Computing Balanced Solutions for International Kidney Exchange Schemes

XIN YE

Computing Balanced Solutions for International Kidney Exchange Schemes

Xin Ye

A Thesis presented for the degree of
Doctor of Philosophy

Durham University

Department of Computer Science
Durham University
United Kingdom
November 21, 2025

Abstract

In kidney exchange programmes (KEPs) patients may swap their incompatible donors leading to cycles of kidney transplants. Nowadays, countries have begun merging their national KEPs into international KEPs (IKEPs) to increase the overall number of transplants (social welfare). To ensure the longer-term stability of IKEPs, a credit-based system is introduced. In each round, every country is prescribed a "fair" initial allocation of kidney transplants. The initial allocation, which is obtained by using solution concepts from cooperative game theory, is adjusted by incorporating credits from the previous round, yielding a target allocation for each country. The overall goal is to find, in each round, an optimal solution, maximizing the total number of kidney transplants that approximates the target allocation vector as close as possible.

However, the maximum exchange cycle length, denoted by ℓ, is often restricted by logistical, practical and legal considerations. It is known that the complexity of computing an optimal solution is polynomial when $\ell = 2$ or $\ell = \infty$; otherwise, it is NP-hard [1]. It is polynomial to find an optimal solution that lexicographically minimizes the country deviations from the target allocation if $\ell = 2$ is permitted [2]. However, in practice, kidney transplants along longer cycles may be performed. To an extreme, namely, $\ell = \infty$, we show that the lexicographical minimization is only polynomial-time solvable for $\ell = \infty$ under additional conditions (assuming $\mathsf{P} \neq \mathsf{NP}$). Nevertheless, the fact that the optimal solutions themselves can be computed in polynomial time if $\ell = \infty$ still enables us to perform an experimental study for a large number of countries. Furthermore, $\ell = 3$ is permitted in some European countries, such as UK and Netherlands. Even though computing optimal solutions for $\ell = 3$ is NP-hard, we perform simulations by formulating integer linear programs and utilizing ILP solvers. As a whole, we perform large-scale and consistent simulation studies for $\ell = 2$, $\ell = \infty$ and $\ell = 3$ and examine how the transitions from $\ell = 2$ to $\ell = \infty$, and from $\ell = 2$ to $\ell = 3$, affect both stability and total social welfare.

Declaration

The work in this thesis is based on research carried out at the Department of Computer Science, Durham University, United Kingdom. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Acknowledgements

My deepest and sincerest gratitude for my PhD journey goes to my supervisors, Daniël Paulusma and Matthew Johnson. Their exceptional research taste, rigorous approach to science, and contagious passion for research have shaped my development in countless ways. I feel incredibly fortunate for their supervision during my PhD journey, which trained me not only how to become an independent researcher but also how to remain curious, brave, joyful and resilient, both within academia and beyond.

I would like to thank George Mertzios and Bart Smeulders for being my internal and external examiners, respectively. I greatly enjoyed the insightful discussions during my PhD defense and deeply value the constructive comments you provided on my PhD thesis.

I would also like to express my heartfelt appreciation to Neil Olver and László Végh. Neil Olver welcomed me into the world of research in algorithmic game theory at the very beginning of my research journey, even before my PhD, with his insightful and creative guidance. László Végh, my mentor during my master's studies, introduced me to Neil and recommended the PhD position to me. Their generous support played a pivotal role in shaping the beginning of my PhD path.

I would like to sincerely thank Márton Benedek for his support through my PhD studies, especially for guiding me through the coding of our extensive simulations at the very beginning of my PhD research. I am also grateful to my other collaborators, Péter Biró, Márton Benedek, Gergerly Csáji, David Manlove, William Pettersson, Rachael Colley and of course, my two amazing supervisors, Daniël Paulusma and Matthew Johnson. I have enjoyed every discussion with them and our collaborations have led to numerous research contributions and significantly accelerated my growth from a junior PhD student to an independent researcher.

Additionally, I would like to thank my office mates in MCS2010, as well as the other academic and administrative staff in the department, whose support, laughter, and companionship brought so much joy to my four years at Durham. I am also grateful to my other friends, both those nearby and those who stay close online, despite being thousands of miles away and separated by time zones of 5, 7, 8 hours, or even more.

Last but certainly not least, my eternal appreciation always goes to my family, especially my parents, Zhibin Ye and Beiwen Lin! Your unwavering support and unconditional love have been my greatest source of strength and guidance throughout life.

Dedication

To my parents.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

For kidney patients, transplantation is the most effective treatment resulting in a longer life expectancy and better long-term quality of life compared to kidney dialysis. Transplanted kidneys could either come from living donors or deceased donors, but transplantation from living donors is generally more effective, especially in the light of patient and graft survival. However, a kidney transplantation might not be possible due to blood-type or tissue-type incompatibilities between a patient and a willing donor. Depending on the country in Europe, 40% or more of patients are incompatible with their willing donors [12]. To circumvent medical incompatibilities, *Kidney Exchange Programmes* (KEPs) have been established in many countries in the last 30 years [12] with the main goal of maximizing the number of transplants. A KEP places all patient-donor pairs in one pool such that donors can be swapped in a cyclic manner; such cycles are illustrated in Figure 1.1. In the survey from 2016 [12], UK operated a total number of 3328 kidney transplants in the KEP, while 28000 kidney patients relied on dialysis and 5215 kidney patients were awaiting for compatible donors. However, establishing national KEPs normally requires ethical viewpoints, law framework, clinical technology and operational practices and many other aspects to be overcome. As a result, KEPs are not allowed in some countries, such as

Finland and Iceland. In countries where KEPs are permitted, implementations and criteria vary but simultaneous transplants within cycles are generally required (see details in Section 1.1).

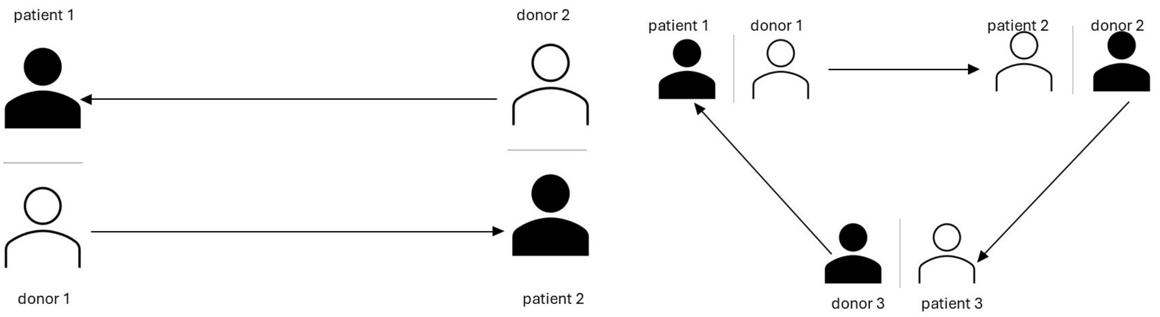

Figure 1.1: Examples of cycles: In the left figure, there are two patient-donor pairs: donor 1 is associated with patient 1 and donor 2 is associated with patient 2. Donor 1 gives a kidney to patient 2, and in return, donor 2 gives a kidney to patient 1, forming a two-way exchange—a length-2 cycle. In the right figure, there are three patient-donor pairs, forming a three way exchange—a length-3 cycle.

1.1 Kidney Exchange

A KEP is often modeled by a *compatibility graph* $G = (V, A)$ 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 $C = u_1u_2 \ldots u_ku_1$, 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 $i \in \{1, \ldots, k+1\}$, with $u_{k+1} := u_1$. This is a *k-way exchange* using the *exchange cycle* C with the cycle length k. The two-way exchange and the three-way exchange are illustrated in the left and right subgraphs of Figure 1.1, respectively.

For KEPs, the simultaneous transplantation across the exchange cycle is generally required in case the patient receives a kidney but the corresponding donor reneges to donate the kidney, causing the whole exchange cycle unable to proceed. Mainly due to the need for simultaneous surgeries in the exchange cycle, longer cycles can pose logistical challenges for kidneys. In addition, longer cycles are more vulnerable, as any donors or patients in the exchange cycle being sick or unfortunately passing away, could easily cause the exchange cycle to fail. Therefore, KEPs impose a bound ℓ, called the *exchange bound*, on the maximum *length* (number of edges) of an exchange cycle, although allowing longer cycles may result in more transplants (see Figure 1.3). The exchange bound varies between countries, typically ranging from 2 to 7. For example, ℓ is 2 for France and Italy, 3 for Austria, Belgium, Poland, Portugal, Spain and UK, 4 for Netherlands and 7 for Czech Republic.

A solution for a KEP is an *ℓ-cycle* packing in the corresponding compatibility graph G. Formally, an ℓ-cycle packing of G is a set $\mathcal{C}$ of directed cycles, each of length at most ℓ, that are pairwise vertex-disjoint; if $\ell = \infty$, we also say that $\mathcal{C}$ is a *cycle packing*. The *size* of $\mathcal{C}$ is the sum of the lengths of the cycles of $\mathcal{C}$. Recall that the primary goal of KEPs is to help as many patients as possible. Hence, to maximize the number of transplants for a given G, we seek an *optimal* solution, that is, a *maximum* ℓ-cycle packing of G, i.e., one that has maximum size. A KEP might have multiple optimal solutions; see Figure 1.2. The longer ℓ might also lead to a greater maximum ℓ-cycle packing size; see Figure 1.3. Increasing ℓ from 2 to 3 results in 3 more transplants and increasing it from 3 to ∞ yields 1 more transplant.

Figure 1.2: Examples illustrating multiple optimal solutions and the unique optimal solution if $\ell = 2$ is only allowed. In the left subgraph, there are three optimal solutions for the KEP: $\{aba\}$, $\{bcb\}$ and $\{aca\}$. Each of the three optimal solution results in the same number of transplants, namely two. While in the right subgraph, the optimal solution is unique, as indicated by four thick arcs representing four transplants.

KEPs operate in rounds, typically every three months in countries like UK and Netherlands, every four months in Spain and monthly in Poland [12]. After round r, some patient-donors might leave the KEP because corresponding patients have received a kidney or passed away or any other reasons, while other new patient-donor pairs may have joined for the next round $r + 1$. This results in a new compatibility graph G^{r+1} for round $r + 1$.

