho et al. (2017), Smeulders et al. (2024) for some results in this setting.

Finally, fairness (versus optimality) issues are also studied for national KEPs, in particular in the US. The US setting is different from the European setting, which is the setting we consider, for the following reasons. As we explained above, in Europe the treatment of patient-donor pairs is regulated centrally implying that all patient-donor pairs are registered at an (I)KEP, leading to large and diverse patient-donor pair pools with many exchange options and matching runs that are conducted typically every three months. In contrast, in the US the transplant centers (hospitals) are independent. They mostly treat incoming patient-donor pairs in-house, and need to be convinced to register patient-donor pairs to one of the three nationwide KEPs (UNOS, APD, and NKR) which compete against each other (see Agarwal et al., 2019). Therefore, in the US, kidney transplants usually take place immediately after the registration of a new pair. How to incentivize KEPs in the US setting to register all their patient-donors pairs (instead of only the hard-to-match pairs) to a joint pool is a challenging problem that has been widely studied; see e.g. Ashlagi et al. (2015), Ashlagi and Roth (2012, 2014), Blum et al. (2017), Hajaj et al. (2015), Toulis and Parkes (2015) for a number of different approaches, which were compared to each other in the recent survey (Barkel et al., 2025). In particular, Hajaj et al. (2015) suggested a credit-based approach, namely to give the nationwide KEPs positive credits for registering easy-to-match pairs and negative credits for registering hard-to-match pairs. Their approach was further analysed by Agarwal et al. (2019) and has been adapted by the NKR but is very different in nature from our credit-based approach.

Our Results

We consider partitioned permutation games and IKEPs, so we assume $\ell = \infty$. The latter assumption is not realistic in kidney exchange. However, the assumption has also been made for national KEPs in order to obtain general results of theoretical interest (Aziz et al., 2021; Biró et al., 2024; Biró et al., 2009; Roth et al., 2004). Moreover, results of studies with $\ell = \infty$ may have wider applications, for instance, in portfolio compression in financial clearing (D’Errico & Roukny, 2021), time exchange (Andersson et al., 2021), and shift-reallocation (Manjunath & Westkamp, 2021). In particular, we wanted to research how the stability of an IKEP and the total number of kidney transplants are affected when moving from one extreme ($\ell = 2$) to the other ($\ell = \infty$).

As mentioned, the KEP model we use is the most basic version, which has been used in the literature for conducting computer simulations on generated instance. In particular, this allows us to make a clean comparison with a previous experimental study for $\ell = 2$ (Benedek et al., 2024b). That is, we ignore the diverse sets of complex, hierarchical optimization criteria that are used in European kidney exchange (Biró et al., 2019, 2021) and only maximize the number of transplants in each matching run. We do note that this objective is always the first objective in every used hierarchy apart from the hierarchy used in the UK, where it is the second objective. We also ignore the presence of non-directed donors and the possibility of having multiple donors registering for one patient. To further justify our choice of not considering non-directed donors, we note that such donors are illegal in some European countries, such as France and Hungary.

Our paper consists of a theoretical part and an experimental part. We start with our theoretical results (Section 2). Permutation games, i.e. partitioned permutation games of width 1, have a nonempty core (Tijs et al., 1984), and a core allocation can be found in polynomial time (Curiel & Tijs, 1986). We generalize these two results to partitioned permutation games of any width c , and also show a complexity dichotomy for testing core membership, which is in contrast with the complexity dichotomy for partitioned matching games, where the complexity jump happens at $c = 3$ (Benedek et al., 2025).

Theorem 3. *The core of every partitioned permutation game is nonempty, and it is possible to find a core allocation in polynomial time. Moreover, for partitioned permutation games of fixed width c , the problem of deciding if an allocation is in the core is polynomial-time solvable if $c = 1$ and coNP-complete if $c \geq 2$.*

Due to Theorem 1, we cannot hope to generalize Theorem 2 to hold for any constant $\ell \geq 3$. Nevertheless, Theorem 1 leaves open the question of whether there is an analogue of Theorem 2 for $\ell = \infty$, the “cycle packing” case; Theorem 2 is concerned only with $\ell = 2$, the “matching” case. We show that the answer to this question is no (assuming $P \neq NP$).

Theorem 4. *For partitioned permutation games even of width 2, the problem of finding an optimal solution that is weakly or strongly close to a given target allocation x is NP-hard.*

We refer to Table 1 for an overview of all the known and new results that we discussed above.

Our last theoretical result is a randomized XP algorithm with parameter n . As we shall prove, derandomizing it requires solving the notorious EXACT PERFECT MATCHING problem in polynomial time. The complexity status of the latter problem is still open since its introduction by Papadimitriou and Yannakakis (1982) in 1982.

Theorem 5. *For a partitioned permutation game (N, v) on a directed graph $G = (V, A)$, the problem of finding an optimal solution that is weakly or strongly close to a given target allocation x can be solved by a randomized algorithm in $|A|^{O(n)}$ time.*

Our theoretical results highlighted severe computational limitations, and we now turn to our simulations. We perform a large-scale experimental study, in which our simulations are strongly guided by our theoretical results. Namely, we note that the algorithm in Theorem 5 is

not practical for instances of realistic size (which we aim for). Moreover, it is a randomized algorithm. Hence, it is not acceptable either in the setting of kidney exchange, as policy makers would only use solution yielding the maximum number of kidney transplants. Therefore and also due to Theorem 4, we formulate the problems of computing a weakly or strongly close optimal solution as integer linear programs, as described in Section 3. This enables us to use an ILP solver. We still exploit the fact that for $\ell = \infty$ (Theorem 1) we can find optimal solutions and values $v(S)$ in polynomial time. In this way we can still perform, in Section 4, simulations for IKEPs up to ten countries, so more than the four countries in the simulations for $\ell = 3$ (Biró et al., 2020; Klimenova et al., 2021), but less than the 15 countries in the simulations for $\ell = 2$ (Benedek et al., 2024b).

For the initial allocations we use two easy-to-compute solution concepts: the benefit value and contribution value and four hard-to-compute solution concepts: the Banzhaf value, Shapley value, nucleolus and tau-value. In particular, it follows from a well-known alternative definition of the nucleolus (Maschler et al., 1979) that the nucleolus is a core allocation for every game with a nonempty core. As by Theorem 3, every partitioned permutation game has a nonempty core, this means that we do not need to consider the core as a separate solution concept in our simulations. Finally, we use the modified Banzhaf value called Banzhaf* value from Benedek et al. (2024b), where the credits are incorporated directly into the coalitional values. As in Benedek et al. (2024b), we research both the case where all countries have the same size and the case where countries can be either of small, medium or large size.

Our simulations show, as in Benedek et al. (2024b), that a credit system using strongly close optimal solutions instead of weakly optimal solutions makes an IKEP up to 56% more balanced¹ without decreasing the overall number of transplants. Our simulations also indicate that out of the seven solution concepts, using the Banzhaf* value leads to the smallest deviations, namely on average, a deviation of at most 0.90% from the initial allocation (but the differences between the Banzhaf* value and the Banzhaf value or the Shapley value are small). Moreover, moving from $\ell = 2$ to $\ell = \infty$ yields on average 46% more kidney transplants (using the same simulation instances generated by the data generator (Pettersson & Trimble, 2021)).

For our simulations, we also kept track of the length of the exchange cycles. It turned out that these may be very large, in particular in the starting round. Previously, the relation between increase in transplants versus increase in cycle length was also researched by Biró (Biró et al., 2009), namely for $\ell = 2$, $\ell = 3$, and $\ell = \infty$. However, their simulations were done in the context of the UK KEP only.

2. The proofs of our theoretical results

We start with Theorem 3. Recall that the width c of a partitioned permutation game (N, v) defined on a directed graph $G = (V, A)$ with vertex partition (V_1, \dots, V_n) is the maximum size of a set V_i .

Theorem 3 (restated). *The core of every partitioned permutation game is nonempty, and it is possible to find a core allocation in polynomial time. Moreover, for partitioned permutation games of fixed width c , the problem of deciding if an allocation is in the core is polynomial-time solvable if $c = 1$ and coNP-complete if $c \geq 2$.*

Proof. We first show that finding a core allocation of a partitioned permutation game can be reduced to finding a core allocation of a permutation game. As the latter can be done in polynomial-time (Curiel & Tijs,

¹ As discussed in Section 4, using the Banzhaf value with $n = 10$ and equal country sizes yields an average relative deviation of 1.97% when selecting weakly close optimal solutions and an average relative deviation of 0.86% when selecting strongly close optimal solutions. So, the improvement in balancedness is $(1.97 - 0.86)/1.97 = 56\%$. For the other solution concepts, the improvements are between 36% and 54%.

Table 1

Survey of complexity results relevant for IKEPs, for fixed cycle length $\ell = 2, \ell \in \{3, 4, \dots\}$ and $\ell = \infty$, respectively, where c is the width of the associated partitioned ℓ -permutation game (N, v) . Here, poly stands for polynomial-time solvable; coNP-c for coNP-complete; NP-h for NP-hard; and coNP-h means coNP-hard. Results in **Bold** are new results shown in this paper, whereas non-referenced results follow directly from the referenced results.

	$\ell = 2$	$\ell \in \{3, 4, \dots\}$	$\ell = \infty$
finding an optimal solution	poly (Abraham et al., 2007)	NP-h (Abraham et al., 2007)	poly (Abraham et al., 2007)
testing core nonemptiness	poly if $c \leq 2$ coNP-hard if $c \geq 3$ (Benedek et al., 2025)	NP-h	poly (Tijs et al., 1984)
finding a core allocation	poly if $c \leq 2$ coNP-h if $c \geq 3$ (Benedek et al., 2025)	NP-h	poly (Curiel & Tijs, 1986)
deciding core membership	poly if $c \leq 2$ coNP-c if $c \geq 3$ (Benedek et al., 2025)	NP-h coNP-c if $c \geq 2$	poly if $c = 1$
finding a weakly close solution to target allocation	poly (Benedek et al., 2025)	NP-h	NP-h
finding a strongly close solution to target allocation	poly (Benedek et al., 2025)	NP-h	NP-h

1986) (and such a core allocation always exists (Tijs et al., 1984)), the same holds for the former.

Let (N, v) be a partitioned permutation game on a graph $G = (V, A)$ with partition (V_1, \dots, V_n) of V . We create a permutation game (N', v') by splitting each V_i into sets of size 1, i.e., every vertex becomes a player in N' . Let x' be a core allocation of (N', v') . For each player $i \in N$, we set $x_i = \sum_{v \in V_i} x'_v$, that is, we sum the x' values of the players in N' corresponding to the vertices that i controls in G . It holds that $x(N) = v(N)$, as (N, v) and (N', v') are defined on the same graph G ; hence, the weight of a maximum weight cycle packing is unchanged.

Suppose there is a blocking coalition $S \subset N$, that is, $v(S) > x(S)$ holds. By the construction of x , it holds that the sum of the x' values over all vertices in $\cup_{i \in S} V_i$ is less than $v(S) = v'(\{u \mid u \in \cup_{i \in S} V_i\})$. Hence, the players in N' corresponding to these vertices would form a blocking coalition to x' for (N', v') , a contradiction. As x' can be found in polynomial-time, so can x .

Now we show that deciding whether an allocation is in the core can be solved in polynomial time for partitioned permutation games with width 1, that is, for permutation games. Let $x \in \mathbb{R}^N$ be an allocation. We create a weight function w_x over the arcs by setting $w_x((u, v)) = x_u - 1$. We claim that if there exists a blocking coalition, then there is a blocking coalition that consists of only vertices along a cycle. In order to see this, let S be a blocking coalition, so $x(S) < v(S)$. By definition, $v(S)$ is the maximum size of a cycle packing $C = \{C_1, C_2, \dots, C_k\}$ in $G[S]$. For $i = 1, \dots, k$, let S_i be the set of vertices in C_i . From

$$x(S_1) + x(S_2) + \dots + x(S_k) \leq x(S) < v(S) = |S_1| + |S_2| + \dots + |S_k|,$$

we find that $x(S_i) < |S_i|$ for at least one set S_i . Hence, such an S_i is also blocking.

So we just need to check whether there is a cycle C_i with vertex set S_i such that $x(S_i) < |S_i|$. Note that $\sum_{(u,v) \in C_i} w_x(u, v) = x(S_i) - |S_i|$. Hence, such a cycle exists if and only if w_x is not conservative (where conservative means no negative weighted cycles exist). The latter can be decided in polynomial-time, for example with the Bellman-Ford algorithm.

Finally, we show that deciding if an allocation x is in the core of a partitioned permutation game is coNP-complete, even if each $|V_i|$ has size 2 (so $c = 2$). Containment in coNP holds, as we can check in polynomial time if a coalition blocks an allocation. To prove hardness, we reduce from a special case of the NP-complete problem EXACT 3-COVER (Hein et al., 1996; Hickey et al., 2008).

Given an instance I of EXACT 3-COVER, we construct a partitioned permutation game (N, v) as follows (see Fig. 3 for an illustration).

- For each element $i \in [3n]$, there is a vertex a_i and a vertex b_i ,
- for each set $S_j \in \mathcal{S}$, there are vertices $s_j^1, s_j^2, s_j^3, t_j^1, t_j^2, t_j^3$,
- there are a further $12n$ vertices x_1, \dots, x_{6n} and y_1, \dots, y_{6n} .

Define the arcs as follows:

- for each $k \in [6n]$, an arc (x_k, x_{k+1}) , where $x_{6n+1} := x_1$.

EXACT 3-COVER

Instance : A family of 3-element subsets of $[3n]$,
 $S = \{S_1, \dots, S_{3n}\}$, where each element belongs to exactly three sets

Question : Is there an exact 3-cover for S , that is, a subset $S' \subset S$ such that each element appears in exactly one of the sets of S' ?

- for each $k \in [3n]$, an arc (b_k, b_{k+1}) , where $b_{3n+1} := b_1$;
- for each $j \in [3n]$, the arcs $(t_j^1, t_j^2), (t_j^2, t_j^3), (t_j^3, t_j^1)$;
- for each $k \in [6n], j \in [3n], l \in [3]$, the arcs (y_k, s_j^l) and (s_j^l, y_k) ; and
- for each set $S_j = \{j_1, j_2, j_3\}, j_1 < j_2 < j_3$, the arcs $(s_j^1, a_{j_1}), (a_{j_1}, s_j^1), (s_j^2, a_{j_2}), (a_{j_2}, s_j^2), (s_j^3, a_{j_3}), (a_{j_3}, s_j^3)$.

This gives a directed graph $G = (V, A)$. The players with their corresponding partition of the vertices are:

- for each $i \in [3n]$, we have a player $A_i = \{a_i, b_i\}$,
- for each $k \in [6n]$, we have a player $X_k = \{x_k, y_k\}$, and
- for each $j \in [3n], l \in [3]$, we have a player $T_j^l = \{s_j^l, t_j^l\}$.

Finally, we define the allocation $x \in \mathbb{R}^N$, as follows:

- $x_{A_i} = 3 - \frac{n+1}{9n^2}$ for each $i \in [3n]$,
- $x_{X_k} = 3 - \frac{2n-1}{18n^2}$ for each $k \in [6n]$, and
- $x_{T_j^l} = 1 + \frac{1}{9n}$ for each $j \in [3n], l \in [3]$.

The size of the maximum cycle packing of G is

$$v(N) = 6n + 6n + 9n + 9n + 3n + 3n = 36n,$$

as every vertex can be covered. This is realized by adding the x_k -cycle, the b_i -cycle, the t_j^l -cycles and then for each a_i , a cycle $\{(a_i, s_j^l), (s_j^l, a_i)\}$ (this can be done, because each element appears exactly three times in the sets, so there is a perfect matching covering the vertex of each element in the bipartite graph induced by the incidence relation between the sets and the elements). We can cover the remaining $6n s_j^l$ vertices by two cycles $\{(y_k, s_j^l), (s_j^l, y_k)\}$ arbitrarily.