The main computational issue for KEPs is how to find an optimal solution (maximum ℓ-cycle packing) in each round. If $\ell = 2$, we can transform, by keeping the "double" arcs, a compatibility graph G into an undirected graph $D = (V, E)$ as

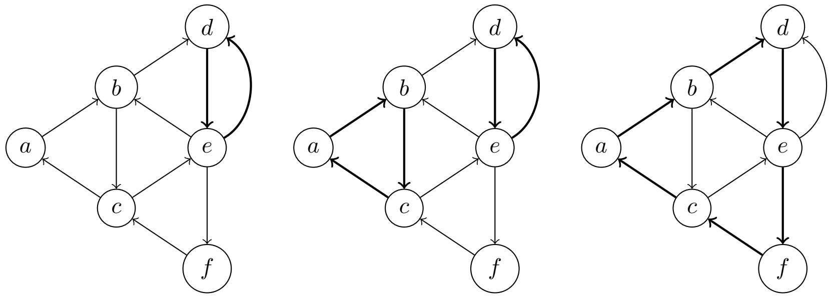

Figure 1.3: A KEP defined on a compatibility graph $G = (V, A)$. Note that we have 6 patient-donor pairs, represented by $V = \{a, b, c, d, e, f\}$, as indicated by 6 vertices in the graph, with compatibilities shown by arcs. In the leftmost subgraph, when the exchange bound $\ell = 2$ is permitted, a maximum 2-cycle packing $\{ded\}$ is selected as indicated by thick arcs, leading to 2 transplants. When ℓ is extended to 3, a maximum 3-cycle packing $\{ded, abca\}$ is selected as indicated by thick arcs in the middle subgraph along with 5 transplants. However, if $\ell = \infty$, a maximum cycle packing $\{abdefca\}$ is selected as indicated by thick arcs in the rightmost subgraph, resulting in 6 transplants, with an increase of 4 and 1 transplants compared to $\ell = 2$ and $\ell = 3$ respectively.

follows (see Figure 1.4). For every $u, v \in V$, we have $uv \in E$ and assign it a weight of 2 if and only if $(u, v) \in A$ and $(v, u) \in A$. It then remains to compute a maximum matching in D, which can be done in polynomial time. We now formally define a matching. Let $M \subseteq E$ be some subset of edges. For a positive edge weighting w of G, the *weight* of M is defined as $w(M) = \sum_{e \in M} w(e)$. For a positive vertex capacity function b (see Chapter 2), we say that M is a *b-matching* of $G = (V, E)$ if each $i \in V$ is incident to at most $b(i)$ edges in M. If $b \equiv 1$, then M is a *matching*.

If we set $\ell = \infty$, a well-known trick works (see e.g. [1]). 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 is NP-hard, as shown by Abraham, Blum and Sandholm [1].

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

As it is NP-hard to find an optimal solution for any constant $\ell \geq 3$, normally in KEPs, we formulate the optimization problem into an integer linear program (ILP) and use ILP solvers to solve it in practice [13]. In Chapter 6, we formulate our problem for $\ell = 3$ as ILPs and use the advanced solver Gurobi to solve them.

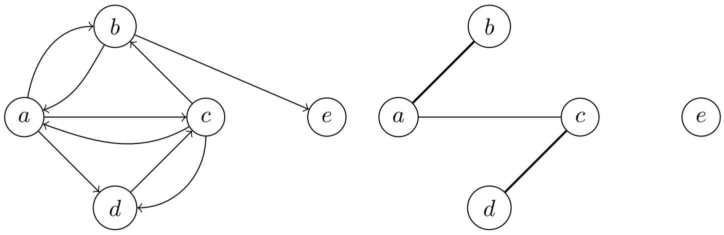

Figure 1.4: A directed compatibility graph (left) is transformed into an undirected graph (right) when $\ell = 2$. The optimal solution corrresponds to the maximum matching $\{ab, cd\}$ as indicated by thick edges in the right subgraph. Note that in the right subgraph, each edge represents 2 compatibilities between the connected vertices. And thus if the maximum matching $\{ab, cd\}$ is selected, it results in 4 kidney transplants for patient-donor pairs $\{a, b, c, d\}$.

In our KEP setting, we only allow patient-donor pairs to join the pool and swap incompatible donors, with the primary goal of maximizing the number of transplants subject to an exchange bound ℓ, exactly as the practical implementation in France [12]. A patient-donor pair only consists of a patient and an incompatible donor. However, in practice, a patient with multiple donors is allowed to join the KEP for the most of European countries, but some countries, such as Belgium and France, do not permit this [14]. In addition, a KEP might also include *non-directed donors*, sometimes referred to as *altruistic donors*, such as UK and Netherlands [12], who don't bring any kidney patients when joining the KEP, instead, are willing to donate one of their kidneys to a kidney patient in the KEP pool. Non-directed donors can initiate an *exchange chain* along with other patient-donor pairs. The chain starts with a non-directed donor, donating one of their kidneys to patient p_1 of patient-donor pair (p_1, d_1). Then the corresponding donor d_1 donates the kidney to the patient p_2 of the next patient-donor pair (p_2, d_2) and continues sequentially until the last patient-donor pair (p_n, d_n). And donor d_n donates the kidney to the Deceased Donor Waiting List (DDWL) [15], where kidney patients awaiting for deceased donor kidney transplantation are prioritized according to their length of their waiting time [16]. It is worth noting that in contrast to the simultaneous transplants along exchange cycles, transplants in an exchange chain do not need to be implemented simultaneously. This is mainly because if any donor of patient-donor pairs in the chain reneges, the head subchain starting with the non-directed donor to the reneging patient-donor pair can still proceed whereas patient-donor pairs on the remaining tail chain neither donate nor receive a kidney. Although exchange chains are implemented in some countries, they fall outside the scope of this thesis. In our setting, we only consider living and incompatible patient-donor pairs in the KEP. In other words, we only consider exchange cycles.

1.2 International Kidney Exchange

As merging pools of national KEPs leads to a greater number of transplants (see Figure 1.5), a growing number of countries today work together with the aim of forming an *international* KEP (IKEP), e.g. Austria and the Czech Republic [17]; Denmark, Norway and Sweden; and Italy, Portugal and Spain [18]. 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* and do not receive improved transplant outcomes, 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 1.1. Let G be the compatibility graph from Figure 1.5. Recall that there could be multiple optimal solutions. In this example, there are two optimal solutions, $\{abdeca\}$ and $\{abefca\}$, when $\ell = 5$. Either of two optimal solutions yields the same maximum number of transplants, namely five kidney transplants. In other words, both optimal solutions guarantee the maximum social welfare for the entire international pool. Note that without cooperation only country 2 can receive two kidney transplants while neither of country 1 nor country 2 receives any transplants in this example. Establishing an IKEP of three countries results in three additional kidney transplants. However, the total number of transplants for each country in the IKEP is highly dependent on the chosen optimal solution. This raises a central question: how does the social welfare to participating countries vary depending on which optimal solution is chosen? We know that patient-donor pairs a, b, c belong to country 1, whereas patient-donor pairs d, e belong to country 2, and patient-donor pair f belongs to country 3. If the optimal solution $\{abdeca\}$ is selected, countries

1, 2 and 3 receive three, two and zero kidney transplants, respectively. Compared to national KEPs, country 1 benefits by receiving three more transplants, while neither of country 2 and 3 loses any transplants. If instead, the alternative optimal solution $\{abefca\}$ is selected, country 1 receives three transplants, and both of country 2 and 3 receive one transplant respectively. In this case, country 2 receives one less transplant while country 1 earns three more transplants and country 3 earns one more transplant compared to national KEPs. Country 2 is losing transplants after joining the IKEP, and might have the motivation to withdraw the IKEP if the unfair outcome continues to occur in the long term.

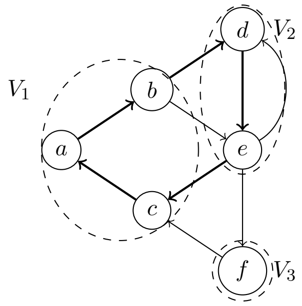

Figure 1.5: An IKEP of three countries defined on a graph $G = (V, A)$ when $\ell = 5$, where $V = V_1 \cup V_2 \cup V_3 = \{a, b, c, d, e, f\}$, representing a total of six patient-donor pairs. If three countries run national KEPs independently, only country 2 could yield two transplants. However, when three countries merge into an IKEP, the optimal solution is either maximum 5-cycle packing $\{abdeca\}$ or $\{abefca\}$. In the IKEP, the number of transplants increases to five, with three more transplants compared to KEPs.

As we see in Example 1.1, it is crucial to ensure both the maximum social welfare and individual rationality for each participating country, that is, no individual country is worse off when the international optimal solution is selected. Recall that there may be multiple optimal solutions for a given round of an IKEP. And thus it is essential to find a balanced solution among the set of optimal solutions that distributes international social welfare fairly. In particular, as an increasing number of countries join the IKEP, the number of optimal solutions may grow exponentially. Addressing IKEP stability is the central motivation behind the thesis.

1.2.1 Credit Systems

To ensure IKEP stability, we now turn to the model introduced by Klimentova et al. [10], which is a *credit-based* system. Formally, for round $r \geq 1$, let G^r be the compatibility graph with associated game (N^r, v^r) (see Chapter 2); let y^r be the initial allocation that distributes the total number of possible kidney transplants in round r amongst the countries in N^r. To obtain a fractional "fair" allocation y^r we use some solution concept S (see Chapter 2 for details on solution concepts and other relevant notions in Cooperative Game Theory); and let $c^r : N^r \to \mathbb{R}$ be a *credit function*, which satisfies $\sum_{p \in N^r} c_p^r = 0$; 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 $\mathcal{C}$ of G^r as optimal solution for round r (out of possibly exponentially many optimal solutions). Let $s_p(\mathcal{C})$ be the actual allocation for country p, specifically the total number of transplants for country p. For $p \in N$, we set $c_p^{r+1} := x_p^r - s_p^r(\mathcal{C})$ 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 $\mathcal{S}$ for the associated game (N^{r+1}, 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; see Example 1.2. Note that for every country $p \in N$ and round $h \geq 2$, it holds that

$$c_p^h = \sum_{t=1}^{h-1} (y_p^t - s_p(\mathcal{C}^t)),$$

so credits are in fact the accumulation of the deviations from the initial allocations. Hence, credits for a country can build up over time, irrespectively of our choice of initial allocations. Later in this section, we will give an explicit example where this happens. However, such situations did not occur in any of our simulations where we used the credit function (see Chapters 4-6).

Apart from explaining the target allocation, we must also determine how to choose in each round a maximum ℓ-cycle packing $\mathcal{C}$ (optimal solution) of the corresponding compatibility graph G to ensure the stability in the long term. We will choose $\mathcal{C}$, such that the vector $s(\mathcal{C})$, with number of kidney transplants entries $s_p(\mathcal{C})$, 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 solutions that are close to the target allocations, yielding consistently low (ideally zero) credit values.

To explain our distance measures, let $|x_p - s_p(\mathcal{C})|$ be the *deviation* of country p from its target x_p if $\mathcal{C}$ is chosen out of all optimal solutions. We order the deviations $|x_p - s_p(\mathcal{C})|$ non-increasingly as a vector $d(\mathcal{C}) = (|x_{p_1} - s_{p_1}(\mathcal{C})|, \ldots, |x_{p_n} - s_{p_n}(\mathcal{C})|)$. We say $\mathcal{C}$ is *strongly close* to x if $d(\mathcal{C})$ is lexicographically minimal over all optimal solutions. If we only minimize $d_1(\mathcal{C}) = \max_{p \in N}\{|x_p - s_p(\mathcal{C})|\}$ over all optimal solutions, we obtain a *weakly close* optimal solution. If an optimal solution is strongly close, it is weakly close, but the reverse might not be true.

Example 1.2. In Fig. 1.6, compatibility graphs for two rounds of an IKEP consisting of three countries are displayed when $\ell = 2$ is only allowed. First assume that we use the Shapley value for the initial allocations (its computation is explained in detail in Chapter 2). Let patient-donor pair a belong to country 1, patient-donor pairs b, c belong to country 2 and patient-donor pair d belong to country 3 in round 1. Note that $\mathcal{C}^1 = \{aba, dcd\}$ is the unique optimal solution in round 1. So, in round 1, we need to use $\mathcal{C}^1$. Recall that $c^1 = (0, 0, 0)$. Then $x^1 = y^1 = (\frac{2}{3}, \frac{8}{3}, \frac{2}{3})$. Moreover, $s(\mathcal{C}^1) = (1, 2, 1)$ and $c^2 = y^1 - s(C^1) = (-\frac{1}{3}, \frac{2}{3}, -\frac{1}{3})$. Hence, after round 1, all patient-donor pairs $\{a, b, c, d\}$ have been helped and leave the IKEP. Let patient-donor pair e from country 1, patient-donor pair f from country 2 and patient-donor pair g from country 3 join the IKEP in round 2. Note that we have two optimal solutions in round 2, $\mathcal{C}^2 = \{ege\}$ and $\mathcal{C}^2_* = \{efe\}$. So, in round 2, we must choose either $\mathcal{C}^2$ or $\mathcal{C}^2_*$, and this choice will be determined by which optimal solution will be closer to the target allocation x^2. Note that $y^2 = (\frac{4}{3}, \frac{1}{3}, \frac{1}{3})$. Hence, $x^2 = y^2 + c^2 = (1, 1, 0)$ and we must choose $\mathcal{C}^2_* = \{efe\}$, which has $s(\mathcal{C}^2_*) = (1, 1, 0)$. Consequently, $c^3 = x^2 - s(\mathcal{C}^2_*) = (0, 0, 0)$. Table 1.1 summarizes an example of computing credits corresponding to Figure 1.6.

Using, for example, the nucleolus (see Chapter 2) for the initial allocations yields the same for round 1 (as the optimal solution for round 1 is unique). However, in round 2, $y^2 = (2, 0, 0)$. Hence, $x^2 = y^2 + c^2 = (\frac{5}{3}, \frac{2}{3}, -\frac{1}{3})$ meaning that again we must pick $\mathcal{C}^2_*$. But now, $c^3 = x^2 - s(\mathcal{C}^2_*) = (\frac{2}{3}, -\frac{1}{3}, -\frac{1}{3})$. This means that in

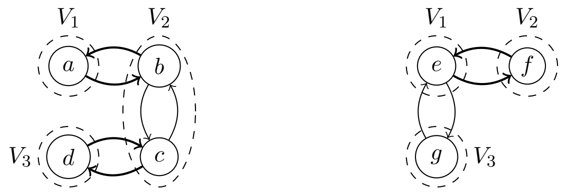

Figure 1.6: An example of the first two rounds of an IKEP with three countries when $\ell = 2$ and the Shapley value is selected. Round 1 is displayed on the left, and round 2 on the right. In this example, both rounds are the same, irrespectively of the initial allocation. This is because round 1 has a unique optimal solution.

round 3, the nucleolus may lead to choosing a different optimal solution from the set of optimal solutions. In that case, different patient-donor pairs may leave the IKEP. Consequently, the compatibility graph for round 4 may be different from the compatibility graph for round 4 if the Shapley value had been used.

We now return to our remark regarding the possible accumulation of credits, as for every country $p \in N$ and round $h \geq 2$, it holds that $c_p^h = \sum_{t=1}^{h-1}(y_p^t - s_p(\mathcal{C}^t))$. Suppose that round 3 and every future round looks exactly the same as round 2, and suppose that the nucleolus is used as the target allocation. Then, $c_1^h = c_1^{h-1} + 1$ for $h \geq 3$. This is clearly not desirable. We monitor whether this situation happens in our simulations. However, as mentioned, we did not see this kind of behaviour occur in our experiments (see Chapters 4–6).

Moreover, by novelly incorporating the credits into an associated game (N^r, v^r) for some round r in an IKEP, we obtain the *credit-adjusted game* $(N^r, \overline{v}^r)$, that is $\overline{v}^r(S) = v^r(S) + \sum_{p \in S} c_p^r$ for every $S \subseteq N^r$. For more details on the credit-adjusted game, see Chapter 2.

1.3 Related Work

1.3.1 Credit Systems

As mentioned, Klimentova et al. [10] introduced a credit system to incentivize the countries for collaborating by sharing the joint benefits in a fair way. In their simulations, they allowed $\ell = 3$ for four countries. They considered two solution concepts from cooperative game theory (see Chapter 2), the potential and benefit

		round 1		
	country 1	country 2	country 3	$\sum$
y^1	$\frac{2}{3}$	$\frac{8}{3}$	$\frac{2}{3}$	4
$x^1 = y^1 + c^1$	$\frac{2}{3}$	$\frac{8}{3}$	$\frac{2}{3}$	4
$s^1(\mathcal{C}^1)$	1	2	1	4
$c^2 = x^1 - s^1(\mathcal{C}^1)$	$-\frac{1}{3}$	$\frac{2}{3}$	$-\frac{1}{3}$	0
		round 2		
	country 1	country 2	country 3	$\sum$
y^2	$\frac{4}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	2
$x^2 = y^2 + c^2$	1	1	0	2
$s^2(\mathcal{C}^2_*)$	1	1	0	2
$c^3 = x^2 - s^2(\mathcal{C}^2_*)$	0	0	0	0

Table 1.1: Summary of computing credits for three countries in an IKEP over the first two rounds when the Shapley value is selected, corresponding to the example of Figure 1.6. By selecting the Shapley value (see Chapter 2), we obtain the initial fair allocations y^1 and y^2 for rounds 1 and 2, respectively.

value. Solution concepts prescribe a set of fair allocations depending on the context and each solution concepts has its own fairness properties. They used the two solution concepts mentioned above as the initial allocations, which become the target allocations after the credit adjustment. Their results showed that using the benefit value yields slightly more balanced solutions. We note that the potential value is outperformed by the benefit value in [10]. Hence, we decided not to consider the potential value in this thesis and replaced it by a different easy-to-compute solution concept, namely the contribution value (see Chapter 2). Biró et al. [11] did simulations for two solution concepts, the benefit value and Shapley value. In their simulations, for three countries allowing $\ell = 3$, they found that the Shapley value produced smaller deviations from the targets on average. Due to Theorem 1, the simulations for $\ell = 3$ in [10, 11] were for up to four countries. Among the set of optimal solutions that satisfy individual rationality, Klimentova et al. [10] and Biró et al. [11] selected the one that minimizes the total deviations across all countries.

Biró et al. [2] considered credit-based compensation systems from a theoretical point of view. They only allowed for $\ell = 2$ and showed the following result.

Theorem 2 ([2]). *For IKEPs when $\ell = 2$, the problem of finding an optimal solution that is strongly close to a given target allocation x is polynomial-time solvable.*

They also showed that the introduction of weights for representing transplant utilities makes the problem NP-hard even for two countries.

1.3.2 Other Approaches

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 [19–21] for some results in this setting. Although the consecutive matching setting is usually more acceptable for the participating countries, it results in a lower total number of kidney transplants compared to IKEPs that merge all patient-donor pairs from individual KEPs into a single, unified pool for each round.

Carvalho and Lodi [19] gave a polynomial-time algorithm for computing a Nash-equilibrium that maximizes the total number of transplants when $\ell = 2$, improving the previously known result of [20] for two countries.

Sun, Todo, and Walsh [22] also considered $\ell = 2$ only. They defined so-called selection ratios using various lower and upper target numbers of kidney transplants for each country. In their setting, a solution is considered to be fair if the minimal ratio across all countries is maximized. They also required a solution to be a maximum matching (optimal solution) and individually rational. They gave theoretical results specifying which models admit solutions with all three properties. Moreover, they provided polynomial-time algorithms for computing such solutions, if they exist.

Blom, Smeulder, and Spieksma [23] introduced mechanisms making the solution of an IKEP *rejection-proof*, meaning that no country has the incentive to reject the selected solution in favor of running a partially or fully internal national KEP, as the rejection does not results in extra transplants. They also showed that computing a set of optimal rejection-proof solutions is Σ_2^p-complete for a constant $\ell \geq 3$. While it is polynomial for $\ell = 2$, this follows from the result of Carvalho and Lodi [19]. They gave a polynomial time algorithm for finding a Nash equilibrium that maximizes the total number of transplants for $\ell = 2$.

Mincu et al. [24] allow participating countries in IKEPs to have different constraints and goals with respect to the national cycles and chains they may get involved in. In particular, they compared the expected gains from cooperation between two countries with varying country sizes and different restrictions on maximum national cycle lengths, and thus also possible different constraints on the segments of the international cycles they are involved in, that is, the national parts of an international cycle. For the theoretical foundations of country-specific parameters in IKEPs, see [25].

1.3.3 the US Setting

Individual rationality [26, 27] and fairness versus optimality [28–31] were initially studied for national KEPs, in particular in the US. However, the US situation is different from many other countries. The US has three nationwide KEPs (UNOS, APD, NKR) [32], and US hospitals work independently and compete with each other. Hence, US hospitals tend to register only the hard-to-match patient-donor pairs to the national KEPs, while they try to process their easy-to-match pairs immediately. As a consequence, the aforementioned papers focused on mechanisms that give incentives for hospitals to register all their patient-donor pairs at the KEP. In particular, NKR (the largest nationwide KEP in the US) uses a credit system to incentivize hospitals to register also their easy-to-match pairs by giving negative credits for registering hard-to-match pairs and positive credits for registering easy-to-match pairs.

We consider IKEPs in the setting of European KEPs which are scheduled in rounds, typically once in every three months [12]. Unlike the US setting, this setting allows a search for optimal exchange schemes. Hence, the situation where easy-to-match patient-donor pairs are taken out of the pool is no longer relevant. Above we discuss existing work for the European setting. As we see, the credit system proposed for the European setting [2, 10] is different from the one used by NKR due to the different nature of the European and US settings.

1.4 Our Contributions

We consider IKEPs with different maximum cycle lengths $\ell \in \{2, 3, \infty\}$ and perform a *large scale* experimental study on finding balanced solutions in IKEPs, up to 15 countries for $\ell = 2$, 10 countries for $\ell = \infty$ and 8 countries for $\ell = 3$, whereas previous studies did experiments up to 4 countries for $\ell = 3$ [10]. Exchange bounds of $\ell = 2$ or $\ell = 3$ are implemented in many European countries, such as France and UK and thus have significant practical implications. Even though the assumption of $\ell = \infty$ is not realistic in kidney exchange, it has also been made for national KEPs to obtain general results of theoretical interest [33–36] and may have wider applications (e.g. in financial clearing). We also aim to research how stability and the total number of kidney transplants change for large international kidney exchange schemes when moving from $\ell = 2$ to $\ell = 3$, and from $\ell = 3$ to $\ell = \infty$.