If we sum up all allocation values we find that

$$\begin{aligned} x(N) &= 6n \cdot \left(3 - \frac{2n-1}{18n^2}\right) + 9n \cdot \left(1 + \frac{1}{9n}\right) + 3n \cdot \left(3 - \frac{n+1}{9n^2}\right) \\ &= 18n - \frac{2n-1}{3n} + 9n + 1 + 9n - \frac{n+1}{3n} \\ &= 36n \\ &= v(N), \end{aligned}$$

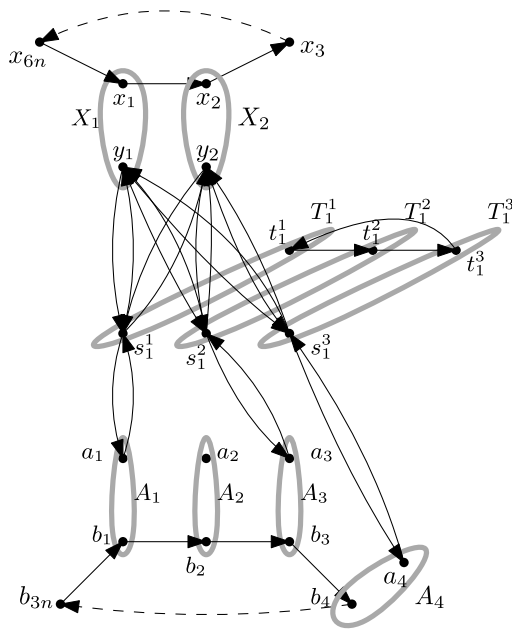


Fig. 3. An illustration for [Theorem 3](#). The figure shows the part of the construction for a set $S_1 = \{1, 3, 4\}$. Dashed arcs denote that there is a cycle through the given vertex whose vertices are not included in the picture. Dark grey bold ellipses denote the players.

so x is an allocation for (N, v) .

We claim I has an exact 3-cover if and only if x is not in the core.

“ \Rightarrow ” First suppose $\{S_{k_1}, \dots, S_{k_n}\}$ is an exact 3-cover in I . We claim that $\mathcal{P} = \{A_i \mid i \in [3n]\} \cup \{T_{k_l}^l \mid i \in [n], l \in [3]\}$ is a blocking coalition. We first show that $v(\mathcal{P}) = 12n$, which can be seen as follows. First, the s_j^l and a_i vertices in the coalition can be covered by 2-cycles, as the corresponding sets form an exact 3-cover. Moreover, the b_i vertices can be covered by the b_i -cycle, as each of the A_i players is in \mathcal{P} , and finally, the t_j^l vertices can be covered too, as for each $j \in \{k_1, \dots, k_n\}$ all of T_j^1, T_j^2, T_j^3 belong to \mathcal{P} . Then, x is not in the core, as

$$\begin{aligned} x(\mathcal{P}) &= 3n \cdot \left(3 - \frac{n+1}{9n^2}\right) + 3n \cdot \left(1 + \frac{1}{9n}\right) \\ &= 9n - \frac{n+1}{3n} + 3n + \frac{n}{3n} \\ &< 12n \\ &= v(\mathcal{P}). \end{aligned}$$

“ \Leftarrow ” Suppose x is not in the core. Then there is a coalition \mathcal{P} with $v(\mathcal{P}) > x(\mathcal{P})$. We write $\mathcal{A} = \bigcup_{i \in [3n]} A_i$ and $\mathcal{X} = \bigcup_{k \in [6n]} X_k$. We claim that $\mathcal{A} \subset \mathcal{P}$ or $\mathcal{X} \subset \mathcal{P}$. For a contradiction, suppose that neither $\mathcal{A} \subset \mathcal{P}$ nor $\mathcal{X} \subset \mathcal{P}$ holds. Clearly, $\mathcal{P} \cap (\mathcal{A} \cup \mathcal{X}) \neq \emptyset$, because m of the T_j^l players can only create a cycle packing of size h , but h of them have together an allocation of $h(1 + \frac{1}{9n}) > h$. So suppose that $|\mathcal{P} \cap (\mathcal{A} \cup \mathcal{X})| = m \leq 9n$. By our assumption, none of the vertices b_i or x_k vertices can be covered. Hence, if there are $h \geq 1$ participating T_j^l players besides them (there must be at least one to have any cycles), then the size of the maximum cycle packing they can obtain is $h + 2 \min\{m, h\} \leq 2m + h$, as at best all t_j^l vertices can be covered, but the other vertices can only be covered with cycles of length 2 by pairing the h s_j^l vertices to m a_i or x_i vertices. But, their assigned allocation in x is at least

$$m \cdot \min\left\{\left(3 - \frac{n+1}{9n^2}\right), \left(3 - \frac{2n-1}{18n^2}\right)\right\} + h \cdot \left(1 + \frac{1}{9n}\right) > 2m + h,$$

a contradiction. Hence, $\mathcal{A} \subset \mathcal{P}$ or $\mathcal{X} \subset \mathcal{P}$ holds.

First suppose that $\mathcal{A} \cup \mathcal{X} \subset \mathcal{P}$. Let the number of participating T_j^l players be h . Then, we have that

$$v(\mathcal{P}) \leq 3n + 6n + 2h + h, \text{ and}$$

$$v(\mathcal{P}) > x(\mathcal{P}) = 18n - \frac{2n-1}{3n} + 9n - \frac{n+1}{3n} + h + \frac{h}{9n}.$$

Hence, we find that $2h > 18n - 1 + \frac{h}{9n} > 18n - 1$, so $h > 9n - 1$, but it also cannot be $9n$, as $18n = 18n - 1 + \frac{9n}{9n}$, a contradiction (as there are only $9n$ T_j^l players).

Suppose next that $\mathcal{X} \subset \mathcal{P}$. Let $0 \leq m = |\mathcal{P} \cap \mathcal{A}| < 3n$. Then, if the number of T_j^l players in \mathcal{P} is $h > 0$, then $v(\mathcal{P}) \leq 6n + h + 2 \cdot h$. We can suppose that $h \leq 6n + m$, because if there are more T_j^l players, then at most $6n + m$ of their s_j^l vertices can be covered, hence the remaining players bring strictly more x value than what they can increase the maximum cycle packing size with. However,

$$\begin{aligned} x(\mathcal{P}) &\geq \left(18n - \frac{2n-1}{3n}\right) + \left(h + \frac{h}{9n}\right) + m \cdot \left(3 - \frac{n+1}{9n^2}\right) \\ &> 18n + h + 3m - 1. \end{aligned}$$

Hence, in order for \mathcal{P} to block, it must hold that $2h > 12n + 3m - 1$, so $h > 6n + 1.5m - 0.5$, contradicting $h \leq 6n + m$, if $m \geq 1$. In the case, when $m = 0$, we get that $6n + 3h > 18n - \frac{2n-1}{3n} + h + \frac{h}{9n}$, so $2h > 12n - \frac{2n-1}{3n} + \frac{h}{9n} > 12n - 1$. From this and $h \leq 6n + 0$, we get that h must be $6n$. However, then $12n > 12n - \frac{2n-1}{3n} + \frac{6n}{9n} > 12n$, which is a contradiction again.

Therefore, suppose that $\mathcal{A} \subset \mathcal{P}$, but \mathcal{X} is not included in \mathcal{P} . Let $0 \leq m = |\mathcal{P} \cap \mathcal{X}| < 6n$. Now, if the number of T_j^l players in \mathcal{P} is h , then $v(\mathcal{P}) \leq 3n + h + 2 \cdot h$, similarly as before. Again, we can suppose that $h \leq 3n + m$ for similar reasons. Furthermore,

$$\begin{aligned} x(\mathcal{P}) &\geq 9n - \frac{n+1}{3n} + h + \frac{h}{9n} + m \cdot \left(3 - \frac{2n-1}{9n^2}\right) \\ &> 9n + h + 3m - 1. \end{aligned}$$

If $m \geq 1$, then this implies that $2h > 6n + 3m - 1$, so $h > 3n + 1.5m - 0.5$, a contradiction. We conclude that $m = 0$. Therefore, $h \leq 3n$ and $2h > 6n - \frac{n+1}{3n} + \frac{h}{9n} > 6n - 1$, so $h = 3n$.

To sum up, we showed that \mathcal{P} must contain \mathcal{A} , must be disjoint from \mathcal{X} and there must be exactly $3n$ T_j^l players inside \mathcal{P} .

We claim that for each $j \in [3n]$, if $T_j^l \in \mathcal{P}$ for some $l \in [3]$, then T_j^l is inside \mathcal{P} for all $l \in [3]$: if not then there must be at least one t_j^l vertex that cannot be covered, hence $v(\mathcal{P}) \leq 12n - 1$, but $x(\mathcal{P}) = 9n - \frac{n+1}{3n} + 3n + \frac{3n}{9n} > 12n - 1$, a contradiction. Therefore, for each set S_j , if $T_j^l \in \mathcal{P}$ for some $l \in [3]$, then $T_j^l \in \mathcal{P}$ for all $l \in [3]$, so there are exactly n sets S_j , such that $T_j^l \in \mathcal{P}$.

Finally, it remains to show that the n sets corresponding to those value of j such that T_j^l is in \mathcal{P} for $l \in [3]$ must be the indices of an exact 3-cover. Suppose that there is an element i that cannot be covered by them. Then, a_i cannot be covered by a cycle packing by \mathcal{P} , so $v(\mathcal{P}) \leq 12n - 1$, which leads to the same contradiction. \square

Before we prove [Theorem 4](#), we need some definitions and a lemma. Let $G = (V, A)$ be a directed graph with a partition (V_1, \dots, V_n) of V for some $n \geq 1$. Recall that for a maximum cycle packing C of G , we let $s_p(C)$ denote the number of arcs (u, v) with $v \in V_p$ that belong to some directed cycle of C . We say that C satisfies a set of intervals $\{I_1, \dots, I_n\}$ if $s_p(C) \in I_p$ for every $p \in \{1, \dots, n\}$.

Lemma 1. For instances (G, \mathcal{V}, I) , where $G = (V, A)$ is a directed graph, $\mathcal{V} = (V_1, \dots, V_n)$ is a partition of V with fixed width c , and $I = \{I_1, \dots, I_n\}$ is a set of intervals, the problem of finding a maximum cycle packing of G satisfying I is polynomially solvable if $c = 1$, and NP-complete if $c \geq 2$ even if $I_p = [1, \infty]$ for every $p \in \{1, \dots, n\}$ or $I_p = [1, 1]$ for every $p \in \{1, \dots, n\}$.

Proof. First suppose $c = 1$. Let v_p be the unique vertex in V_p . We can assume that each I_p contains either 0 or 1, else no cycle packing satisfying I exists. If $1 \notin I_p$, we can delete v_p and redefine G as the graph that remains. If this decreases the size of the maximum cycle packing, then we conclude that no desired maximum cycle packing exists. Let U be the set of vertices v_p for which $0 \notin I_p$.

The problem reduces to finding a maximum cycle packing such that each vertex in U is covered. For this, we transform $G = (V, A)$ into a

bipartite graph H with partition classes V and V' , where V' is a copy of V . For each $v \in V \cup U$ and its copy $v' \in V' \cup U'$, we add the edge uv' with weight 0 (we do not add these for the vertices of U). For each $(u, v) \in A$, we add the edge uv with weight 1. It remains to find in polynomial time a maximum weight perfect matching in H , if there is any and check whether its weight is the same as the size of a maximum cycle packing in the original directed graph. If there is a perfect matching with that weight, then in the maximum cycle packing it corresponds to, each vertex in U must be covered with a cycle. In the other direction, if there is a maximum cycle packing covering each vertex in U , then the perfect matching it corresponds to has the desired weight and we need no nonexistent uv' edge for any $u \in U$ indeed.

Now suppose $c \geq 2$. Containment in NP is trivial, as we can check in polynomial time if an arc set consists of vertex disjoint cycles or not, and for each player we can compute the number of incoming arcs.

To prove completeness, as in the proof of [Theorem 3](#), we reduce from the NP-complete problem EXACT 3-COVER. Given an instance I of EXACT 3-COVER, we construct an instance I' of our problem as follows (see also [Fig. 4](#)):

- For each element $i \in [3n]$, we create a vertex a_i ,
- for each set $S_j \in \mathcal{S}$, we create vertices $s_j^1, s_j^2, s_j^3, t_j^1, t_j^2, t_j^3$,
- we create $2n$ source vertices x_1, \dots, x_{2n} and $2n$ sink vertices y_1, \dots, y_{2n} .

Define the arcs as follows:

- for each $k \in [2n]$, an arc (y_k, x_k) ,
- for each $k \in [2n], j \in [3n]$, the arcs (x_k, t_j^1) and (t_j^3, y_k) ,
- for each $j \in [3n]$, the arcs (t_j^1, t_j^2) and (t_j^2, t_j^3) , and
- for each set $S_j = \{j_1, j_2, j_3\}, j_1 < j_2 < j_3$, the arcs $(s_j^1, a_{j_1}), (a_{j_1}, s_j^1), (s_j^2, a_{j_2}), (a_{j_2}, s_j^2), (s_j^3, a_{j_3})$ and (a_{j_3}, s_j^3) .

This gives a directed graph $G = (V, A)$. We partition V into players as follows:

- for each $i \in [3n]$, we have a player $A_i = \{a_i\}$,
- for each $k \in [2n]$, we have a player $X_k = \{x_k\}, Y_k = \{y_k\}$, and
- for each $j \in [3n], l \in [3]$, we have a player $T_j^l = \{s_j^l, t_j^l\}$.

The maximum cycle packing of G has size $16n$. This is because the x_i, y_i vertices allow $2n$ cycles of length 5 through t_j^1, t_j^2, t_j^3 triples covering $10n$ vertices. The rest of the t_j^l vertices cannot be covered. Also, for the other s_j^l and a_i vertices, they span a directed bipartite graph, so at most $3n + 3n$ vertices can be covered, as we have only $3n$ a_i vertices. And $6n$ can be covered indeed, as we can just choose an arbitrary s_j^l neighbour for each a_i and pair them with a 2-cycle.

The interval for each player is $[1, \infty]$. Since the size of the maximum cycle packing of G is $16n$, which is the same as the number of players, if there is a solution that satisfies these intervals, then it also must satisfy the intervals $[1, 1]$ for each player. Hence the last two statements of the Lemma are equivalent in this instance.

As a maximum cycle packing in G has size $16n$, which is equal to the sum of the lower bounds, G has a cycle packing satisfying every interval if and only if G has a maximum cycle packing satisfying every interval. We claim I admits an exact 3-cover if and only if G admits a cycle packing satisfying every interval.

“ \Rightarrow ” First suppose I has an exact 3-cover S_1, \dots, S_{l_n} . We create a cycle packing C of G . For each $j \in \{1, \dots, l_n\}$, we add the cycles $\{(s_j^1, a_{j_1}), (a_{j_1}, s_j^1)\}, \{(s_j^2, a_{j_2}), (a_{j_2}, s_j^2)\}, \{(s_j^3, a_{j_3}), (a_{j_3}, s_j^3)\}$. For $j \notin \{1, \dots, l_n\}$ we add the arcs $(t_j^1, t_j^2), (t_j^2, t_j^3)$. Finally, for each $i \in [2n]$, we add the arcs $(y_i, x_i), (x_i, t_{j_i}^1), (t_{j_i}^3, y_i)$, where j_i is the i th smallest index among the indices $[3n] \setminus \{1, \dots, l_n\}$.

Clearly, C is a cycle packing. Each A_i has an incoming arc, as S_1, \dots, S_{l_n} is an exact 3-cover. As there are exactly $2n$ sets not in the set cover, all of the corresponding players T_j^l have one incoming arc in a cycle of the form $\{(y_i, x_i), (x_i, t_{j_i}^1), (t_{j_i}^1, t_{j_i}^2), (t_{j_i}^2, t_{j_i}^3), (t_{j_i}^3, y_i)\}$, and so did each X_i and each Y_i . Hence, all lower bounds are satisfied.

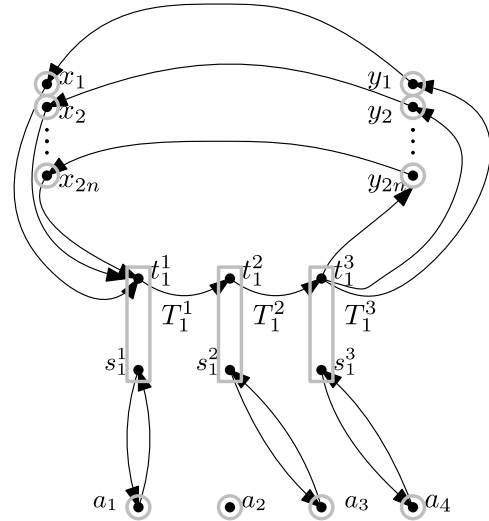


Fig. 4. Illustration for [Lemma 1](#), showing the construction for a set $S_1 = \{1, 2, 4\}$. Grey bold lines mark the players.

***q*-EXACT PERFECT MATCHING**

Instance : An undirected bipartite graph $B = (U, W; E)$, where each edge is coloured with one of $\{1, \dots, q\}$, and numbers k_1, \dots, k_q .

Question : Is there a perfect matching in B consisting of k_q edges of each colour q ?

“ \Leftarrow ” Now suppose G admits a cycle packing satisfying every interval. Then, as X_i has an incoming arc for all $i \in [2n]$, all (x_i, y_i) arcs are included in the cycle packing. This means that there are $2n$ such $j \in [3n]$, such that the arcs $(t_j^1, t_j^2), (t_j^2, t_j^3)$ are included in a cycle $\{(y_i, x_i), (x_i, t_j^1), (t_j^1, t_j^2), (t_j^2, t_j^3), (t_j^3, y_i)\}$ of C .

From the above, we have that there are n indices j from $[3n]$, such that none of the players T_j^1, T_j^2, T_j^3 have incoming arcs of this form. Hence, all these players can only have incoming arcs from a player A_i . As each such T_j^l must have one incoming arc, it follows that for all these j , the cycles $\{(a_{j_1}, s_j^1), (s_j^1, a_{j_1})\}, \{(a_{j_2}, s_j^2), (s_j^2, a_{j_2})\}, \{(a_{j_3}, s_j^3), (s_j^3, a_{j_3})\}$ are included in C , so they are vertex disjoint. Hence, the corresponding sets must form an exact 3-cover. \square

Theorem 4 (restated). For partitioned permutation games even of width 2, the problem of finding an optimal solution that is weakly or strongly close to a given target allocation x is NP-hard.

Proof. Recall that x_p denotes the target for the number of arcs (u, v) with $v \in V_p$ that belong to some directed cycle of C . Letting each $I_p = [x_p, x_p]$ and applying [Lemma 1](#), we see that finding a cycle packing where each $s_p(C)$ is equal x_p (so differs by at most 0) is NP-complete. Thus it is NP-hard to find the maximum cycle packing that minimizes $d_1(C) = \max_{p \in N} \{|x_p - s_p(C)|\}$; that is, to find a solution that is weakly close to a given target and similarly it is also hard to find a strongly close solution. \square

In the remainder of our paper, the following problem plays an important role:

For $q = 2$, this problem is also known as EXACT PERFECT MATCHING, which, as mentioned, was introduced by Papadimitriou and Yannakakis ([Papadimitriou & Yannakakis, 1982](#)) and whose complexity status is open for more than 40 years. In the remainder of this section, we will give both a reduction to this problem and a reduction from

this problem. We start with the former in the proof of our next result (Theorem 6), from which Theorem 5 immediately follows.

Let x be an allocation for a partitioned permutation game (N, v) on a graph $G = (V, A)$. For a maximum cycle packing C , $d'(C) = (|x_{p_1} - s_{p_1}(C)|, \dots, |x_{p_n} - s_{p_n}(C)|)$ is the unordered deviation vector of C .

Theorem 6. For a partitioned permutation game (N, v) on a directed graph $G = (V, A)$ and a target allocation x , it is possible to generate the set of unordered deviation vectors in $|A|^{O(n)}$ time by a randomized algorithm.

Proof. Let (N, v) be a partitioned permutation game with n players, defined on a directed graph $G = (V, A)$ with vertex partition V_1, \dots, V_n . As mentioned, we reduce to q -EXACT PERFECT MATCHING for an appropriate value of q . From (N, v) and a vector $d' = (d'_1, \dots, d'_n)$ with $d'_p \geq 0$ for every $p \in N$, we define an undirected bipartite graph $B = (U, W; E)$ with coloured edges: for each vertex $v \in V$, there is a vertex $v^{in} \in U$ and a vertex $v^{out} \in W$ and an edge $v^{in}v^{out} \in E$ that has colour $n + 1$; for each arc $(u, v) \in A$, there is an edge $u^{out}v^{in}$, which will be coloured p if $v \in V_p$. Let $k_{n+1} = |V| - v(N)$ and, for $p \in \{1, \dots, n\}$, let $k_p = d'_p$.

We observe that G has a maximum cycle packing C with $s_p(C) = d'_p$ if and only if B has a perfect matching with k_p edges of each colour $p \in \{1, \dots, n + 1\}$.

As each k_i can only have a value between 0 and $|E| = |A| + n = O(|A|)$, the above reduction implies that the set of unordered deviation vectors has size $|A|^{O(n)}$ for any allocation x for (N, v) . We can find each of these vectors in $|A|^{O(n)}$ time by a randomized algorithm, as q -EXACT PERFECT MATCHING is solvable in $|E|^{O(q)}$ time with q colours with a randomized algorithm (Gurjar et al., 2012). \square

We cannot hope to derandomize the algorithm from Theorem 6 without first solving 2-EXACT PERFECT MATCHING problem in polynomial time, as we now show.

Theorem 7. Every instance of 2-EXACT PERFECT MATCHING can be reduced in polynomial time to checking whether a partitioned permutation game (N, v) with only 2 players has a solution with no deviation from a target allocation x .

Proof. Take an instance $I = (B, k_1, k_2)$ of 2-EXACT PERFECT MATCHING. Let $B = (U, W; E)$ be the bipartite graph in I with $|U| = |W| = n$ for some integer n whose edges are coloured either red (colour 1) or blue (colour 2). We may suppose that $n = k_1 + k_2$, otherwise I is clearly a no-instance. We construct a digraph G from B by replacing every edge $e = uw$ with a directed 3-cycle on arcs $(u, v_e), (v_e, w), (w, u)$, where w_e is a new vertex that has only u and v as its neighbours in G .

Let the first player's set V_1 consist of all vertices v_e , for which e is a red edge in E and let the other player's vertex set be $V_2 = V \setminus V_1$. This defines a partitioned permutation game (N, v) . Let the target allocation x be given by $x = (k_1, 3n - k_1)$.

We claim that (B, k_1, k_2) is a yes-instance of 2-EXACT PERFECT MATCHING if and only if G has a cycle packing C with $s_1(C) = k_1$ and $s_2(C) = 3n - k_1$. Indeed, from a perfect matching M of B containing exactly k_1 red edges, we can substitute each edge $uw \in M$ by the corresponding 3-cycle (u, v_e, w) , such that exactly k_1 vertices from V_1 and $3n - k_1$ from V_2 are covered. It is also clear that this is a maximum size cycle packing, as at most n of the v_e vertices can be covered in any cycle packing. In the other direction, suppose we have a maximum size cycle packing with $s_1(C) = k_1$ and $s_2(C) = 3n - k_1$. Then, it covers k_1 of the vertices v_e corresponding to red edges and $n - k_1 = k_2$ of the vertices v_e corresponding to the blue edges. Also, it must be that for each such covered vertex v_e for $e = uw$, we have the arcs (u, v_e) and (v_e, w) in C . Therefore, these edges induce a perfect matching in B with exactly k_1 red and k_2 blue edges. \square

3. ILP formulation

In this section, we show how to find an optimal solution of a partitioned permutation game that is strongly close to a given target allocation x by solving a sequence of Integer Linear Programs (ILPs). Let

(N, v) be a partitioned permutation game defined on a directed graph $G = (V, A)$ with vertex partition V_1, \dots, V_n for some $n \geq 1$. Recall that for a maximum cycle packing C of G , we let $s_p(C)$ denote the number of arcs (u, v) with $v \in V_p$ that belong to some directed cycle of C . Recall also that $|x_p - s_p(C)|$ is the deviation of country $p \in N$ from its target x_p if C is chosen as optimal solution. Moreover, in the vector $d(C) = (|x_{p_1} - s_{p_1}(C)|, \dots, |x_{p_n} - s_{p_n}(C)|)$, the deviations $|x_p - s_p(C)|$ are ordered non-increasingly. Finally, we recall that C is strongly close to x if $d(C)$ is lexicographically minimal over all optimal solutions for (N, v) .

In the kidney exchange literature the following ILP is called the edge-formulation (see e.g. Abraham et al., 2007). For each $ij \in A$ in the graph G , let $e_{ij} \in \{0, 1\}$ be a (binary) edge-variable; this yields a vector e with entries e_{ij} .

$$\mu^* := \max_e \sum_{ij \in A} e_{ij} \quad \text{s.t.} \quad \text{(edge-formulation)}$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:ij \in A} e_{ij} \quad \forall i \in V \quad (1)$$

$$\sum_{j:ji \in A} e_{ji} \leq 1 \quad \forall i \in V \quad (2)$$

Constraint (1) represents the well-known Kirchoff law, whereas (2) ensures that every node is covered by at most one cycle. The objective function provides a maximum cycle packing, which has size μ^* .

This ILP has $|A|$ binary variables and $2|V|$ constraints. In the following we are going to sequentially find largest country deviations d_t^* ($t \geq 1$) and the corresponding minimal number n_t^* of countries receiving that deviation. We achieve this by solving an ILP of similar size for each d_t^* and n_t^* , so two ILPs per iteration t . By similar size, we mean that in each iteration we are going to add $|N|$ binary variables and a single additional constraint, while $|N| \leq |V|$ holds by definition and typically $|N|$ is much smaller than $|A|$. Meanwhile, since at every iteration we are going to fix the deviation of at least one additional country (we will not necessarily know which country, we are only going to keep track of the number of countries with fixed deviation), the number of iterations are at most $|N|$ (as $t \leq |N|$). Hence, we will solve no more than $2|N|$ ILPs, among which the largest has $O(|A| + |N|^2)$ binary variables and $O(|V| + |N|) = O(|V|)$ constraints.

Once we have μ^* we solve the following ILP to find d_1^* :

$$d_1^* := \min_{e, d_1} d_1 \quad \text{s.t.} \quad \text{(ILP}_{d_1})$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:ij \in A} e_{ij} \quad \forall i \in V \quad (1)$$

$$\sum_{j:ji \in A} e_{ji} \leq 1 \quad \forall i \in V \quad (2)$$

$$\sum_{ij \in A} e_{ij} = \mu^* \quad (3)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq d_1 \quad \forall p \in N \quad (4)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq d_1 \quad \forall p \in N \quad (5)$$

Constraints (1)–(3) guarantee that all solutions are, in fact, maximum cycle packings. These constraints will be part of the formulations throughout the entire ILP-series. Constraints (4) and (5), together with the objective function, guarantee that we minimize the largest country deviation. Note that for each country p , exactly one of $\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p$ and $x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij}$ is positive and exactly one is negative, unless both are zero. However, as soon as we reach $d_t^* \leq 1/2$ we have found a strongly close maximum cycle packing. Hence, in the remainder, we assume $d_t^* > 1/2$ for each t such that the series continues with $t + 1$.

(ILP $_{d_1}$) has one additional continuous variable (d_1) and $2|N| + 1$ additional constraints. For every country $p \in N$, we have that $|\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p| \leq d_1^*$. However, there exists a smallest subset

$N_1 \subseteq N$ (which may not necessarily be unique) such that

$$\left| \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \right| = d_1^* \quad \forall p \in N_1$$

$$\left| \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \right| \leq d_2^* < d_1^* \quad \forall p \in N \setminus N_1$$

In a solution of (ILP_{d_1}) , let n_1 be the number of countries with deviation d_1^* . We need to determine if there is another solution of (ILP_{d_1}) with fewer than n_1 , possibly n_1^* , countries having deviation d_1^* . For this purpose we must be able to distinguish between countries unable to have less than d_1^* deviation and countries for which the deviation is at most d_2^* , the latter value unknown at this stage. In order to make this distinction, we determine a lower bound on $d_1^* - d_2^*$ by examining the target allocation x . We will then set ϵ in the next ILP to be strictly smaller than this lower bound.

The number of vertices for a country covered by any cycle packing is an integer. Hence, the number of possible country deviations is at most $2|N|$ and depends only on x . The fractional part of a country deviation is either $\text{frac}(x_p)$ or $1 - \text{frac}(x_p)$. Therefore, to find the minimal positive difference in between the deviations of any two countries p and q , we have to compare the values $\text{frac}(x_p)$ and $1 - \text{frac}(x_p)$, with $\text{frac}(x_q)$ or $1 - \text{frac}(x_q)$ and take the minimum of those four possible differences. We let ϵ be a small positive constant that is weakly smaller than the minimum possible positive difference between any two countries except in the unique case when $\text{frac}(x_p) = 0.5$ for every $p \in N$. This is because in that case, $\text{frac}(x_p) = 1 - \text{frac}(x_p) = \text{frac}(x_q) = 1 - \text{frac}(x_q) = 0.5$, meaning that all four possible differences are zero. However, should this happen, then the minimum possible positive difference of any two countries' deviation can be trivially found as 1, and then we can simply take any ϵ with $0 < \epsilon \leq 1$.