Our motivation is fourfold. Firstly, these days, IKEPs have a growing number of countries. We do simulations for IKEPs with the larger number of countries, as mentioned, 15 for $\ell = 2$ and 8 for $\ell = 3$ (compared to 4 countries in the previous studies [10]) and 10 for $\ell = \infty$. We contribute to the simulation studies on IKEPs regarding $\ell = 2$, $\ell = 3$ and $\ell = \infty$ in a consistent and thorough way, which are strictly guided by our and previous theoretical results. Secondly, we investigate the effect of using 7 widely accepted solution concepts as initial allocations, adjusted by credits, which result in 7 target allocations and enhance the robustness of our simulation results. In contrast, previous studies have only considered 2 solution concepts [2,10], such as the Shapley value and benefit value. Computing 7 solution concepts requires overcoming significant computing challenges, as some solution concepts, such as the Shapley value and nucleolus, are hard to compute, while others, like the tau value, might not even always exist. Thirdly, we introduce the strongly close optimal solution and show the complexity of computing strongly close optimal solutions for $\ell = \infty$. It is polynomial-time solvable for $\ell = 2$ [2] while it is NP-hard (if $\mathsf{P} \neq \mathsf{NP}$) for $\ell = \infty$. Based on that, we present our simulation results, including strongly close optimal solutions and weakly close optimal solutions with $\ell \in \{2, 3, \infty\}$. This represents an advancement over previous studies, which primarily focused on the optimal solution that minimizes the total deviations [10, 11]. Last but not least, we introduce a novel credit compensation system, which incorporates credits into the game model directly, leading to credit-adjusted games. In the following, I will outline the structure of the thesis.

In Chapter 2, we define some background information from cooperative game theory [37], including solution concepts for prescribing the initial allocations. We also define relevant cooperative games, namely matching games, permutation games and their generalizations. Finally, we summarize the known computational complexity results of the core, nucleolus and Shapley value respectively, for assignment games, b-assignment games, matching games, b-matching games and partitioned matching games.

In Chapter 3, we describe the identical simulation setup for all simulations across $\ell = 2$, $\ell = 3$ and $\ell = \infty$, the selection of optimal solutions and metrics for evaluating stability under different values of ℓ.

For Chapters 4–6, we present our simulations results on three different values of ℓ. The chapters are structured to begin with two extreme cases, namely, $\ell = 2$ and $\ell = \infty$, and then proceed to the case of $\ell = 3$. This order also reflects the increasing computational difficulty: computing optimal solutions is polynomial-time solvable for $\ell = 2$ and $\ell = \infty$, but NP-hard for $\ell = 3$; see Theorem 1.

In Chapter 4, due to Theorem 2, we are able to conduct simulations for strongly close optimal solutions up to 15 countries for $\ell = 2$ with equal and varying country sizes and for a large variety of different solution concepts. Also, we present our simulation results from the perspective of stability, the average number of transplants across 24 rounds and the computational time for $\ell = 2$.

In Chapter 5, we start with our theoretical results for $\ell = \infty$. Our theoretical results highlight severe computational limitations, and we now turn to our simulations. We still exploit the fact that for $\ell = \infty$ (Theorem 1) we can find optimal solutions and values of associated games (N, v) (see Chapter 2) in polynomial time. In this way we can still perform simulations for IKEPs up to 10 countries, in which our simulations are strongly guided by our theoretical results. We formulate the problems of computing a weakly or strongly close optimal solution as ILPs. This enables us to use an ILP solver. We also present our simulation results and do comparisons with $\ell = 2$.

In Chapter 6, since computing the optimal solution for $\ell = 3$ is NP-hard, we formulate our problems into ILPs and use the ILP solver to solve them. We conduct our simulations with the identical simulation setup for up to 8 countries, larger than the 4 countries in the simulations for $\ell = 3$ [10, 11], but less than 15 countries for $\ell = 2$ and 10 countries for $\ell = \infty$. Additionally, we present the results of our simulations and compare results of $\ell = 2$.

In Chapter 7, we sum up our theoretical and simulation results for $\ell = 2$, $\ell = \infty$ and $\ell = 3$. Our simulations show that a credit system using strongly close optimal solutions instead of weakly optimal solutions makes an IKEP most balanced, without decreasing the overall number of transplants. The exact improvement is determined by the choice of solution concept. The credit system makes the IKEP more balanced. We used seven solution concepts for computing the initial allocations. Our simulations indicate that out of these seven solution concepts, the Banzhaf* value (a variant of the Banzhaf value) yields the most balanced solution. Furthermore, for equal country sizes, moving from $\ell = 2$ to $\ell = 3$ yields on average 2.73% more kidney transplants, while extending from $\ell = 3$ to $\ell = \infty$ further increases transplants by an average of 42.06% based on the same simulation instances generated by the data generator [38]. However, the exchange cycles may be very large for $\ell = \infty$, in particular in the starting round. Chapter 7 contains some concluding remarks, In this chapter we also present relevant open problems resulting from our work.

1.5 Publications

The work related to this thesis has been published in the following Table 1.2, while the journal version of [39] is still under review and Chapter 6 is a new chapter that contains unpublished material. All publications are joint work with collaborators, and the order of author follows the traditionally alphabetical convention in the field of theoretical computer science. I made significantly contributions to every paper and was the main contributor to simulations performed in each of the papers listed in Table 1.2.

Title	**Authors**	**Publication Venue and Year**
The Complexity of Matching Games: A Survey	Márton Benedek, Péter Biró, Matthew Johnson, Daniël Paulusma, **Xin Ye**	Journal of Artificial Intelligence Research, 2023 [40]
Computing Balanced Solutions for Large International Kidney Exchange Schemes	Márton Benedek, Péter Biró, Daniël Paulusma, **Xin Ye**	Journal of Autonomous Agents and Multi-Agent Systems, 2024 [41]
Computing Balanced Solutions for Large International Kidney Exchange Schemes When Cycle Length is Unbounded	Márton Benedek, Péter Biró, Gergely Csáji, Matthew Johnson, Daniël Paulusma, **Xin Ye**	Proceedings of 23rd International Conference on Autonomous Agents and Multiagent Systems, 2024 [39]

Table 1.2: List of Publications

CHAPTER 2

Cooperative Game Theory

In this Chapter, we introduce some background information on Cooperative Game Theory in Section 2.1 and additional concepts from Graph Theory in Section 2.2. This enables us to formally define the concept of a matching game, a permutation game and closely related generalizations of matching games and permutation games in Section 2.3, including assignment games, b-matching games, b-assignment games, partitioned matching games, ℓ-permutation games, permutation games, partitioned ℓ-permutation games and partitioned permutation games. In the same section, we also define the fractional matching game, a known relaxation of a matching game. Specifically, in the context of IKEPs, we also model the case of $\ell = 2$ as a partitioned matching game, the case of $\ell = \infty$ as a partitioned permutation game and the case of $\ell = 3$ as a partitioned 3-permutation game in this section. In Section 2.4 we discuss the well-known network bargaining games, which have a close relationship with matching games.

In Sections 2.5–2.7 we survey known computational complexity results of the core, nucleolus and Shapley value, respectively, for assignment games, b-assignment games, matching games, b-matching games and partitioned matching games. Unless specified otherwise, running times of algorithms are expressed in the input size

$n + m$ of the underlying (weighted) graph $G = (V, E)$ where $n = |V|$ and $m = |E|$. In Section 2.5 we explain to what extent solutions for network bargaining games correspond to finding core allocations of matching games and their variants.

2.1 Basic Terminology

Cooperative Game Theory considers *fair* distributions of joint profit if all parties involved collaborate (the notion of fairness depends on the context). Before describing the role of fairness in our setting, we first give some terminology and we refer to [42] for more details on cooperative game theory. A *(cooperative) game* is a pair (N, v), where N is a set of n *players* and $v : 2^N \to \mathbb{R}$ is a *value function* with $v(\emptyset) = 0$. A subset $S \subseteq N$ is a *coalition*. If, for every possible partition $(S_1, \ldots, S_r)$ of N it holds that $v(N) \geq v(S_1) + \cdots + v(S_r)$, then players will benefit most by forming the *grand coalition* N. The problem is then how to fairly distribute $v(N)$ amongst the players of N.

An *allocation* (or *pre-imputation*) for a game (N, v) is a vector $x \in \mathbb{R}^N$ with $x(N) = v(N)$; here, for $S \subseteq N$, we write $x(S) = \sum_{i \in S} x_i$. A *solution concept* prescribes a set of allocations for every game. Each solution concept has its own fairness properties and computational complexities; the choice of a certain solution concept depends on the context.

The solution concepts considered in this thesis are the Shapley value, Banzhaf value, Banzhaf* value (a variant of the Banzhaf value), nucleolus, tau value, benefit value, contribution value and the core. They all prescribe a unique allocation except for the *core*.

The *core* was formally introduced by Gillies [43] and is the best-known solution concept. It 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, as no subset S of N will benefit from forming their own coalition. But the core of a cooperative game may be empty. In Section 2.3 we illustrate by an example that this may happen even in cases involving only three players.

Should the core be empty we might seek an allocation x that satisfies all core constraints "as much as possible" by solving the following linear program:

$$\begin{aligned} \text{(LC)} \quad \varepsilon_1 := \quad & \max \varepsilon \\ & x(S) \geq v(S) + \varepsilon \quad \text{for all } S \in 2^N \backslash \{\emptyset, N\} \\ & x(N) = v(N). \end{aligned}$$

The set of allocations x for which (x, ε_1) is an optimum solution of (LC) was introduced by Maschler, Peleg and Shapley [44] as the *least core* of (N, v). The least core belongs to the core if the core is non-empty, which is the case if and only if $\varepsilon_1 \geq 0$. Note that by excluding the sets $\emptyset$ and N in the definition of (LC), the least core might be a proper subset of the core, should the core be non-empty.

In contrast to the core, the *Shapley value* $\phi(N, v)$ always exists. It assigns every player $i \in N$:

$$\phi_i(N, v) = \sum_{S \subseteq N \backslash \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \Big(v(S \cup \{i\}) - v(S) \Big).$$

We call $v(S \cup \{i\}) - v(S)$ the marginal contribution of player i to coalition $S \cup \{i\}$. In essence, the Shapely value assigns player i the weighted average of all marginal contributions of player i, where the weight of each marginal contribution reflects the assumption that all possible orderings are equally likely. However, the Shapley value might be negative but not for the games we consider. Moreover, the Shapley value might not be in the core even if the core is non-empty, as we demonstrate in Section 2.3 with an example. However, it is well known that a convex game, that is, for every two coalitions S and T it holds that $v(S \cup T) + v(S \cap T) \geq v(S) + v(T)$, has a non-empty core that contains the Shapley value [45].