We will distinguish between countries having minimal deviation of d_1^* and others through additional binary variables. Since later in the ILP series we will need to distinguish between countries fixed at different deviation levels, let us introduce binary variables z_p^t , where $z_p^t = 1$ indicates that $p \in N_t$.

$$\min_{z^1, e} \sum_{p \in N} z_p^1 \quad \text{s.t.} \quad (ILP_{N_1})$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:ij \in A} e_{ij} \quad \forall i \in V \quad (1)$$

$$\sum_{j:ji \in A} e_{ji} \leq 1 \quad \forall i \in V \quad (2)$$

$$\sum_{ij \in A} e_{ij} = \mu^* \quad (3)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq d_1^* - \epsilon(1 - z_p^1) \quad \forall p \in N \quad (6)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq d_1^* - \epsilon(1 - z_p^1) \quad \forall p \in N \quad (7)$$

As discussed, for each country p , the left hand side of either (6) or (7) is negative (i.e., would be satisfied even with $z_p^1 = 0$). For those countries whose deviation cannot be lower than d_1^* , however, the (positive) left hand side of either (6) or (7) will require $z_p^1 = 1$. Thus, given an optimal solution z^{1*} of (ILP_{N_1}) , let $n_1^* := \sum_{p \in N} z_p^{1*}$ be the minimal number of countries receiving the largest country deviations. It is guaranteed that the non-increasingly ordered country deviations at a strongly close maximum cycle packing starts with exactly n_1^* many d_1^* values, followed by some $d_2^* < d_1^*$. (ILP_{N_1}) has $|A| + |N|$ binary variables and $2|V| + 2|N| + 1$ constraints. Now, to find d_2^* , we solve the following ILP:

$$\min_{d_2, e, z^1} d_2 \quad \text{s.t.} \quad (ILP_{d_2})$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:ij \in A} e_{ij} \quad \forall i \in V \quad (1)$$

$$\sum_{j:ji \in A} e_{ji} \leq 1 \quad \forall i \in V \quad (2)$$

$$\sum_{ij \in A} e_{ij} = \mu^* \quad (3)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq d_1^* \quad \forall p \in N \quad (8)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq d_1^* \quad \forall p \in N \quad (9)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq d_2 + z_p^1 d_1^* \quad \forall p \in N \quad (10)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq d_2 + z_p^1 d_1^* \quad \forall p \in N \quad (11)$$

$$\sum_{p \in N} z_p^1 = n_1^* \quad (12)$$

(ILP_{d_2}) has $|A| + |N|$ binary variables and one continuous variable (d_2) with $2|V| + 4|N| + 2$ constraints, and guarantees that we find the minimal second-largest country deviation d_2^* while exactly n_1^* countries deviation is kept at d_1^* . Finding n_2^* follows a similar approach, where L is a large constant satisfying $L \geq 2(d_1^* - d_2^*)$:

$$\min_{z^1, z^2, e} \sum_{p \in N} z_p^2 \quad \text{s.t.} \quad (ILP_{N_2})$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:ij \in A} e_{ij} \quad \forall i \in V \quad (1)$$

$$\sum_{j:ji \in A} e_{ji} \leq 1 \quad \forall i \in V \quad (2)$$

$$\sum_{ij \in A} e_{ij} = \mu^* \quad (3)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq d_2^* - \epsilon(1 - z_p^2) + z_p^1 L \quad \forall p \in N \quad (13)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq d_2^* - \epsilon(1 - z_p^2) + z_p^1 L \quad \forall p \in N \quad (14)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq d_1^* \quad \forall p \in N \quad (8)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq d_1^* \quad \forall p \in N \quad (9)$$

$$\sum_{p \in N} z_p^1 = n_1^* \quad (12)$$

$$z_p^1 + z_p^2 \leq 1 \quad \forall p \in N \quad (15)$$

Subsequently we follow a similar approach for all $t \geq 3$, until either $|N| = n_1^* + n_2^* + \dots + n_t^*$ or we terminate because $d_t^* \leq 1/2$. Until reaching one of these conditions we iteratively solve the following two ILPs, introducing additional $|N|$ binary variables and an additional constraint to both. Let L be a large constant satisfying $L \geq d_t^*$, e.g. $L = d_{t-1}^*$.

$$\min_{d_t, e, (z^i)_{i=1}^{t-1}} d_t \quad \text{s.t.} \quad (ILP_{d_t})$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:ij \in A} e_{ij} \quad \forall i \in V \quad (1)$$

$$\sum_{j:ji \in A} e_{ji} \leq 1 \quad \forall i \in V \quad (2)$$

$$\sum_{ij \in A} e_{ij} = \mu^* \quad (3)$$

$$\sum_{i=1}^{t-1} z_p^i \leq 1 \quad \forall p \in N \quad (16)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq d_t + \sum_{i=1}^{t-1} z_p^i d_i^* \quad \forall p \in N \quad (17)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq d_t + \sum_{i=1}^{t-1} z_p^i d_i^* \quad \forall p \in N \quad (18)$$

$$\sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} - x_p \leq \sum_{i=1}^{t-1} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t-1} z_p^i\right)L \quad \forall p \in N \quad (19)$$

$$x_p - \sum_{j \in V_p} \sum_{i:ij \in A} e_{ij} \leq \sum_{i=1}^{t-1} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t-1} z_p^i\right)L \quad \forall p \in N \quad (20)$$

$$\sum_{p \in N} z_p^i = n_i^* \quad \forall i \in \{1, \dots, t-1\} \quad (21)$$

In the following formulation L' is a large constant satisfying $L' \geq d_t^* - \varepsilon$.

$$\min_{(z_p^i)_{i=1}^t, e} \sum_{p \in N} z_p^t \quad \text{s.t.} \quad (\text{ILP}_{N_t})$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:i \in A} e_{ij} \quad \forall i \in V \quad (1)$$

$$\sum_{j:ji \in A} e_{ji} \leq 1 \quad \forall i \in V \quad (2)$$

$$\sum_{ij \in A} e_{ij} = \mu^* \quad (3)$$

$$\sum_{i=1}^{t-1} z_p^i \leq 1 \quad \forall p \in N \quad (16)$$

$$\sum_{j \in V_p} e_{ij} - x_p \leq d_t^* - \varepsilon(1 - z_p^t) + \sum_{i=1}^{t-1} z_p^i d_i^* \quad \forall p \in N \quad (22)$$

$$x_p - \sum_{j \in V_p} e_{ij} \leq d_t^* - \varepsilon(1 - z_p^t) + \sum_{i=1}^{t-1} z_p^i d_i^* \quad \forall p \in N \quad (23)$$

$$\sum_{j \in V_p} e_{ij} - x_p \leq \sum_{i=1}^t z_p^i d_i^* + \left(1 - \sum_{i=1}^t z_p^i\right)L' \quad \forall p \in N \quad (24)$$

$$x_p - \sum_{j \in V_p} e_{ij} \leq \sum_{i=1}^t z_p^i d_i^* + \left(1 - \sum_{i=1}^t z_p^i\right)L' \quad \forall p \in N \quad (25)$$

$$\sum_{p \in N} z_p^i = n_i^* \quad \forall i \in \{1, \dots, t-1\} \quad (21)$$

From the above, we conclude that the following theorem holds, (in which the bounds on the number of ILPs, variables and constraints are not necessarily tight).

Theorem 8. For a partitioned permutation game (N, v) defined on a graph $G = (V, A)$, it is possible to find an optimal solution that is strongly close to a given target allocation x by solving a series of at most $2|N|$ ILPs, each having $O(|A| + |N|^2)$ binary variables and $O(|V|)$ constraints.

Note that if we just want to find a weakly close optimal solution we can stop after solving the first ILP.

4. Simulations

In this section we describe our simulations for $\ell = \infty$ in detail. Our goals are

1. to examine the benefits of strongly close optimal solutions over weakly close optimal solutions or arbitrarily chosen optimal solutions;
2. to examine the benefits of using credits;
3. to examine the exchange cycle length distribution; and
4. to compare all these results with those for the other extreme case where $\ell = 2$ (Benedek et al., 2024b).

For our fourth aim, we are especially interested in the increase in the total number of patients treated when we compare the $\ell = 2$ and $\ell = \infty$ cases.

4.1. The set up

To allow a fair comparison we use the same set up as in Benedek et al. (2024b), and in addition, we also use the data from Benedek (2021) that was used in Benedek et al. (2024b). That is, for our simulations, we take the same 100 compatibility graphs G_1, \dots, G_{100} , each with roughly² 2000 vertices from Benedek (2021), Benedek et al. (2024b). As real medical data is unavailable to us for privacy reasons, the data from Benedek (2021) was obtained using the data generator from Pettersson and Trimble (2021). This data generator was used in many papers and is the most realistic synthetic data generator available; see also Delorme et al. (2022).

We use every G_i ($i \in \{1, \dots, 100\}$) as the basis to perform simulations for n countries, where we let $n = 4, \dots, 10$. In order to do this, we first consider the case where all countries have equal size. That is, we partition the vertices V_i of the graph G_i into the same n sets $V_{i,1}, \dots, V_{i,n}$ as in Benedek et al. (2024b) which are of equal size $2000/n$ (subject to rounding), so $V_{i,p}$ is the set of patient-donor pairs of country p . For round 1, we construct a compatibility graph $G_i^1(n)$ as a subgraph of G_i of size roughly 500. We add the remaining patient-donor pairs of G_i as vertices by a uniform distribution between the remaining rounds $2, \dots, 24$. So, starting with $G_i^1(n)$, we run an IKEP of 24 rounds in total. This gives us 24 compatibility graphs $G_i^1(n), \dots, G_i^{24}(n)$. As in Benedek et al. (2024b), any patient-donor pair whose patient is not helped within four rounds will automatically be deleted from the pool.

We also consider a situation where countries have different sizes. We note that there are numerous ways one could allow for differently sized countries. However, we follow the approach of Benedek et al. (2024b), as this allows us to compare the results with those for $\ell = 2$. That is, we consider the same 100 compatibility graphs G_1, \dots, G_{100} as before; the only difference is that we now partition each V_i into the same n sets $V_{i,1}, \dots, V_{i,n}$ as in Benedek et al. (2024b), such that

- approximately $n/3$ are small, that is, have size roughly $1000/n$ (subject to rounding);
- approximately $n/3$ are medium, that is, have size roughly $2000/n$; and
- approximately $n/3$ are large, that is, have size roughly $3000/n$.

A (24-round) simulation instance consists of

- (i) the data needed to generate a graph $G_i^1(n)$ and its successors $G_i^2(n), \dots, G_i^{24}(n)$
- (ii) an indication of whether the countries must be of the same size or can have different sizes (in the way explained above);
- (iii) a choice of solution concept for computing the initial allocation (see Section 4.2); and
- (iv) a choice of type of optimal solution (see Section 4.3).

Our code for obtaining the simulation instances can be found in a GitHub repository (Ye, 2023), along with the compatibility graphs data and the seeds for the randomization.

We now discuss our choice for the initial allocations and optimal solutions.

4.2. The initial allocations

For the initial allocations y we use seven known solution concepts, which we all define below:

- the Shapley value,
- the Banzhaf value,
- the nucleolus,

² The graphs G_i might not have exactly 2000 vertices: depending on the specific number of countries that we consider, we may need to do some rounding in order for the countries to have the right sizes.

- the tau value,
- the benefit value
- the contribution value, and
- the Banzhaf* value (which, we recall, is a conceptually different solution concept).

We define each of these solution concepts below. Each of them prescribes exactly one allocation for the partitioned permutation games associated with the compatibility graphs. For computing these allocations we need the v -values of the partitioned permutation games.

We can compute a single value $v(S)$ in polynomial time by Theorem 1 through a transformation into a bipartite graph and solving a maximum weight perfect matching problem (see Section 1). For this we use the package LEMON version 1.3.1 of Dezsó et al. (2010).

Let (N, v) be a cooperative game. The Shapley value $\phi(N, v)$ (Shapley, 1953) is defined by setting for $p \in N$,

$$\phi_p(N, v) = \sum_{S \subseteq N \setminus \{p\}} \frac{|S|!(n - |S| - 1)!}{n!} (v(S \cup \{p\}) - v(S)).$$

To define the next solution concept, we first introduce unnormalized Banzhaf value $\psi(N, v)$ (Banzhaf, 1964) defined by setting for $p \in N$,

$$\psi_p(N, v) := \sum_{S \subseteq N \setminus \{p\}} \frac{1}{2^{n-1}} (v(S \cup \{p\}) - v(S)).$$

As ψ_p may not be an allocation, the (normalized) Banzhaf value $\bar{\psi}(N, v)$ of a game (N, v) was introduced and defined by setting for $p \in N$,

$$\bar{\psi}_p(N, v) := \frac{\psi_p(N, v)}{\sum_{q \in N} \psi_q(N, v)} \cdot v(N).$$

Whenever we mention the Banzhaf value, we will mean $\bar{\psi}(N, v)$. To compute the Shapley value and Banzhaf value in our simulations for even up to $n = 10$ countries, we were still able to implement a brute force approach relying on the above definitions.

We now define the nucleolus. The excess for an allocation x of a game (N, v) and a nonempty coalition $S \subsetneq N$ is defined as $e(S, x) := x(S) - v(S)$. Ordering the $2^n - 2$ excesses in a non-decreasing sequence yields excess vector $e(x) \in \mathbb{R}^{2^n - 2}$. The nucleolus η of a game (N, v) is the unique allocation (Schmeidler, 1969) that lexicographically maximizes $e(x)$ over the set of allocations x with $x_i \geq v(\{i\})$ (assuming this set is nonempty, as is the case for partitioned permutation games). To compute the nucleolus, we use the Lexicographical Descent method of Benedek et al. (2021), which is the state-of-the-art method in nucleolus computation.

Next, we define the tau value, introduced by Tijs (1981). First, let $b_p = v(N) - v(N \setminus p)$ be the utopia payoff of $p \in N$, leading to a vector of utopia payoffs $b \in \mathbb{R}^N$. Now, for $p \in S \subseteq N$, we let

$$R(S, p) = v(S) - \sum_{q \in S \setminus p} b_q$$

be the remainder for player p in S , which is what would remain if all players apart from p would leave a coalition S with their utopia payoff. We now set for $p \in N$,

$$a_p = \max_{S \ni p} R(S, p),$$

leading to the vector $a \in \mathbb{R}^N$ of minimal rights. A game (N, v) is quasibalanced if $a \leq b$ and $a(N) \leq v(N) \leq b(N)$. For a quasibalanced game, the tau value τ of a game (N, v) is defined by setting for $p \in N$,

$$\tau_p = \gamma a_p + (1 - \gamma) b_p,$$

where $\gamma \in [0, 1]$ is determined by $\tau(N) = v(N)$. Note that γ is unique, unless $a = b$ (in which case we can simply take $\tau = b$).

The tau value is only defined for quasibalanced games. However, all partitioned permutation games have a nonempty core by Theorem 3, and thus they are quasibalanced. As for the Shapley value and Banzhaf value, we were still able to compute the tau value by using a brute force approach based on its definition even for $n = 10$ countries.

The surplus of a game (N, v) is defined as $\text{surp} = v(N) - \sum_{p \in N} v(\{p\})$. As long as $\sum_{p \in N} \alpha_p = 1$, we can allocate $v(\{p\}) + \alpha_p \cdot \text{surp}$ to each player $p \in N$. In this way we obtain the benefit value (Klimentova et al., 2021) by setting for each $p \in N$,

$$\alpha_p = \frac{v(N) - v(N \setminus \{p\}) - v(\{p\})}{\sum_{q \in N} (v(N) - v(N \setminus \{q\}) - v(\{q\}))},$$

and we obtain the contribution value (Benedek et al., 2022) by setting for each $p \in N$,

$$\alpha_p = \frac{v(N) - v(N \setminus \{p\})}{\sum_{q \in N} (v(N) - v(N \setminus \{q\}))}.$$

Both the benefit value and contribution value are easy to compute, as we only need to compute $2n - 1$ v -values. We also note that the benefit value and contribution value do not exist if the denominator is zero. However, this did not happen in our simulations.

Finally, we consider a recent variant of the Banzhaf value, which was introduced in Benedek et al. (2024b). We first define for some round h in an IKEP the credit-adjusted game (N^h, \bar{v}^h) , in which the credits are directly incorporated, that is, $\bar{v}^h(S) = v^h(S) + \sum_{p \in S} c_p^h$ for every $S \subseteq N$. As explained in Benedek et al. (2024b), this alternative way of processing credits does not make any difference for any of the solution concepts that we consider except for one: the (normalized) Banzhaf value. We denote the adjusted Banzhaf value as the Banzhaf* value. In our simulations, we compute it by using brute force as we do for the Banzhaf value.

The solution concepts defined above have several fairness properties. For example, they may prescribe an allocation x_i for a game (N, v) that is individual rational, that is, $x_p \geq v(\{p\})$ for every $p \in N$. This property holds for the nucleolus, benefit value and contribution value by definition. For the tau value, which is only defined on quasibalanced games, it follows from the observation that for every $p \in N$, $\tau_p \geq a_p \geq v(\{p\})$. The Shapley value and unnormalized Banzhaf value are not individual rational in general. However, they are readily seen to be individual rational for superadditive games, i.e., games (N, v) for which $v(S \cup T) \geq v(S) + v(T)$ for all pairs $S, T \subsetneq N$ with $S \cap T = \emptyset$, which include the partitioned permutation games. In contrast, the normalized Banzhaf value is not individual rational for partitioned permutation games, as the following example shows.

Example 3. Let $G = (\{a, b, c\}, \{(a, b), (b, a), (b, c), (c, b)\})$ and $V_1 = \{a\}$, $V_2 = \{b\}$ and $V_3 = \{c\}$. Now, $\eta = (0, 2, 0)$ is the only core allocation, and we also note that $\phi_1, \psi_1, \bar{\psi}_1 > 0$. If we extend G by adding $V_4 = \{d, e\}$ with arcs $(d, e), (e, d)$, then $\psi_4 = v(\{4\}) = 2 > \bar{\psi}_4$.

As discussed in Section 1, the nucleolus η is not only individual rational but even belongs to the core (so $\eta(S) \geq v(S)$ for every $S \subsetneq N$). For a more detailed description on these solution concepts and their fairness properties, we refer to Peters (2008), Young (1985). Here, we only highlight one other fairness property to illustrate that policy makers must make a choice on what fairness properties they find most important (in addition to any computational considerations). A solution concept for a game (N, v) is coalitional monotone if for every $i \in N$ the following holds: if the value of each of the coalitions player i is a member of weakly increases, while the values of all the other coalitions remain the same, then the payoff of player i must also weakly increase. The Shapley value and the Banzhaf value have this fairness property, whilst the nucleolus does not (Young, 1985). However, whereas the nucleolus always belongs to the core of the game, Example 3 shows that the Shapley and Banzhaf values of permutation games may not be core allocations. In fact, Young (1985) even proved that no coalition monotone solution concept can guarantee to prescribe a core allocation.

Fairness properties, such as individual rationality, core membership and coalition monotonicity, are highly relevant for decision and policy making in international kidney exchange as well. As we have seen, such properties may be in conflict with each other. Hence, IKEPs should decide which fairness properties they prefer when selecting a certain

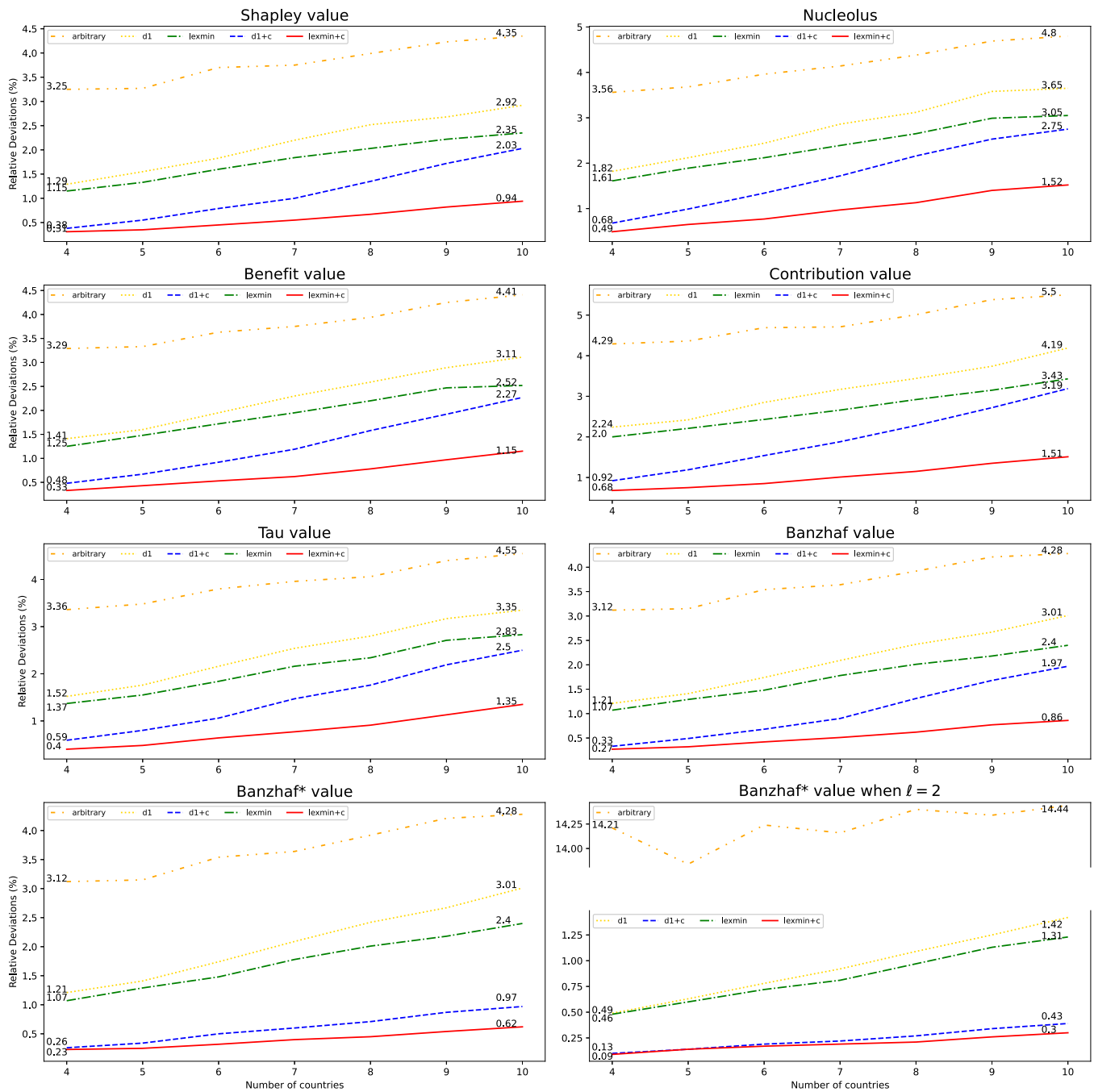


Fig. 5. Average total relative deviations for each of the seven solution concepts under the five different scenarios for equal country sizes, where the number of countries n is ranging from 4 to 10. For comparison, the lower right figure displays a result from Benedek et al. (2024b) for $\ell = 2$, namely for the Banzhaf* value, which behaved best for $\ell = 2$. We recall that the Banzhaf value and Banzhaf* value coincide when credits are not incorporated, and this is also reflected in the two corresponding figures.

solution concept. We also note that (fractional) target allocations are typically not possible to achieve exactly. This is, for instance, illustrated by Example 3, where for the original compatibility graph G , the nucleolus is $(0, 2, 0)$ while the only two optimal solutions are $(1, 1, 0)$ and $(0, 1, 1)$. However, the credit system offers a way to compensate for any deviations from target allocations in the long term.

4.3. The optimal solutions

As optimal solutions we compute the following for each compatibility graph:

1. an arbitrary optimal solution;
2. a weakly close optimal solution; and
3. a strongly close optimal solution.

In this way we can measure the effect of using weakly close optimal solutions over arbitrarily chosen ones as well as the effect of using strongly close optimal solutions over weakly close ones. We compute the weakly and strongly close optimal solutions by solving a sequence of ILPs, as described in Section 3. We use Gurobi Optimizer version 10.0.02(linux64) to create environments, build and solve ILPs (via the Gurobi C++ interface on a cloud system). As not every ILP might be solved within a reasonable time, we give each ILP a time limit of one hour. As a minor

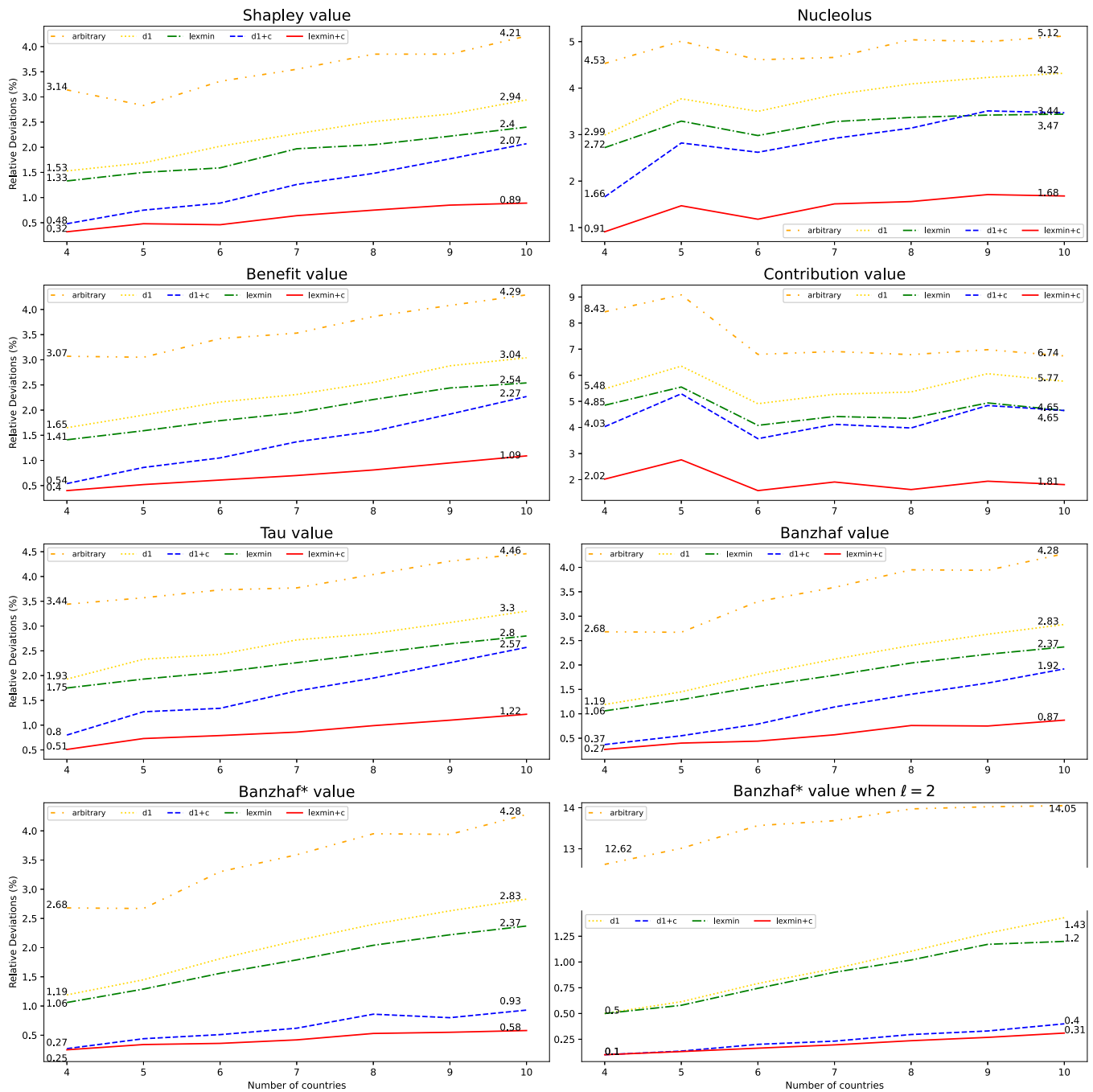


Fig. 6. Average total relative deviations for each of the seven solution concepts under the five different scenarios for varying country sizes, where the number of countries n is ranging from 4 to 10. For comparison, the lower right figure displays a result from Benedek et al. (2024b) for $\ell = 2$, namely for the Banzhaf* value, which behaved best for $\ell = 2$.

comment, we recall that the values $v(N)$ of the partitioned permutation games used in the ILPs have already been computed for the initial allocations, and we simply reuse these to save some computation time.

4.4. Computational environment and scale

In our large-scale experimental study, we run our simulations both without and with (“+c”) the credit system, and as mentioned for the settings, where an arbitrary optimal solution (“arbitrary”), weakly optimal solution (“d1”) or strongly close optimal solution (“lexmin”) is chosen. This leads to the following five scenarios for each of the six selected

solution concepts, the Shapley value, Banzhaf value, nucleolus, tau value, benefit value and contribution value:

- arbitrary,
- d1,
- d1 + c,
- lexmin,
- lexmin + c;

plus two scenarios for the Banzhaf* value:

- d1 + c,
- lexmin + c,

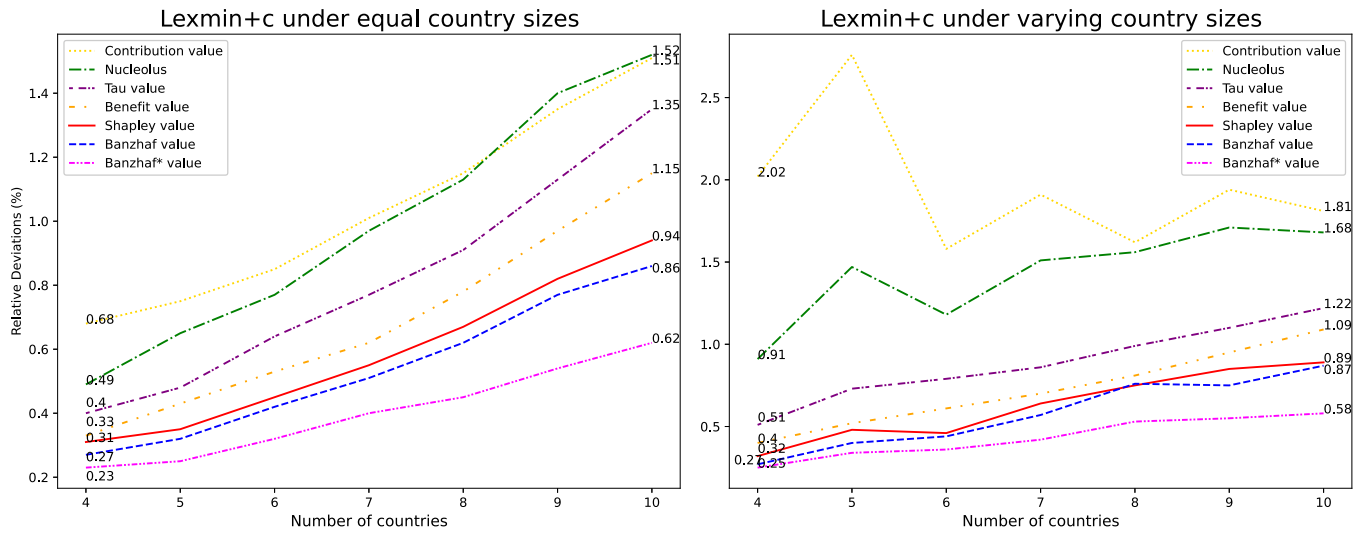


Fig. 7. Average total relative deviations for all solution concepts in the *lexmin+c* scenario, where the number of countries n ranges from 4 to 10.

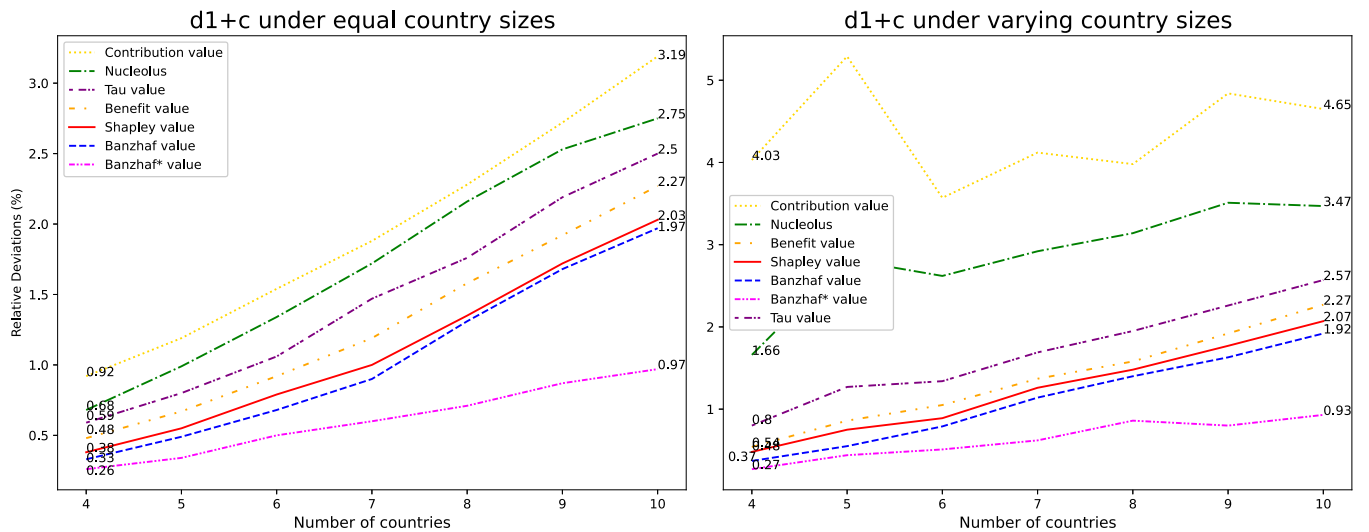


Fig. 8. Average total relative deviations for all solution concepts in the *d1+c* scenario, where the number of countries n ranges from 4 to 10.