Moreover, the Shapely value is the unique singleton solution concept that satisfies the following three axioms of fairness [46].

Axiom 1 ("Dummy axiom"): If player i with respect to $S \subset N$, $i \notin S$ is such that $v(S \cup \{i\}) - v(S) = \phi_i(N, v)$ for all $S \not\ni i$, then

$$v(\{i\}) = \phi_i(N, v).$$

Axiom 2 ("Symmetry"): If $i \neq j$ with respect to $S \subset N$, $i, j \notin S$ is such that $v(S \cup \{i\}) = v(S \cup \{j\})$ for all $S \not\ni i, j$, then

$$\phi_i(N, v) = \phi_j(N, v).$$

Axiom 3 ("Linearity"): For any two games (N, v) and (N, w), where both v and w are nonnegative value functions with $v(\emptyset) = w(\emptyset) = 0$, then

$$\phi(N, v + w) = \phi(N, v) + \phi(N, w).$$

We now define another well-known solution concept, namely the nucleolus. In contrast to the Shapley value, the nucleolus is always in the core if the core is non-empty. The *excess* of a non-empty coalition $S \subsetneq N$ for an allocation x is given by $e(S, x) = x(S) - v(S)$. Ordering the $2^{|N|} - 2$ excesses non-decreasingly gives us the *excess vector* $e(x)$. The *nucleolus* $\eta(N, v)$ is the unique allocation [47] that lexicographically maximizes $e(x)$ over the set of *imputations* $I(N, v)$, which consists of all allocations x with $x_i \geq v(\{i\})$ for $i \in N$; the nucleolus is only defined if $I(N, v)$ is non-empty. Solution concepts, such as the nucleolus, are individual rational, because they only prescribe allocations that are imputations. In Section 2.5 we give an example, for which we compute the nucleolus.

An alternative equivalent definition of the nucleolus is given by Maschler, Peleg and Shapley [44]; solve for $r \geq 1$, the following sequence of linear programs until a unique solution, which gives us the nucleolus, remains:

$$
\begin{aligned}
(\mathrm{LP}_r)\ \varepsilon_r := \max \varepsilon & \\
& x(S) \geq v(S) + \varepsilon \quad \text{for all } S \in 2^N \backslash \{\mathcal{S}_0 \cup \ldots \cup \mathcal{S}_{r-1}\} \\
& x(S) = v(S) + \varepsilon_i \quad \text{for all } S \in \mathcal{S}_i \quad (0 \leq i \leq r-1) \\
& x \in I(N, v),
\end{aligned}
$$

where $\varepsilon_0 = -\infty$, $\mathcal{S}_0 = \emptyset$ and for $i \geq 1$, $\mathcal{S}_i$ consists of all coalitions S with $x(S) = v(S) + \varepsilon_i$ for every optimal solution (x, ε_i) of (LP_i). From this definition and the definition of the least core, it follows that if the core is non-empty, then the nucleolus exists and even belongs to the least core.

Additionally, we define the pre-kernel, introduced by Maschler, Peleg and Shapley [48]. We let

$$s_{ij}(x) = \min\{e(S, x) \mid S \subseteq N \text{ with } i \in S, j \notin S\}.$$

The *pre-kernel* of a game (N, v) is the set of all allocations $x \in \mathbb{R}^N$ with $s_{ij}(x) = s_{ji}(x)$ for every pair of distinct players $i, j \in N$.

Next, we define the tau value, introduced by Tijs [49]. 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

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

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 *quasi-balanced* if $a \leq b$ and $a(N) \leq v(N) \leq b(N)$. The tau value is only defined for quasi-balanced games. The tau value τ of a quasi-balanced 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 *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 $p \in N$. Such allocations ensure individual rationality, as $\alpha_p \cdot \text{surp} \geq 0$ holds for each player $p \in N$ in our setup and we immediately obtain $x(\{p\}) \geq v(\{p\})$ for all players $p \in N$. In this way we obtain the *benefit value* 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* 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 only $2n+1$ v-values need to be computed when $n \geq 3$ and only $2n-1$ v-values for $n = 2$. We also note that the benefit value and contribution value do not exist if the denominator is zero.

Moreover, the benefit value coincides with the tau value when the game is convex [50]. Examples of non-convex games where the benefit value coincides with the tau value can be found in Section 2.3.

To define the next solution concept, we first introduce *unnormalized Banzhaf value* $\psi(N, v)$ [51] 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* $\overline{\psi}(N, v)$ of a game

(N, v) was introduced and defined by setting for $p \in N$,

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

Whenever we mention the Banzhaf value, we will mean $\overline{\psi}(N, v)$.

Recall the credit-adjusted game $(N^r, \overline{v}^r)$ for round r in Chapter 1, that is $\overline{v}^r(S) = v^r(S) + \sum_{p \in S} c_p^r$ for every $S \subseteq N^r$. For solution concepts that are *covariant (under strategic equivalence)*, i.e., that prescribe the same set of allocations both for (N^r, v^r) and for $(N^r, \gamma v^r + \delta)$ for every $\gamma > 0$ and every $\delta \in \mathbb{R}^n$, this credit based system works in exactly the same way as the one introduced by Klimentova et al. [10]. All solution concepts that we consider in simulations (see Chapter 3) have this property except for one: the (normalized) Banzhaf value. We denote the adjusted Banzhaf value as the *Banzhaf* value*.

2.2 Additional Graph Theory Terminology

Throughout this section, the graph G is an undirected graph with no self-loops and no multiple edges. Let V be the vertex set of G, and let E be the edge set of G. We denote an edge between two vertices i and j as ij (or ji). For a subset $S \subseteq V$, we let $G[S]$ denote the subgraph of G *induced* by S, that is, $G[S]$ is the graph obtained from G after deleting all vertices that are not in S (and their incident edges). We say that G is *bipartite* if there exist two disjoint vertex subsets A and B such that $V = A \cup B$ and every edge in E joins a vertex in A to a vertex in B.

A function f that maps some set X to $\mathbb{R}$ is positive if $f(x) > 0$ for every $x \in X$. For an integer t, we write $f \equiv t$ if $f(x) = t$ for every $x \in X$, and we write $f \leq t$ if $f(x) \leq t$ for every $x \in X$.

A *fractional b-matching* of G is an edge mapping $f : E \to [0, 1]$ such that $\sum_{e: i \in e} f(e) \leq b(i)$ for each $i \in V$. The *weight* of a fractional b-matching f is defined as $w(f) = \sum_{e \in E} w(e)f(e)$. If $b \equiv 1$, then f is a *fractional matching*. Note that f is a b-matching of G if and only if $f(e) \in \{0, 1\}$ for every $e \in E$. We say that f is a *half-b-matching* of G if $f(e) \in \{0, \frac{1}{2}, 1\}$ for every $e \in E$. If $b \equiv 1$, then a half-b-matching of G is also called a *half-matching*. See Figure 2.1 for an example

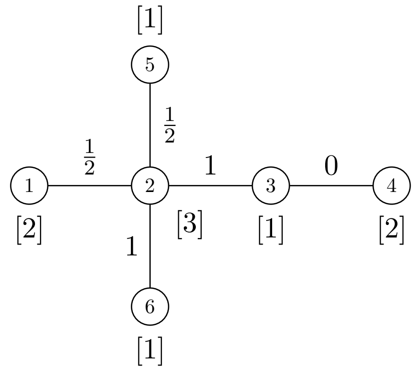

Figure 2.1: An example of a half-b-matching f in a graph G with $w \equiv 1$ and where b is given by $b(3) = b(5) = b(6) = 1$; $b(1) = b(4) = 2$; and $b(2) = 3$, as illustrated by the $[x]$ labels in the figure. Note that $f(12) = f(25) = \frac{1}{2}$; $f(23) = f(26) = 1$; $f(34) = 0$; and $w(f) = 3$.

of a half b-matching.

2.3 Matching Games and Generalizations

In this section, we define matching games, assignment games, b-matching games, b-assignment games, partitioned matching games, ℓ-permutation games, permutation games, partitioned ℓ-permutation games and partitioned permutation games. We also show how two examples can be seen as ℓ-permutation games and partitioned matching games, respectively.

Definition 1. [9] A *matching game* on a undirected graph $G = (V, E)$ with a positive edge weighting w is the game (N, v) where $N = V$ and for $S \subseteq N$, the value $v(S)$ is the maximum weight $w(M)$ over all matchings M of $G[S]$. We obtain a *uniform matching game* if $w \equiv 1$.

Matching games on bipartite graphs are commonly known as *assignment games* [7].

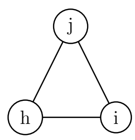

Figure 2.2: The graph K_3, whose corresponding uniform matching game has an empty core.

Already uniform matching games may have an empty core. For example, consider the uniform matching game (N, v) defined on a graph G that is a triangle with vertices h, i, j and edges hi, ij and jh; see also Figure 2.2. The reason for core-emptiness for matching games is that every imputation x has a so-called blocking pair. That is, a pair of adjacent players $\{i, j\}$ in a matching game (N, v) is a *blocking pair* for an imputation x if $x_i + x_j < w(ij)$ holds. The *blocking value* of $\{i, j\}$ for x is defined as

$$e_x(ij)^+ = \max\{0, w(ij) - (x_i + x_j)\}.$$

If $\{i, j\}$ is not blocking x, then $e_x(ij)^+ = 0$; otherwise, $e_x(ij)^+$ expresses the extent to which $\{i, j\}$ blocks x. We let $B(x) = \sum_{ij \in E} e_x(ij)^+$ be the *total blocking value* for x. Deng, Ibaraki and Nagamochi [9] showed that an imputation x is in the core of a matching game (N, v) defined on a graph G with positive edge weighting w if and only if for every $ij \in E$,

$$x_i + x_j \geq w(ij),$$

so if and only if $B(x) = 0$ and x has no blocking pairs. This follows as if there is a blocking pair then x is not in the core, and, for the other direction, suppose that S is a blocking coalition. Let M be a maximum weight matching in $G[S]$. Then $x(S) < v(S) = w(M) = \sum_{e_i \in M} w(e_i)$, and so for at least one of the edges $ij \in M$, $x_i + x_j < w(ij)$ must hold; hence x not in the core. For a (larger) example of a matching game with a non-empty core, we refer to Example 2.3 in Section 2.5.

However, already any (uniform) matching game that contains a 3-vertex path uvw as a subgraph is not convex: take $S = \{u, v\}$ and $T = \{v, w\}$ and note that $v(S \cup T) + v(S \cap T) = 1 < 2 = v(S) + v(T)$. Recall that in convex games, the benefit value coincides with the tau value; see also Section 2.1. Nevertheless, the tau value and benefit value may also coincide for non-convex uniform matching games: e.g. take the 4-vertex cycle with unit edge weights, for which the allocation $x \equiv \frac{1}{2}$ is both the tau and benefit value.

Recall that the tau value is only defined under quasi-balanced games; see Section 2.1. Matching games with a non-empty core are quasi-balanced according to Bondareva–Shapley Theorem [52, 53]. However, as mentioned, even uniform matching games may have an empty core.

Theorem 3. *It is possible to determine in polynomial time if a uniform matching game is quasi-balanced.*

Proof. Let (N, v) be a uniform matching game. We first recall the definition of quasi-balanced games in Section 2.1: a game (N, v) is quasi-balanced if $a \leq b$ and $a(N) \leq v(N) \leq b(N)$, where a and b are n-dimensional vectors such that for every $p \in N$, $b_p = v(N) - v(N \setminus \{p\})$ and $a_p = \max_{S \ni p} R(S, p) = \max_{S \ni p}(v(S) - \sum_{q \in S \setminus \{p\}} b_q)$. We note that computing the value $v(S)$ is polynomial-time solvable for every $S \subseteq N$ due to Theorem 2. Hence, we can compute b in polynomial time. As we can also verify if $a \leq b$ holds if we are given a and b, it remains to show that we can compute a_p in polynomial time for every $p \in N$.

Let $p \in N$. Let S^* be a coalition containing p such that $R(S^*, p)$ is maximum over all $S \subseteq N$ with $p \in S$. We claim that without loss of generality

$$S^* = \{p\} \cup \{j \mid v(N \setminus \{j\}) = v(N)\},$$

which implies that we can compute a_p in polynomial time. In order to see that our choice of S is correct, we first assume that S^* does not contain some player $j \in N \setminus \{p\}$ with $v(N) = v(N \setminus \{j\})$. As $v(S^* \cup \{j\}) \geq v(S^*)$, we can replace S^* by $S^* \cup \{j\}$. We now suppose that S^* contains a player $j \in N \setminus \{p\}$ with $v(N) \neq v(N \setminus \{j\})$, so $v(N) = v(N \setminus \{j\}) + 1$. As $v(S^* \setminus \{j\}) \geq v(S^*) - 1$, we can replace S^* by $S^* \setminus \{j\}$. In other words, we may indeed assume that $S^* = \{p\} \cup \{j \mid v(N \setminus \{j\}) = v(N)\}$. □

Definition 2. [4] A *b-matching game* on a undirected graph $G = (V, E)$ with a positive vertex capacity function b and a positive edge weighting w is the game (N, v) where $N = V$ and for $S \subseteq N$, the value $v(S)$ is the maximum weight $w(M)$ over all b-matchings M of $G[S]$.

We refer to Figure 2.3 for an example. If $b \equiv 1$, then we have a matching game. If G is bipartite, then we speak of a *b-assignment game*. These games were introduced by Sotomayor [8] as multiple partner assignment games. If $b \equiv t$ for some integer t,

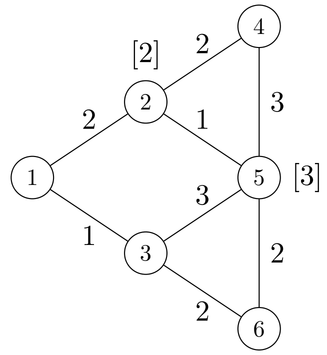

Figure 2.3: A b-matching game (N, v) with six players, where $b \equiv 1$ except $b(2) = 2$ and $b(5) = 3$, so $b^* = 3$. Note that $v(N) = 10$ (take $M = \{12, 35, 45, 56\}$).

then we speak of a t-matching game or a t-assignment game. Note that 1-assignment games correspond to assignment games. A b-matching game is called a *multiple partners matching game* in the paper of Biró et al. [4].

Definition 3. A *ℓ-permutation game* on a directed graph $G = (V, A)$ with a positive edge weighting w is the game (N, v) where $\ell \in \{2, 3, \dots, \infty\}$ and $N = V$. For $S \subseteq N$, the value $v(S)$ is the maximum weight $w(\mathcal{C})$ over all ℓ-cycle packings of $G[S]$. Two special cases are well studied. We obtain a matching game if $\ell = 2$ and a *permutation game* if $\ell = \infty$.

We can now model an KEP with maximum exchange cycle length at most ℓ as an ℓ-permutation game (N, v) that is defined on a compatibility graph $G = (V, A)$ where

- the set $N = V = \{1, \dots, n\}$ is a set of patient-donor pairs participating in the KEP; and
- the set A may have an arc weighting w, where $w(uv)$ expresses the utility of a transplant exchange between patient-donor pairs u and v (if the aim is to maximize the number of transplants we can set $w \equiv 1$).

In the next definition, the sets N (of players) and V (of vertices) are different, as we now associate each player $i \in N$ with a distinct subset V_i of the vertex set V.

Definition 4. A *partitioned matching game* [2] on a undirected graph $G = (V, E)$ with a positive edge weighting w and partition $(V_1, \dots, V_n)$ of V is the game (N, v),

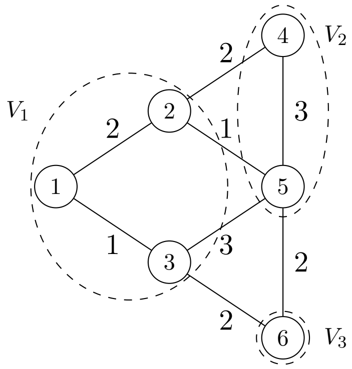

Figure 2.4: A partitioned matching game (N, v) of width $c = 3$ defined on a graph $G = (V, E)$ with an edge weighting w. Note that N consists of three players and that V is partitioned into $\{1, 2, 3\}, \{4, 5\}, \{6\}$, as indicated by the dotted circles. Also note that $v(N) = 7$.

where $N = \{1, \ldots, n\}$, and for $S \subseteq N$, the value $v(S)$ is the maximum weight $w(M)$ over all matchings M of $G[\bigcup_{i \in S} V_i]$.

Partitioned matching games were introduced as *generalized matching games* by Biró et al. [2] and are also known as *international matching games* [54]. We say that

$$c = \max\{|V_i| \mid 1 \leq i \leq n\}$$

is the *width* of (N, v). Note that if $V_i = \{i\}$ for every $i \in N$, then we obtain a matching game (which has width 1). We refer to Figure 2.4 for an example of a partitioned matching game with width 3.

We can now model an IKEP with pairwise exchanges only as a partitioned matching game (N, v) that is defined on an undirected graph $G = (V, E)$ where

- the set $N = \{1, \ldots, n\}$ is a set of countries participating in the IKEP;
- the set V is partitioned into subsets $V_1, \ldots, V_n$, where V_i consists of the patient-donor pairs of country i for every $i \in N$; and
- the graph G may have an edge weighting w, where for an edge uv, $w(uv)$ expresses the utility of a transplant exchange between patient-donor pairs u and v (if the aim is to maximize the number of transplants we can set $w \equiv 1$).

Apart from IKEPs, partitioned matching games can also be used to model economic settings where multi-organizations own pools of clients [55]. As we will discuss further in Section 2.5, Biró et al. [2] performed a theoretical study for partitioned matching games and showed how b-matching games and partitioned matching games are related to each other.

Moreover, partitioned matching games are *superadditive*, that is, for every two disjoint coalitions S and T, it holds that $v(S \cup T) \geq v(S) + v(T)$. As shown by Staudacher and Anwander [56], this means that the benefit value coincides with the Gately point [57],[1] as long as the Gately point is unique. For superadditive games, the latter holds if there exists at least one player p with $v(N) - v(N \setminus \{p\}) - v(\{p\}) > 0$ [56]. For superadditive games that do not satisfy this condition, we have for every $p \in N$ that $v(N) - v(N \setminus \{p\}) - v(\{p\}) = 0$, and thus the benefit value does not exist.

Definition 5. A *partitioned ℓ-permutation game* on a directed graph $G = (V, A)$ with a partition $(V_1, \ldots, V_n)$ of V is the game (N, v), where $\ell \in \{2, 3, \ldots, \infty\}$ and $N = \{1, \ldots, n\}$. For $S \subseteq N$, the value $v(S)$ is the maximum weight $w(\mathcal{C})$ of an ℓ-cycle packing $\mathcal{C}$ of $G[\bigcup_{i \in S} V_i]$. We obtain a *partitioned matching game* if $\ell = 2$, a *partitioned* 3*-permutation game* if $\ell = 3$ and a *partitioned permutation game* if $\ell = \infty$.

We give an example of a 3-partitioned permutation game (N, v) of width 1 with an empty core. This example can be readily generalized to work for every $\ell \geq 4$ by making the paths of length 3 between the vertices a, b, c longer.

Example 2.1. Let $G = (V, A)$ be the underlying graph that has vertex set $V = \{a, b, c, u_{ba}, u_{ca}, u_{bc}\}$ and arc set $A = \{(a, b), (b, u_{ba}), (u_{ba}, a), (a, c), (c, u_{ca}), (u_{ca}, a),$ $(c, b), (b, u_{bc}), (u_{bc}, c)\}$, see also Figure 2.5. Let $V_p = \{p\}$ for every $p \in N$. Note that G has exactly three maximum 3-cycle packings, each of which consist of exactly one 3-vertex cycle, namely $abu_{ba}a$, $bu_{bc}cb$, or $cu_{ca}ac$. For a contradiction, assume that (N, v) has a nonempty core. Let x be a core allocation. As x belongs to the core of (N, v), we find that $x \geq 0$, $x(N) = v(N) = 3$ and $x(S) \geq v(S) = 3$ for $S = \{a, u_{ba}, b\}$. Hence, $x_p = 0$ for $p \notin \{a, u_{ba}, b\}$. By symmetry, this means that

[1]The Gately point of a game (N, v) was originally defined as the point where the potential for each player to "disrupt" (by leaving N) is minimal. It occurs when the disruption, defined as $\frac{v(N) - v(N \setminus \{i\}) - x_i}{x_i - v(\{i\})}$, is equal for every player $i \in N$.

$x_p = 0$ for every $p \in N$, a contradiction. □

We model an international KEP with maximum cycle length at most ℓ where $\ell \in \{2, 3, \ldots, \infty\}$ as a partitioned ℓ-permutation game (N, v) that is defined on a directed graph $G = (V, A)$.

As mentioned in Chapter 1, our primary goal in KEPs is to maximize the number of kidney transplants, that is to find optimal solutions on the corresponding compatibility graph $G = (V, A)$. In IKEPs, we additionally ensure the stability of participating countries by selecting the optimal solution that approximates the target allocation as cloesly as possible. Therefore, we set $w \equiv 1$ for Chapters 3–6.

We finish this section by defining a known relaxation of matching games. A *fractional matching game* on a graph $G = (V, E)$ with a positive edge weighting w is the game (N, v) where $N = V$ and for $S \subseteq N$, the value $v(S)$ is the maximum weight $w(M)$ over all fractional matchings M of $G[S]$. We can define the notions of a *fractional b-assignment game* and *fractional b-matching game* analogously. These notions have been less studied than their non-fractional counterparts and are not the focus of our thesis. Nevertheless, we will mention relevant complexity results for these games as well.