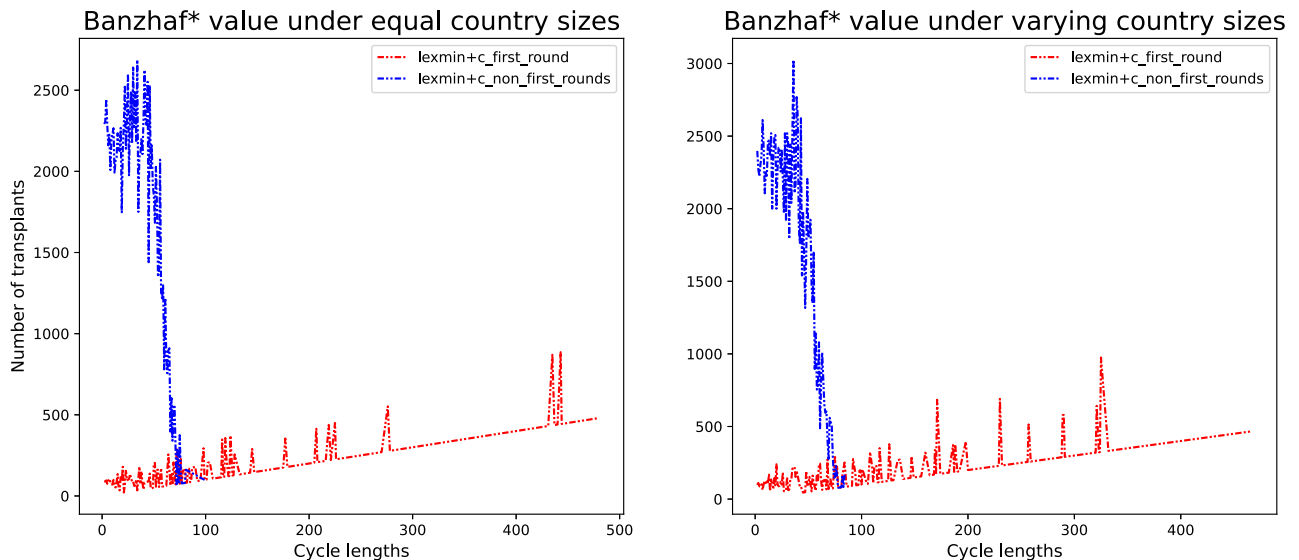


Fig. 9. Cycle distribution for the Banzhaf* value in the *lexmin+c* scenario, where the x-axis represents the cycle length, and the y-axis shows the number of kidney transplants involved in a cycle of that length.

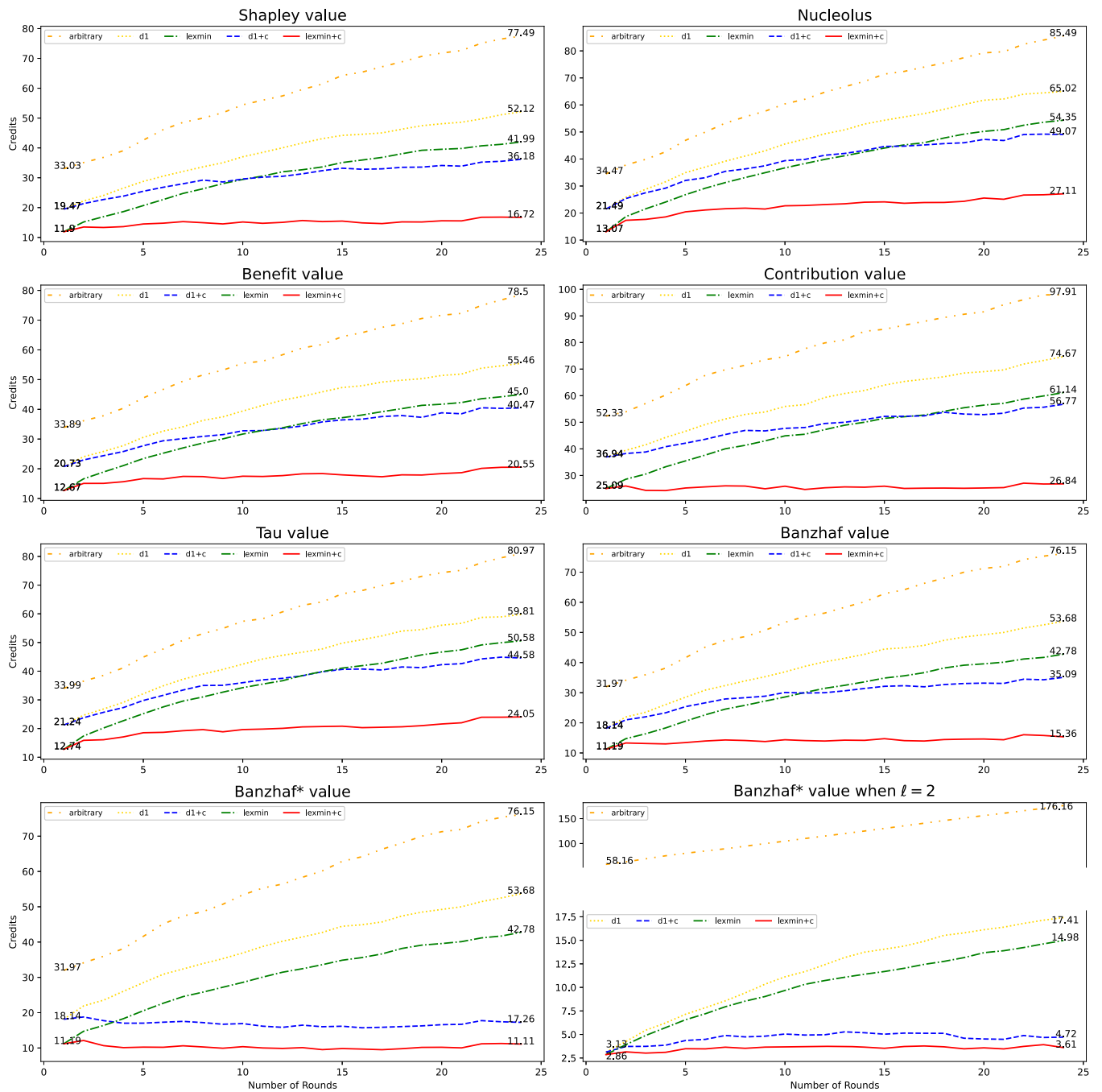


Fig. 10. Average credits for each of the seven solution concepts under the five different scenarios for equal country sizes, where the number of countries is $n = 10$ and the period ranges from 1 to 24. For comparison, the lower right figure displays a result from Benedek et al. (2024b) for $\ell = 2$, namely for the Banzhaf* value, which also behaved best for $\ell = 2$.

as without credits the Banzhaf* value coincides with the Banzhaf value.

Note that for the *arbitrary* scenario, the use of credits is irrelevant. Hence, in total, we run the same set of simulations for $5 \times 6 + 2 = 32$ different combinations of scenarios and solution concepts. We consider both the situation where all countries have the same size and a situation where they have different sizes. Moreover, we vary the number of countries n from 4 to 10 (so seven different numbers) and have 100 compatibility graphs G_i . Hence, the scale of our experimental study is very large. Namely, the total number of 24-round simulation instances is

$$32 \times 2 \times 7 \times 100 = 44800.$$

All simulations were run on a dual socket server with AMD EPYC 7702 64-Core Processor with 2.00 GHz base speed and 256GB of RAM, where each simulation was given eight cores and 1G temporary disk space. As we will compare our results for $\ell = \infty$ with the known results for $\ell = 2$ from Benedek et al. (2024b), let us point out that the simulations in Benedek et al. (2024b) were run on a desktop PC with AMD Ryzen 9 5950X 3.4GHz CPU and 128 GB of RAM, running on Windows 10 OS and C++ implementation in Visual Studio (the code is available in GitHub repository (Benedek, 2021)).

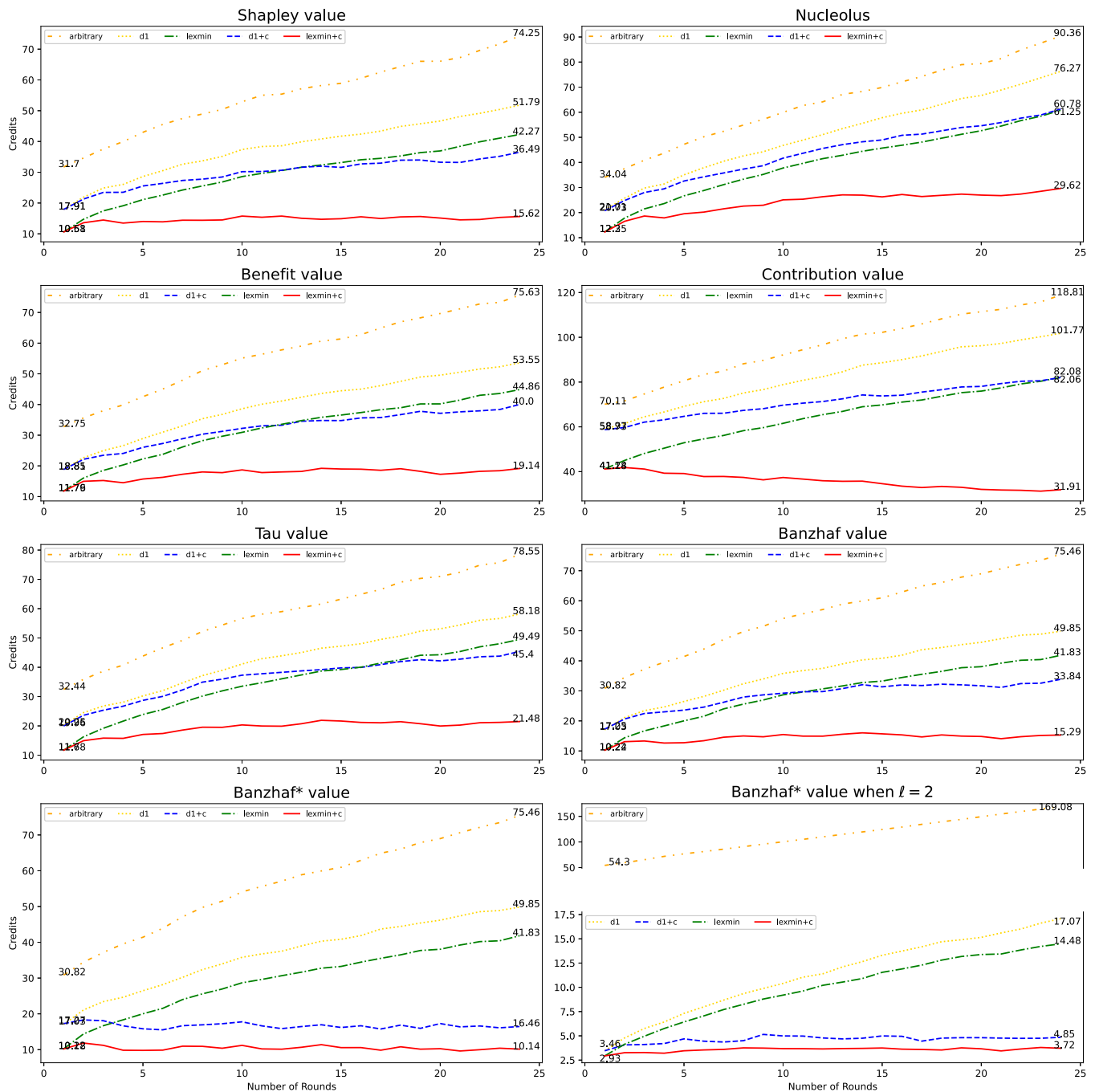


Fig. 11. Average credits for each of the seven solution concepts under the five different scenarios for varying country sizes, where the number of countries is $n = 10$ and the period ranges from 1 to 24. For comparison, the lower right figure displays a result from Benedek et al. (2024b) for $\ell = 2$, namely for the Banzhaf* value, which also behaved best for $\ell = 2$.

4.5. Evaluation measures

Let y^* be the total initial allocation of a single simulation instance, that is, y^* is obtained by taking the sum of the 24 initial allocations of each of the 24 rounds of that instance. Let C^* be the union of the chosen maximum cycle packings in each of the 24 rounds. We use the total relative deviation defined as

$$\frac{\sum_{p \in N} |y_p^* - s_p(C^*)|}{|C^*|}.$$

For each choice of solution concept, choice of scenario and choice of number of countries, we run 100 simulation instances. We take the average of the 100 total relative deviations to obtain the average total relative

deviation. Taking the maximum relative deviation

$$\frac{\max_{p \in N} |y_p^* - s_p(C^*)|}{|C^*|}$$

gives us the average maximum relative deviation as our second evaluation measure. As we shall see, both evaluation measures lead to the same conclusions.

4.6. simulation results for $\ell = \infty$ and comparison with $\ell = 2$

In this section, we present our simulation results for $\ell = \infty$ and compare these results with the simulation results for $\ell = 2$ (Benedek et al.,

2024b). We start with Fig. 5. In this figure we display our main results for equal country sizes, that is, when all countries are of the same size (see also Section 4.1), and we compare the seven different solution concepts under the five different scenarios for $\ell = \infty$ with each other. Fig. 5 also shows the effects of weakly and strongly close solutions and the credit system. As solution concepts have different complexities, we believe such a comparison might be helpful for policy makers in choosing a specific solution concept and scenario. Fig. 5 shows that using an arbitrary maximum cycle packing in each round indeed makes the kidney exchange scheme significantly more unbalanced, with average total relative deviations over 4% for all initial allocations y . Fig. 5 also shows that the effect of both selecting a strongly close solution (to ensure being close to a target allocation) and using a credit function (for fairness, to keep deviations small) is significant. Finally, we observe that for all solution concepts and all scenarios, the deviations are slowly increasing when the number of countries is increasing. However, the rate of increase is slower for strongly close optimal solutions than for weakly close optimal solutions, while the “starting” point of the line is lower when credits are employed.

The above conclusions are in line with the results under the setting where $\ell = 2$ (Benedek et al., 2024b). However, for $\ell = 2$, the effect of using arbitrary optimal solutions is much worse, while deviations are smaller than for $\ell = \infty$ when weakly close or strongly close optimal solutions are chosen. In Fig. 5 we illustrated this for the Banzhaf* value. From Fig. 5 we see that the Banzhaf* value in the $lexmin + c$ scenario provides the smallest deviations from the target allocations (as when $\ell = 2$ (Benedek et al., 2024b)). However, all solution concepts are within 1.52% (for $lexmin + c$) and, as mentioned, which solution concept to select should be decided by the policy makers of the IKEP.

Fig. 6 shows the same kind of results as Fig. 5 but now for varying country sizes, that is, when we have small, medium and large countries, divided exactly as in Benedek et al. (2024b) for $\ell = 2$ (see also Section 4.1). Note that subject to minor fluctuations we can draw the same conclusions from Fig. 6 for varying country sizes as we did from Fig. 5 for equal country sizes.

Figs. 5 and 6 highlight the comparison between different scenarios. For an easier comparison between the effects of choosing different solution concepts for prescribing the initial allocations, we grouped together all the $lexmin + c$ plots in of Figs. 5 and 6 in Fig. 7 and all the $d1 + c$ plots from Figs. 5 and 6 in Fig. 8. Figs. 7 and 8 both show an ordering from the Banzhaf* value, which has the best performance, to the contribution value which has the worst in almost all cases for the two most important scenarios $lexmin + c$ and $d1 + c$.