2.4 Network Bargaining Games

Some economic situations involve only preferences of players. These situations are beyond the scope of this thesis. The economic setting that is described in this section includes payments. As we will explain in Section 2.5, this setting is closely related

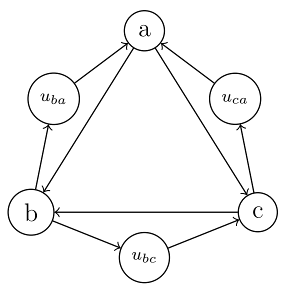

Figure 2.5: An example of a 3-partitioned permutation game of width 1 with an empty core.

to the setting of b-matching games.

Let G be a graph with a positive edge weighting w and a positive vertex capacity function b. Previously we defined a b-matching game (N, v) on (G, b, w), but the triple (G, b, w) is also known as a *network bargaining game*. Again, the vertices are players, but now an edge ij represents a possible contract between players i and j. The weight $w(ij)$ represents the value of the contract, while the *capacity* $b(i)$ is the maximum number of contracts player i can commit to. A b-matching M can now be viewed as a set of pairwise contracts between players. However, M might be *blocked* by two players i and j. To define the notion of a blocking pair in this context we need some further terminology.

If $ij \in M$, we distribute the value $w(ij)$ of the contract ij by defining *pay-offs* $p(i, j) \geq 0$ and $p(j, i) \geq 0$ to i and j, respectively, such that

$$p(i, j) + p(j, i) = w(ij).$$

If $ij \notin M$, then we set $p(i, j) = p(j, i) = 0$. The vector p with entries $p(i, j)$ is a *pay-off* and the pair (M, p) is a *solution* for (G, b, w). The *total pay-off vector* p^t is defined by $p^t(i) = \sum_{j:ij \in E} p(i, j)$ for every $i \in N$. The *utility* $u_p(i)$ is equal to $\min\{p(i, j) \mid ij \in M\}$ if i is incident to $b(i)$ edges of M; otherwise $u_p(i) = 0$. We can now define a *blocking pair* with respect to (M, p) as a pair $\{i, j\}$ with $ij \in E \setminus M$ such that

$$u_p(i) + u_p(j) < w(ij).$$

The goal is to decide if (G, b, w) has a *(pairwise) stable* solution, that is, a solution with no blocking pairs. This problem is called the STABLE FIXTURES WITH PAYMENTS (SFP) problem [4]. Restrictions of SFP are called:

- MULTIPLE PARTNERS ASSIGNMENT (MPA) if G is bipartite [8];
- STABLE ROOMMATES WITH PAYMENTS (SRP) if $b \equiv 1$ [58];
- STABLE MARRIAGE WITH PAYMENTS (SMP) if G is bipartite, $b \equiv 1$ [6].

We denote instances of SRP and SMP by (G, w) instead of $(G, 1, w)$.

It is known that SFP (and thus MPA, SMP, SRP) is polynomial-time solvable [4, 5].

The main reason for discussing network bargaining games in our thesis is due to their insightful relationship with the core of b-matching games (see Section 2.5).

2.5 Complexity Aspects of the Core

In Section 2.5.1 we survey complexity results for the problems of core membership, core non-emptiness and finding core allocations for matching games and the variants of matching games defined in Section 2.3.

2.5.1 Testing Core Membership, Non-Emptiness and Finding Core Allocations

We consider three natural problems, which have also been studied for other games and with respect to other solution concepts. Each of these problems takes as input a game (N, v).

P1. (core membership) decide if a given allocation x belongs to the core, or else find a coalition S with $x(S) < v(S)$;

P2. (core non-emptiness) decide if the core is non-empty; and

P3. (core allocation computation) find an allocation in the core (if it is non-empty).

We note that if P1 is polynomial-time solvable for some class of games $\mathcal{G}$, then by using the ellipsoid method [59,60], it follows that P2 and P3 are also polynomial-time solvable for $\mathcal{G}$.

In the remainder we explain that the problems SMP, MPA, SRP and SFP are closely related to problems P1–P3 for the corresponding matching game variant. To help the reader keep track of this we refer to Table 2.1 for an overview (below we will explain the results in this table as part of our discussion).

	bipartite graphs	general graphs
$b \equiv 1$	SMP assignment game	SRP matching game
stable solution core	*exists*[1] *always non-empty*[2]	*may not exist* *may be empty*
any b	MPA b-assignment game	SFP b-matching game
stable solution core	*exists*[3] *always non-empty*[3]	*may not exist* *may be empty*

Table 2.1: The four variants of network bargaining games and their counterpart matching games. Recall that SFP (and thus MPA, SMP, SRP) is polynomial-time solvable [4,5]. The references for the statements on the existence of a stable solution and on the core are [1] [6], [2] [7], [3] [8]. See Table 2.2 for a summary of the complexity results for matching games.

We start by considering the assignment games and b-assignment games. The problems SMP and MPA are closely related to problems P2 and P3 for assignment games and b-assignment games, respectively. Shapley and Shubik [7] proved that every core allocation of an assignment game corresponds to a pay-off vector in a stable solution for SMP and vice versa. Moreover, Koopmans and Beckmann [6] proved that SMP, and thus P2, only has yes-instances and that a stable solution, and thus a core allocation of every assignment game, can be found in polynomial time. Finally, Sotomayor [8] extended all these results to MPA and b-assignment games; recall that p^t denotes the total pay-off vector.

Theorem 4 ([8]). *Every instance (G, b, w) of* MPA *has a stable solution, which can be found in polynomial time. Moreover, if (M, p) is a stable solution for (G, b, w), then M is a maximum weight b-matching of G and p^t is a core allocation of the b-assignment game defined on (G, b, w).*

The converse of the second statement of Theorem 4 does not hold, as there may exist an allocation in the core which does not correspond to a stable solution [8].

In contrast to the situation for assignment games, the core of a matching game defined on a non-bipartite graph may be empty; just recall the example from Section 2.3 where G is the triangle and $w \equiv 1$. Problem P1 is linear-time solvable for matching games, and thus also for assignment games. Recall from Section 2.3 that it suffices to check if for an allocation x, we have for every $ij \in E$,

$$x_i + x_j \geq w(ij).$$

As P1 is polynomial-time solvable, we immediately have that P2 and P3 are also polynomial-time solvable for matching games. In particular, the following connection with SRP is well known; note that for SRP, total pay-off vectors and pay-off vectors are in 1-1-correspondence (see also Example 2.2).

Theorem 5 ([58]). *A solution (M, p) for an instance (G, w) of* SRP *is stable if and only if M is a maximum weight matching of G and p^t is a core allocation of the matching game defined on (G, w).*

Deng, Ibaraki and Nagamochi [9] followed a more direct approach for solving P2 and P3 for matching games. Later, a faster algorithm for P2 and P3 was given by Biró et al. [61]. This algorithm runs in $O(nm + n^2 \log n)$ time and is based on an alternative characterization of the core of a matching game, which follows from a result of Balinski [62].

Theorem 6 ([61]). *The core of a matching game (N, v), on a graph G with positive edge weighting w, is non-empty if and only if the maximum weight of a matching in G equals the maximum weight of a half-matching in G.*

In order to use Theorem 6 algorithmically, Biró, Kern and Paulusma [61] used the duplication technique of Nemhauser and Trotter Jr. [63], as illustrated in Figure 2.6. That is, for an input graph G with edge weighting w, first construct in polynomial-time the so-called *duplicate* G^d with edge weighting w^d, as follows. For every vertex $i \in V$, we introduce two vertices i' and i'' in G^d. For every edge $ij \in E$, we introduce the edges $i'j''$ and $i''j'$, each of weight $w^d(i'j'') = w^d(i''j') = \frac{1}{2}w(ij)$, in G^d. It is not difficult to see that the maximum weight of a half-matching in G equals the maximum weight of a matching in G^d. Hence, it remains to compare the latter weight with the maximum weight of a matching in G. One can compute both these values in $O(n^3)$ time by using Edmonds' algorithm [64].

We now give two examples. The goal of Example 2.2 is to illustrate the working

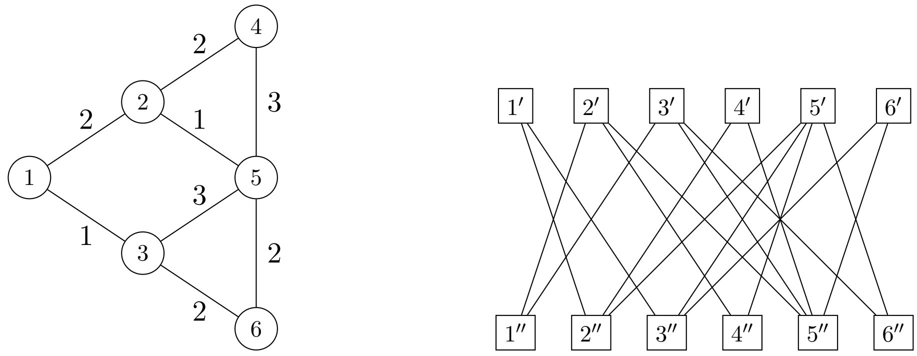

Figure 2.6: *Left:* an example of a matching game on a graph G. *Right:* the duplicate G^d of G; note that, for instance, $w^d(3'5'') = \frac{3}{2}$. See also Examples 2.2 and 2.3.

of Theorems 5 and 6, whereas Example 2.3 gives another illustration of the power of Theorem 6.

Example 2.2. Consider the graph G and its duplicate G^d in Figure 2.6. The matching $M = \{12, 36, 45\}$ is a maximum weight matching of G with maximum weight 7. The maximum weight of a matching in G^d is also equal to 7. Hence, by Theorem 6, we find that the core is non-empty and thus the nucleolus $\eta = (\frac{1}{2}, \frac{3}{2}, \frac{3}{2}, 1, 2, \frac{1}{2})$ is a core allocation, as can be readily verified. Moreover, the pair (M, p), where p is defined by $p(1,2) = \frac{1}{2}$, $p(2,1) = \frac{3}{2}$, $p(3,6) = \frac{3}{2}$, $p(4,5) = 1$, $p(5,4) = 2$ and $p(6,3) = \frac{1}{2}$ and else $p(i,j) = 0$, is a stable solution for the corresponding instance (G, w) of SRP, in line with Theorem 5. On the other hand, the Shapley value $\phi = (\frac{23}{30}, \frac{41}{30}, \frac{36}{30}, \frac{31}{30}, \frac{55}{30}, \frac{24}{30})$ is not a core allocation, as $v(\{4,5\}) = 3 > \frac{86}{30} = \phi(\{4,5\})$. ◇

Example 2.3. Let G' be the graph obtained from the graph G in Figure 2.6 after removing vertex 6. Unlike the matching game on G, the matching game on G' has an empty core. The reason is that the maximum weight $(5\frac{1}{2})$ of a matching in G'^d is larger than the maximum weight (5) of a matching in G', and we can apply Theorem 6 again. ◇

By Theorem 4, every b-assignment game is a yes-instance for P2 and P3 is polynomial-time solvable for b-assignment games. In contrast, Biró et al. [4] proved that P1 is coNP-complete even for uniform 3-assignment games.