We note that Figs. 7 and 8 also show that (as expected) the effect of varying the country sizes is stronger if the number n of countries is relatively small, especially when $n \in \{4, 5, 6\}$.

Our Second Evaluation Measure. We can draw exactly the same conclusions as above if we use the average maximum relative deviation instead of the average total relative deviation. **The Appendix of the paper can be found in a separate Supplementary Material.** We refer to Appendix A for the analogs of Figs. 6–8 if we use the average maximum relative deviation as our evaluation measure.

Incomplete Instances. In the simulations for $\ell = 2$ (Benedek et al., 2024b), ILPs were not used, and all simulation instances were solved to optimality. However, our new simulations for $\ell = \infty$ heavily relied on ILPs, and we recall that we imposed a time limit of one hour for our ILP solver (Gurobi) to solve an ILP. Given the nature of ILPs, it is not surprising that there were several ILPs that could not be solved within the 1-h time limit. We call the corresponding simulation instances *incomplete*. So, for such instances, there was one ILP in some round, which our ILP solver could not handle within the 1-h time limit. We call other simulation instances *complete*.

All ILPs that were not finished within one hour were of the (ILP_{d_t}) type, almost always for $t > 1$. In fact, for equal country sizes, only one single incomplete simulation instance occurred at $d1 / d1 + c$ (for

Table 2

Average number of incomplete simulation instances for equal country sizes.

#unsolved / n	4	5	6	7	8	9	10	Total
Shapley value	0.00%	0.00%	0.00%	1.50%	1.75%	6.75%	3.50%	1.93%
Nucleolus	0.00%	0.00%	0.00%	0.50%	0.00%	0.00%	0.25%	0.11%
Benefit value	0.25%	0.25%	1.25%	0.25%	0.75%	0.75%	0.00%	0.50%
Contribution value	0.75%	0.75%	1.75%	1.00%	1.50%	0.50%	2.25%	1.21%
Banzhaf value	0.00%	0.75%	1.50%	1.75%	0.50%	4.25%	4.50%	1.89%
Tau value	0.00%	0.25%	0.00%	0.00%	0.00%	1.00%	0.00%	0.18%
Banzhaf* value	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.50%	0.07%
total: d1	0.00%	0.00%	0.00%	0.00%	0.00%	0.17%	0.00%	0.02%
total: lexmin	0.17%	0.33%	0.50%	1.33%	0.83%	4.17%	3.33%	1.52%
total: d1 + c	0.00%	0.00%	0.00%	0.00%	0.00%	0.14%	0.00%	0.02%
total: lexmin + c	0.43%	0.86%	2.14%	1.71%	1.86%	3.71%	3.29%	2.00%

Table 3

Average number of incomplete simulation instances for varying country sizes.

#unfinished / n	4	5	6	7	8	9	10	Total
Shapley value	0.00%	0.00%	0.00%	0.50%	0.00%	0.50%	2.00%	0.43%
Nucleolus	0.00%	0.00%	0.00%	0.00%	0.00%	0.75%	0.00%	0.11%
Benefit value	0.25%	0.00%	0.75%	0.00%	0.00%	0.25%	0.75%	0.29%
Contribution value	0.00%	0.00%	0.50%	0.00%	0.50%	0.50%	0.25%	0.25%
Banzhaf value	0.00%	1.00%	0.25%	0.25%	1.25%	0.50%	2.00%	0.75%
Tau value	0.00%	0.00%	0.00%	0.00%	0.25%	0.00%	0.00%	0.04%
Banzhaf* value	0.00%	1.00%	0.00%	0.00%	0.50%	0.50%	2.00%	0.57%
total: d1	0.00%	0.17%	0.00%	0.00%	0.00%	0.17%	0.00%	0.05%
total: lexmin	0.00%	0.17%	0.33%	0.33%	0.50%	0.50%	1.67%	0.50%
total: d1 + c	0.00%	0.29%	0.14%	0.00%	0.14%	0.00%	0.00%	0.08%
total: lexmin + c	0.14%	0.29%	0.43%	0.14%	0.71%	1.00%	2.00%	0.67%

Table 4

Average number of kidney transplants for equal country sizes for $n = 10$ when $\ell = 2$ (Benedek et al., 2024b) and $\ell = \infty$.

Solution concepts/scenarios	arbitrary	d1	d1 + c	lexmin	lexmin + c
$\ell = \infty$	benefit value	1781.58	1781.65	1781.46	1782.58
	contribution value	1781.58	1781.52	1780.50	1782.52
	Nucleolus	1781.58	1781.80	1780.45	1781.98
	Shapley value	1781.58	1781.54	1780.92	1782.02
	Banzhaf value	1781.58	1781.83	1780.38	1782.46
	Banzhaf* value	1781.58	1781.83	1781.08	1782.46
Tau value	1781.58	1782.22	1780.90	1782.12	
$\ell = 2$	benefit value	1221.38	1222.4	1223.44	1222.54
	contribution value	1221.38	1222.58	1222.56	1223.24
	Nucleolus	1221.38	1222.28	1221.18	1221.84
	Shapley value	1221.38	1221.6	1221.2	1223.04
	Banzhaf value	1221.38	1222.62	1221.26	1220.9
	Banzhaf* value	1221.38	1222.62	1220.86	1220.9
Tau value	1221.38	1223.96	1223.24	1224.26	

Table 5

Average number of kidney transplants for varying country sizes for $n = 10$ when $\ell = 2$ (Benedek et al., 2024b) and $\ell = \infty$.

Solution concepts/scenarios	arbitrary	d1	d1 + c	lexmin	lexmin + c
$\ell = \infty$	benefit value	1763.57	1763.39	1763.37	1763.73
	contribution value	1763.57	1763.45	1762.90	1763.80
	Nucleolus	1763.66	1764.02	1762.79	1763.74
	Shapley value	1764.04	1763.09	1762.99	1763.71
	Banzhaf value	1763.24	1763.34	1762.85	1763.47
	Banzhaf* value	1763.24	1763.34	1763.09	1763.47
Tau value	1763.27	1763.54	1762.99	1763.62	
$\ell = 2$	benefit value	1203.8	1206.14	1205.92	1205.06
	contribution value	1204.06	1205.7	1205.46	1205.36
	Nucleolus	1204.06	1205.34	1205.38	1205.52
	Shapley value	1204.06	1205.7	1206.64	1206.34
	Banzhaf value	1204.06	1206.14	1205.56	1205.88
	Banzhaf* value	1204.06	1206.14	1206.34	1205.88
Tau value	1204.06	1206.86	1205.92	1207.32	

Table 6
Relative ratios for equal country sizes under $lexmin + c$, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	46.82%	39.70%	38.99%	34.55%	31.40%	31.41%	28.17%
	$\ell = 2$	41.70%	37.92%	35.29%	30.85%	28.00%	26.71%	26.52%
	difference	5.12%	1.78%	3.70%	3.70%	3.40%	4.69%	1.66%
Banzhaf value	$\ell = \infty$	45.91%	40.92%	39.50%	36.87%	30.95%	30.57%	27.61%
	$\ell = 2$	43.90%	37.07%	34.88%	30.05%	29.27%	26.89%	26.26%
	difference	2.01%	3.85%	4.62%	6.82%	1.68%	3.68%	1.35%
nucleolus	$\ell = \infty$	47.35%	39.50%	38.25%	35.21%	32.95%	29.81%	27.02%
	$\ell = 2$	45.85%	40.05%	36.57%	33.68%	30.84%	29.44%	26.75%
	difference	1.51%	-0.54%	1.67%	1.52%	2.11%	0.36%	0.27%
tau value	$\ell = \infty$	47.11%	41.16%	39.47%	34.02%	33.91%	31.57%	27.07%
	$\ell = 2$	42.00%	38.53%	37.70%	32.41%	30.31%	28.68%	27.69%
	difference	5.11%	2.63%	1.78%	1.62%	3.60%	2.89%	-0.62%
benefit value	$\ell = \infty$	46.24%	40.37%	39.72%	34.89%	34.08%	30.44%	27.10%
	$\ell = 2$	42.91%	38.13%	36.62%	30.11%	31.76%	28.57%	26.02%
	difference	3.33%	2.25%	3.11%	4.78%	2.32%	1.87%	1.08%
contribution value	$\ell = \infty$	44.41%	37.06%	38.30%	34.77%	31.76%	28.55%	26.41%
	$\ell = 2$	43.19%	40.15%	37.50%	31.94%	29.80%	28.19%	25.90%
	difference	1.23%	-3.09%	0.80%	2.83%	1.96%	0.36%	0.51%
Banzhaf* value	$\ell = \infty$	45.12%	40.51%	38.89%	35.16%	31.22%	29.59%	27.22%
	$\ell = 2$	43.33%	37.57%	34.47%	30.42%	28.48%	27.23%	25.33%
	difference	1.78%	2.94%	4.42%	4.74%	2.74%	2.36%	1.89%

nine countries, using the tau value). Tables 2 and 3 summarize the distribution of incomplete simulation instances aggregated over different choices of solution concepts or scenarios. See Appendix B for a complete breakdown of the averages in both these tables. Here, we only observe from Tables 2 and 3 that the average number of incomplete instances is relatively low, both for equal country sizes (at most 2% in total) and varying country sizes (at most 0.75% in total).

Given that the average number of incomplete instances is relatively low, and moreover that the solutions found for incomplete simulation instances are all maximum cycle packings (which might not be weakly or strongly close), policy makers could therefore still decide to use them. For this reason, we decided to include the incomplete simulation instances in Figs. 5–8. We believe this was justified after doing some additional research. That is, we also constructed the same figures as Figs. 5 and 6 but without the incomplete simulation instances; see Appendix B. It turned out that the largest percentage points difference in Fig. 5 of the average total deviations is only 0.059% (with an average of 0.0024%). Hence, the quality of the “current-best” optimal solutions for the incomplete simulation instances are almost indistinguishably close to those for the complete simulation instances.

Number of Kidney Transplants and Cycle Length. When $\ell = \infty$ instead of $\ell = 2$, we expect more kidney transplants, because exchange cycles may now have any size. In Tables 4–5 and Fig. 9 we quantified this.

First, from Table 4 we see that for $n = 10$ and equal country sizes it is possible to achieve 46% more kidney transplants when $\ell = \infty$ instead of $\ell = 2$, and that this is irrespective of the chosen scenario or chosen solution concept, as the total number of kidney transplants are nearly identical, both for $\ell = 2$ and $\ell = \infty$. Table 5 shows the results for varying country sizes, for which we can draw the same conclusions. We refer to Tables 19 and 20 in Appendix D for the corresponding results for $n = 4, \dots, 9$, which show the same behaviour.

Second, Fig. 9 shows that setting $\ell = \infty$ may indeed lead to long cycles, with even more than 400 vertices. Fig. 9 shows in particular that these long cycles all occur in the first round. The reason is that we start the first round with roughly 500 vertices and introduce the remaining 1500 patient-donor pairs in the later rounds by a uniform distribution. Hence, the compatibility graphs in later rounds are all smaller.

Influence of Using Credits. We now discuss the power of our credit system. Does it effectively reduce deviations from the target allocation? Can it keep deviations on a consistently low level in the long-term? Theoretically it is possible that credits keep accumulating, which would make them ineffective (see Benedek et al., 2024b for a theoretical example of this behaviour when $\ell = 2$).

In Figs. 10 and 11 we show to what extent credits accumulate for equal and varying country sizes, respectively. If credits are not incorporated, we can still compute and track them as we did in these two figures. We note that over a period of 24 rounds, credits accumulate more and more (but at different rates) under all three scenarios *arbitrary*, *d1* and *lexmin*, even though especially under *d1* and *lexmin* we do find solutions that are relatively close to the target allocations (as we saw from Figs. 5 and 6). However, only when we also incorporate credits we not only find solutions that are close to the target solutions but that also ensure stability. We find that the Banzhaf* value under *lexmin + c* is also the best in maintaining consistently low levels of credits, much like in the case of $\ell = 2$ (Benedek et al., 2024b). Again, the main difference between $\ell = \infty$ and $\ell = 2$ is that the *arbitrary* scenario performs much worse for $\ell = 2$.

Deviations for Worst-off Countries. Finally, we consider the ratio of the average maximum relative deviation and the average total relative deviation as a measure for the proportion of the total country deviations being due to the worst-off country. We call this ratio the *relative ratio*. For stability reasons, low relative ratios are preferable. From Table 6 we conclude that for the most interesting scenario *lexmin + c* and equal country sizes, the relative ratio decreases as the number of countries increases (as expected). In this figure we also included the relative ratios for $\ell = 2$ (generated from the data used in Benedek et al., 2024b). We note that the relative ratio is slightly better for $\ell = 2$. This extends to varying countries as well; see Appendix C. In this appendix we also included the figures for the other scenarios as well. From these figures, we note that for *d1 + c* the relative ratio for $\ell = \infty$ is lower than for $\ell = 2$. The *arbitrary* scenario showed consistently lower deviations for $\ell = \infty$ than for $\ell = 2$. However, this is not the case for the relative ratio. For equal country sizes, $\ell = \infty$ leads to consistently higher relative ratios than $\ell = 2$ does, whereas for varying country set sizes the opposite holds. For the remaining scenarios (*lexmin* and *d1*), there is no clear distinction.

Table 7

Average CPU time for a single 24-round simulation instance, broken down into the different computational tasks. Here, the times for data preparation and graph building are average times taken over all scenarios and solution concepts, whereas the time for each solution concept is the average time taken over all scenarios. The total times for the scenarios are average times taken over all solutions concepts, where “total” refers to the total computation time (using up to 32 threads for each simulation instance), which includes computing the initial allocations, and only excludes the time for data preparation and graph building.

CPU time / n	4	5	6	7	8	9	10
data preparation	0.01	0.01	0.01	0.01	0.01	0.01	0.01
graph building	0.04	0.04	0.04	0.04	0.04	0.04	0.04
Shapley value	12.32	25.28	48.32	94.70	188.7	393	745.2
Nucleolus	10.75	22.33	44.41	86.83	165.5	342	707.8
Benefit value	7.87	8.41	9.64	11.31	12.68	14.45	15.81
Contribution value	7.39	8.80	10.35	12.44	14.15	16.68	17.00
Banzhaf value	10.56	21.73	42.06	89.00	174.75	343.06	689.62
Banzhaf* value	13.03	25.48	50.98	89.12	173.76	359.56	687.96
Tau value	17.76	29.94	56.70	99.66	199	358.1	796.2
total: arbitrary	11.63	20.35	35.35	65.49	123.81	257.33	530.78
total: d1	66.81	85.85	42.97	127.49	171.84	315.08	473.38
total: lexmin	305.04	688.20	1044.89	1929.32	2771.54	4912.33	6737.04
total: d1 + c	93.26	87.09	43.93	121.11	166.00	330.56	504.52
total: lexmin + c	470.29	1030.02	2010.87	2514.62	4032.30	5764.23	7990.36

4.7. Computation times

Table 7 summarizes the average computational time of solving a single (24-round) simulation instance. As expected, computing initial allocations using the two easy-to-compute solution concepts (benefit and contribution values) is inexpensive, especially compared to the other, more sophisticated but hard-to-compute solution concepts.

In Table 7 we also see that the computation time for the *arbitrary* scenario, which does not require initial allocations, is less time-consuming than for the other four scenarios. This is in line with Theorem 1, which gave us a polynomial time algorithm for computing an arbitrary optimal solution, whereas we needed to rely on using ILPs for computing optimal solutions that are weakly close (*d1* and *d1 + c*) or even strongly close (*lexmin* and *lexmin + c*). From Table 7, it can be clearly seen that the computation of the ILP series requires the majority of the work, especially for the *lexmin* and *lexmin + c* scenarios: these involve longer series of ILPs than *d1* and *d1 + c*, for which we need to solve only (ILP_{d1}).

On a side note, we observe from Table 7 that when *n* increases, the computation time increases roughly within the expected rate, apart from a few notable exceptions: Gurobi’s ILP solver handles, for some reason, (ILP_{d1}) far more easily for *n* = 6 than for *n* = 5 (or even *n* = 4). We have no explanation for this; it may well be due to the nature of Gurobi, in which we have no insights.

5. Conclusions

We introduced the class of partitioned permutation games and first proved a number of complexity results.

In particular, we showed that every partitioned permutation game has a nonempty core and that we can compute a core allocation of a partitioned permutation game in polynomial time. We also proved that for partitioned permutation games of fixed width *c*, the problem of deciding if an allocation is in the core is polynomial-time solvable if *c* = 1 but is already coNP-complete if *c* = 2. Moreover, we proved that in this setting even the problem of finding a weakly optimal solution for a given target allocation *x* is NP-hard, and we gave an exact randomized XP-time algorithm for solving this problem. Our theoretical results contrast some known results for partitioned ℓ -permutation games where ℓ is some constant, as surveyed in Table 1. As an open theoretical problem we ask:

Open Problem 1. Determine the complexity of computing the nucleolus for permutation games and for partitioned permutation games.

Our new results guided our simulations for IKEPs, where we could include up to ten countries, with exchange bound $\ell = \infty$. Our simulations showed the clear benefits of using a credit system with strongly close optimal solutions, namely an improvement in balance up to 56%. The exact improvement depends on the choice of solution concept, with the Banzhaf* value yielding the best results, namely on average, a deviation of at most 0.90%. Our simulations showed a significant improvement (46% on average) in the total number of kidney transplants if cycles of any length are allowed compared to the case where $\ell = 2$ (Benedek et al., 2024b). In our simulations, we first let all countries be of the same size. We then examined, in the same way as was done in Benedek et al. (2024b), whether our conclusions would change for varying country sizes. Just as in Benedek et al. (2024b), this turned out not to be the case.

For future research, it would be interesting to determine to what extent our theoretical and experimental results change if we allow non-directed (altruistic) donors or multiple donors registering for one patient. We did not consider such types of donors for this paper, as our goal was to compare our results with the previously known results for $\ell = 2$, where non-directed donors and multiple donors for one patient were also excluded. Moreover, we recall that matching games and permutation games may be defined on edge-weighted graphs, in which each edge $e \in A$ has some weight $w(e)$ reflecting the expected utility of the corresponding kidney transplant. It would be interesting to do simulations in the presence of edge weights. Furthermore, we recall that more complex hierarchical optimization criteria than only optimizing the number of transplants are used in European kidney exchange (Biró et al., 2019, 2021). We leave conducting simulations with such criteria for future research as well.

Finally, for future research we will consider the more realistic exchange bounds $\ell \in \{3, 4, 5\}$. We note that the simulations done in Biró et al. (2020), Klimentova et al. (2021) were only for $\ell = 3$; a more limited number of solution concepts; for IKEPs with up to four countries; and only for scenarios that use weakly close optimal solutions. They also used different data sets. For our follow-up study for $\ell \in \{3, 4, 5\}$, we must now also overcome, as in Biró et al. (2020), Klimentova et al. (2021), the additional computational obstacle of not being able to compute an optimal solution for a compatibility graph in a KEP round and the values $v(S)$ of the associated permutation game in polynomial time (see Theorem 1). Current techniques for $\ell \in \{3, 4, 5\}$ therefore involve, besides ILPs based on the edge-formulation, ILPs based on the cycle-formulation, with a variable for each cycle of length at most ℓ (see, for example, Constantino et al., 2013; Delorme et al., 2024; Dickerson et al., 2016). Hence, computing a single value $v(S)$ will become significantly more expensive, and even more so for increasing ℓ . For expensive solution concepts, such as the Shapley or nucleolus, we must compute an exponential number of values $v(S)$. Without any new methods, we expect it will not be possible to do this for simulations up to the same number of countries (ten) as we did for $\ell = \infty$.

CRedit authorship contribution statement

Márton Benedek: Writing – original draft, Software, Methodology, Investigation, Conceptualization; **Péter Biró:** Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization; **Gergely Csáji:** Writing – original draft, Visualization, Methodology, Formal analysis, Conceptualization; **Matthew Johnson:** Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization; **Daniël Paulusma:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization; **Xin Ye:** Writing – review & editing, Writing – original draft, Software, Methodology, Investigation, Formal analysis, Conceptualization.

Declaration of interests

We declare no conflict of interest. All of our the funding sources are mentioned in both in the paper and in the online declaration.

Acknowledgments

Benedek was supported by the National Research, Development and Innovation Office of Hungary (OTKA Grant No. K138945); Biró by the Hungarian Scientific Research Fund (OTKA, Grant No. K143858) and the Hungarian Academy of Sciences (Momentum Grant No. LP2021-2); Csáji by the Hungarian Scientific Research Fund, OTKA, Grant No. K143858, and by the Doctoral Student Scholarship Program (number C2258525) of the Cooperative Doctoral Program of the Ministry of Innovation and Technology financed by the National Research, Development, and Innovation Fund; and Paulusma was supported by the Leverhulme Trust (Grant RF-2022-607) and EPSRC (Grant EP/X01357X/1). This work has made use of the Hamilton HPC Service of Durham University. In particular, we thank Rob Powell for all his help with the initial set up of our simulations.

We also thank the three anonymous reviewers of our paper for all their useful comments for improving the readability of our paper.

Supplementary material

Supplementary material associated with this article can be found in the online version at [10.1016/j.ejor.2025.12.046](https://doi.org/10.1016/j.ejor.2025.12.046).

References

- Abraham, D. J., Blum, A., & Sandholm, T. (2007). Clearing algorithms for barter exchange markets: enabling nationwide kidney exchanges. *Proceedings EC 2007*, (pp. 295–304).
- Agarwal, N., Ashlagi, I., Azevedo, E., Featherstone, C. R., & Karaduman, Ö. (2019). Market failure in kidney exchange. *American Economic Review*, 109, 4026–4070.
- Andersson, T., Cseh, Á., Ehlers, L., & Erlanson, A. (2021). Organizing time exchanges: Lessons from matching markets. *American Economic Journal: Microeconomics*, 13, 338–373.
- Ashlagi, I., Fischer, F., Kash, I. A., & Procaccia, A. D. (2015). Mix and match: A strategyproof mechanism for multi-hospital kidney exchange. *Games and Economic Behavior*, 91, 284–296.
- Ashlagi, I., & Roth, A. E. (2012). New challenges in multihospital kidney exchange. *American Economic Review*, 102, 354–359.
- Ashlagi, I., & Roth, A. E. (2014). Free riding and participation in large scale, multi-hospital kidney exchange. *Theoretical Economics*, 9, 817–863.
- Aziz, H., Cseh, Á., Dickerson, J. P., & McElfresh, D. C. (2021). Optimal kidney exchange with immunosuppressants. *Proceedings AAAI 2021*, (pp. 21–29).
- Banzhaf, J. F., III (1964). Weighted voting doesn't work: A mathematical analysis. *Rutgers Law Review*, 19, 317–343.
- Barkel, M., Colley, R., Delorme, M., Manlove, D., & Pettersson, W. (2025). Operational research approaches and mathematical models for kidney exchange: A literature survey and empirical evaluation. *European Journal of Operational Research*, to appear.
- Benedek, M. (2021). International kidney exchange scheme. https://github.com/blrzsvrzs/int_kidney_exchange.
- Benedek, M., Biró, P., Csáji, G., Johnson, M., Paulusma, D., & Ye, X. (2024a). Computing balanced solutions for large international kidney exchange schemes when cycle length is unbounded. *Proceedings AAMAS 2024*, (pp. 2153–2155).
- Benedek, M., Biró, P., Johnson, M., Paulusma, D., & Ye, X. (2023). The complexity of matching games: A survey. *Journal of Artificial Intelligence Research*, 77, 459–485.
- Benedek, M., Biró, P., Kern, W., Pálvolgyi, D., & Paulusma, D. (2025). Partitioned matching games for international kidney exchange. *Mathematical Programming*, 214, 723–758.
- Benedek, M., Biró, P., Kern, W., & Paulusma, D. (2022). Computing balanced solutions for large international kidney exchange schemes. *Proceedings AAMAS 2022*, (pp. 82–90).
- Benedek, M., Biró, P., Paulusma, D., & Ye, X. (2024b). Computing balanced solutions for large international kidney exchange schemes. *Autonomous Agents and Multi-Agent Systems*, 38, 18.
- Benedek, M., Fliege, J., & Nguyen, T. (2021). Finding and verifying the nucleolus of cooperative games. *Mathematical Programming*, 190, 135–170.
- Biró, P., Gyetvai, M., Klimentova, X., Pedrosa, J. P., Pettersson, W., & Viana, A. (2020). Compensation scheme with Shapley value for multi-country kidney exchange programmes. *Proceedings ECMS 2020*, (pp. 129–136).
- Biró, P., Haase-Kromwijk, B., & Andersson, T., et al. (2019). Building kidney exchange programmes in Europe – an overview of exchange practice and activities. *Transplantation*, 103, 1514–1522.
- Biró, P., Klijn, F., Klimentova, X., & Viana, A. (2024). Shapley-scarf housing markets: Respecting improvement, integer programming, and kidney exchange. *Mathematics of Operations Research*, 49, 1938–1972.
- Biró, P., Van de Klundert, J., Manlove, D., Pettersson, W., Andersson, T., Burnapp, L., Chromy, P., Delgado, P., Dworzczak, P., Haase, B. et al. (2021). Modelling and optimisation in European kidney exchange programmes. *European Journal of Operational Research*, 291 (2), 447–456.
- Biró, P., Manlove, D. F., & Rizzi, R. (2009). Maximum weight cycle packing in directed graphs, with application to kidney exchange programs. *Discrete Mathematics, Algorithms and Applications*, 1, 499–518.
- Blum, A., Caragiannis, I., Haghtalab, N., Procaccia, A. D., Procaccia, E. B., & Vaish, R. (2017). Opting into optimal matchings. *Proceedings SODA 2017*, (pp. 2351–2363).
- Böhmig, G. A., Fronck, J., Slavcev, A., Fischer, G. F., Berlakovich, G., & Viklicky, O. (2017). Czech-Austrian kidney paired donation: First European cross-border living donor kidney exchange. *Transplant International*, 30, 638–639.
- Carvalho, M., & Lodi, A. (2023). A theoretical and computational equilibria analysis of a multi-player kidney exchange program. *European Journal of Operational Research*, 305, 373–385.
- Carvalho, M., Lodi, A., Pedrosa, J. a. P., & Viana, A. (2017). Nash equilibria in the two-player kidney exchange game. *Mathematical Programming*, 161, 389–417.
- Constantino, M., Klimentova, X., Viana, A., & Rais, A. (2013). New insights on integer-programming models for the kidney exchange problem. *European Journal of Operational Research*, 231, 57–68.
- Curiel, I. J., & Tijs, S. H. (1986). Assignment games and permutation games. *Methods of Operations Research*, 54, 323–334.
- Delorme, M., García, S., Gondzio, J., Kalcsics, J., Manlove, D., & Pettersson, W. (2024). New algorithms for hierarchical optimisation in kidney exchange programmes. *Operations Research*, 72, 1654–1673.
- Delorme, M., García, S., Gondzio, J., Kalcsics, J., Manlove, D. F., Pettersson, W., & Trimble, J. (2022). Improved instance generation for kidney exchange programmes. *Computers & Operations Research*, 141, 105707.
- Dezso, B., Jüttner, A., & Kovács, P. (2010). LEMON - An open source C++ graph template library. *Proceedings WGT@ETAPS, electronic notes in theoretical computer science*, 264, 23–45.
- Dickerson, J. P., Manlove, D. F., Plaut, B., Sandholm, T., & Trimble, J. (2016). Position-indexed formulations for kidney exchange. *Proceedings of EC 2016*, (pp. 25–42).
- D'Errico, M., & Roukny, T. (2021). Compressing over-the-counter markets. *Operations Research*, 69 (6), 1660–1679.
- Edmonds, J. (1965). Paths, trees and flowers. *Canadian Journal of Mathematics*, 17, 449–467.
- Gurjar, R., Korwar, A., Messner, J., Straub, S., & Thierauf, T. (2012). Planarizing gadgets for perfect matching do not exist. *Proceedings MFCS 2012, lecture notes in computer science*, 7464, 478–490.
- Hajaj, C., Dickerson, J. P., Hassidim, A., Sandholm, T., & Sarne, D. (2015). Strategy-proof and efficient kidney exchange using a credit mechanism. *Proceedings AAAI 2015*, (pp. 921–928).
- Hein, J., Jiang, T., Wang, L., & Zhang, K. (1996). On the complexity of comparing evolutionary trees. *Discrete Applied Mathematics*, 71, 153–169.
- Hickey, G., Dehne, F., Rau-Chaplin, A., & Blouin, C. (2008). SPR distance computation for unrooted trees. *Evolutionary Bioinformatics*, 4, EBO–S419.
- Klimentova, X., Viana, A., Pedrosa, J. a. P., & Santos, N. (2021). Fairness models for multi-agent kidney exchange programmes. *Omega*, 102, 102333.
- Manjunath, V., & Westkamp, A. (2021). Strategy-proof exchange under trichotomous preferences. *Journal of Economic Theory*, 193, 105197.
- Maschler, M., Peleg, B., & Shapley, L. S. (1979). Geometric properties of the kernel, nucleolus, and related solution concepts. *Mathematics of Operations Research*, 4, 303–338.
- Papadimitriou, C. H., & Yannakakis, M. (1982). The complexity of restricted spanning tree problems. *Journal of the ACM*, 29, 285–309.
- Peters, H. (2008). Game theory: A Multi-leveled approach. Springer.
- Pettersson, W., & Trimble, J. (2021). Kidney matching tools data set generator. <https://wpettersson.github.io/kidney-webapp/#/generator>.
- Roth, A. E., Sönmez, T., & Ünver, M. U. (2004). Kidney exchange. *Quarterly Journal of Economics*, 119, 457–488.
- Roth, A. E., Sönmez, T., & Ünver, M. U. (2005). Pairwise kidney exchange. *Journal of Economic Theory*, 125, 151–188.
- Schmeidler, D. (1969). The nucleolus of a characteristic function game. *SIAM Journal on Applied Mathematics*, 17, 1163–1170.
- Shapley, L. S. (1953). A value for n -person games. *Annals of Mathematical Studies*, 28, 307–317.
- Smeulders, B., Blom, D., & Spieksma, F. C. R. (2024). Identifying optimal strategies in kidney exchange games is σ_1^c -complete. *Mathematical Programming*, 203, 347–368.
- Sun, Z., Todo, T., & Walsh, T. (2021). Fair pairwise exchange among groups. *Proceedings IJCAI 2021*, (pp. 419–425).
- Tijs, S., Parthasarathy, T., Potters, J. A. M., & Prasad, V. R. (1984). Permutation games: Another class of totally balanced games. *OR Spektrum*, 6, 119–123.
- Tijs, S. H. (1981). Bounds for the core of a game and the t -value. In O. Moeschlin & D. Pallaschke (Eds.), *Game theory and mathematical economics*. North-Holland, Amsterdam, (pp. 123–132).
- Toulis, P., & Parkes, D. C. (2015). Design and analysis of multi-hospital kidney exchange mechanisms using random graphs. *Games and Economic Behavior*, 91, 360–382.
- Valentín, M. O., García, M., Costa, A. N., Bolotinha, C., Guirado, L., Vistolli, F., Breda, A., Fiaschetti, P., & Dominguez-Gil, B. (2019). International cooperation for kidney exchange success. *Transplantation*, 103, 180–181.
- Ye, X. (2023). International kidney exchange data. https://github.com/Arya1531/international_kidney_exchange_program/tree/main.
- Young, H. P. (1985). Monotonic solutions of cooperative games. *International Journal of Game Theory*, 14, 65–72.