On the positive side, Biró et al. [4] also showed that for $b \leq 2$, P1 (and thus P2 and P3) is polynomial-time solvable even for b-matching games. They did this

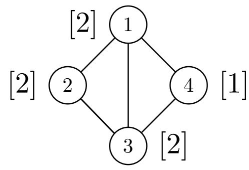

Figure 2.7: The graph G with $b(1) = b(2) = b(3) = 2$; $b(4) = 1$; and $w \equiv 1$ from Example 2.4.

by a reduction to the polynomial-time solvable *tramp steamer* problem. The latter problem is that of finding a cycle C in a graph G with maximum profit-to-cost ratio

$$\frac{p(C)}{w(C)},$$

where p and w represent profit and cost functions, respectively, that are defined on $E(G)$. However, there is a flaw in the argument of Biró et al. [4], which was later corrected by Sanità and Verberk [65].

Before considering partitioned matching games, it remains to discuss P2 and P3 for b-matching games for which $b \not\leq 2$. Theorem 5 implies that besides the aforementioned direct algorithms, any polynomial-time algorithm for SRP can also be used for immediately solving P2 and P3 for the corresponding matching game. As such, the following theorem for SFP, shown independently by Biró et al. [4] and Farczadi, Georgiou and Könemann [5], is initially promising for b-matching games.

Theorem 7 ([4,5])**.** SFP *can be solved in polynomial time. Moreover, if* (M, p) *is a stable solution for an instance* (G, b, w) *of* SFP*, then* M *is a maximum weight b-matching of* G *and* p^t *is a core allocation of the b-matching game defined on* (G, b, w).

By Theorem 7 we can first solve SFP for a given instance (G, b, w) and if a stable solution is found, then we immediately obtain a core allocation for the b-matching game defined on (G, b, w). However, Theorem 7 does not give us anything if (G, b, w) has no stable solutions. The following example given by Biró et al. [4] shows that the corresponding b-matching game may still have a core allocation, even if $b \leq 2$. Farczadi, Georgiou and Könemann [5] gave a different example for $b \leq 2$, where the core allocation even belongs to the pre-kernel.

Example 2.4. Consider the graph G with vertex capacity function b from Fig-

	P1	P2	P3
assignment game	poly	yes[1]	poly
matching game	poly (trivial)	poly[2]	poly
b-assignment game			
if $b \leq 2$	poly	yes	poly
if $b \leq 3$	coNPc[3]	yes[4]	poly[4]
b-matching game			
if $b \leq 2$	poly[3]	poly	poly
if $b \leq 3$	coNPc[3]	coNPh[5]	coNPh
partitioned matching game			
if $c \leq 2$	poly[5]	poly[5]	poly
if $c \leq 3$	coNPc[5]	coNPh[5]	coNPh

Table 2.2: Complexity dichotomies for the core (P1–P3), with the following shorthand notations, yes: all instances are yes-instances; poly: polynomial-time; coNPc: coNP-complete; and coNPh: coNP-hard. The seven hardness results in the table hold even if $w \equiv 1$. The references are [1] [7], [2] [9], [3] [4], [4] [8], [5] [2]. The unreferenced results follow from the referenced results. On a side node, partitioned permutation games are not included here but new results will be given in Chapter 6.

ure 2.7. We have $v(N) = 3$ in the corresponding uniform b-matching game; take $M = \{12, 23, 34\}$. Note that $(1, 1, 1, 0)$ is a core allocation. Biró et al. [4] proved that an instance of SFP has a stable solution if and only if the maximum weight of a b-matching in the underlying graph equals the maximum weight of a half-b-matching. Now, the maximum weight of a half b-matching in G is $3\frac{1}{2}$; take $f(12) = f(23) = 1$ and $f(13) = f(14) = f(34) = \frac{1}{2}$, while the maximum weight of a b-matching is $v(N) = 3$. So no stable solution for (G, b, w) exists. $\diamond$

From Example 2.4 we find that Theorem 5 cannot be generalized to SFP. Indeed, some time later, Biró et al. [2] proved that P2 and thus P3 are in fact coNP-hard even for uniform b-matching games if $b \leq 3$.

All the results for P1–P3 are summarized in Table 2.2. This table also displays a dichotomy for partitioned matching games. This dichotomy is obtained after establishing a close relationship between the core of partitioned matching game and the core of a b-assignment game, which we discuss below.

First, let (N, v) be a partitioned matching game of width c defined on an undirected graph $G = (V, E)$ with a positive edge weighting w and with a partition $(V_1, \ldots, V_n)$ of V. We assume that $c \geq 2$, as otherwise we obtain a matching game. Construct

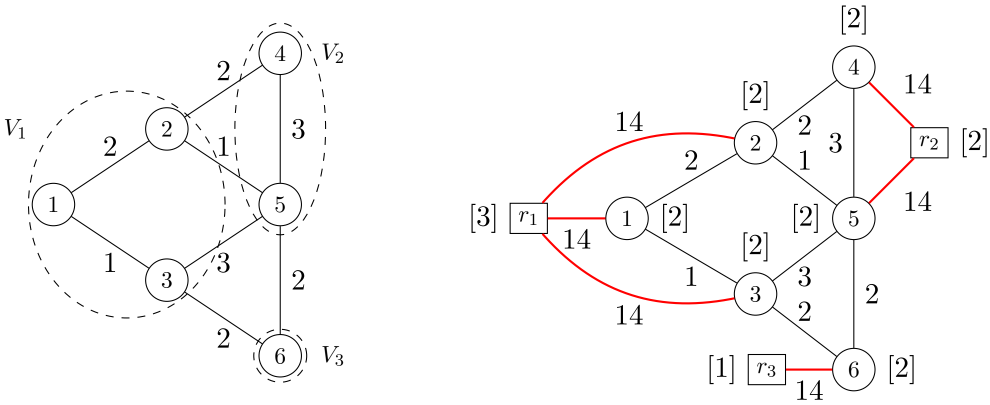

Figure 2.8: *Left:* the example from Figure 2.4: a partitioned matching game (N, v) with three players and with width $c = 3$. Recall that $v(N) = 7$. *Right:* the reduction to the b-matching game $(\overline{N}, \overline{v})$. Note that $|\overline{N}| = 9$ and that for every $i \in \overline{N}$, $b_i \leq c$.

a graph $\overline{G} = (\overline{N}, \overline{E})$ with a positive vertex capacity function b and a positive edge weighting $\overline{w}$ as follows.

- Put the vertices of V into $\overline{N}$ and the edges of E into $\overline{E}$.
- For each V_i, add a new vertex r_i to $\overline{N}$ that is adjacent to all vertices of V_i and to no other vertices in $\overline{G}$.
- Let $\overline{w}$ be the extension of w to $\overline{E}$ by giving each new edge weight $2v(N)$.
- In $\overline{N}$, give every $u \in V$, capacity $b(u) = 2$ and every r_i capacity $b(r_i) = |V_i|$.

We denote the b-matching game defined on $(\overline{G}, b, \overline{w})$ by $(\overline{N}, \overline{v})$. See Figure 2.8 for an example (based on Figure 2.4), and note that, by construction, we have $b \leq c$. Biró et al. [2] used the above construction to show the following theorem.[2]

Theorem 8 ([2])**.** P1–P3 *can be reduced in polynomial time from partitioned matching games of width c to b-matching games with $b \leq c$.*

As a consequence, P1–P3 are polynomial-time solvable for partitioned matching games of width $c \leq 2$; see also Table 2.2.

[2]In the paper of Biró et al. [2], Theorems 8 and 9 are explicitly proven for P2. The fact that they hold for P1 and P3 as well is implicit in their proofs; see the work of Biró et al. [2] for the details.

Now, let (N, v) be a b-matching game defined on a graph $G = (V, E)$ with a positive vertex capacity function b and a positive edge weighting w. Construct a graph $\overline{G} = (\overline{V}, \overline{E})$ with a positive edge weighting $\overline{w}$ and partition $\mathcal{V}$ of $\overline{V}$ by applying the aforementioned construction of Tutte [66].

- Replace each vertex $i \in V$ with capacity $b(i)$ by a set V_i of $b(i)$ vertices $i^1, \ldots, i^{b(i)}$.
- Replace each edge $ij \in E$ by a tree T_{ij} connecting the copies of i to the copies of j. The tree T_{ij} consists of a central edge with end-vertices i_j and j_i, where i_j is adjacent to all copies of i, and j_i is adjacent to all copies of j.
- Give every edge in T_{ij} weight $w(ij)$.
- Let $\overline{V}$ consist of the sets V_i and the 2-vertex sets $\{i_j, j_i\}$.

We denote the partitioned matching game defined on $(\overline{G}, w)$ with partition $\mathcal{V}$ by $(\overline{N}, \overline{v})$. See Figure 2.9 (based on Figure 2.3) for an example. We let b^* be the maximum $b(i)$-value for the vertex capacity function b. Note that, by construction, we have that $c = b^*$ and that uniformity is preserved (that is, if the original b-matching game is uniform, then the constructed partitioned matching game will be uniform as well). Biró et al. [2] used the above construction to show the following theorem.

Theorem 9 ([2])**.** P1–P3 *can be reduced in polynomial time from b-matching games to partitioned matching games of width $c = b^*$, preserving uniformity.*

As a consequence, P1 is coNP-complete and P2, P3 are coNP-hard, even for uniform partitioned matching games of width $c \leq 3$; see also Table 2.2.

2.6 Complexity Aspects of the Nucleolus

The sequence (LP_r) has length at most n (as each time the dimension of the feasible regions decreases). Nevertheless, computing the nucleolus takes exponential time in

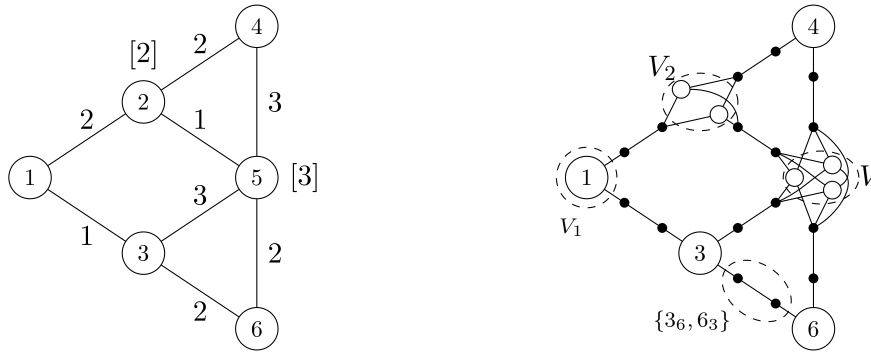

Figure 2.9: *Left*: the example from Figure 2.3: a b-matching game (N, v) with six players, where $b \equiv 1$ apart from $b(2) = 2$ and $b(5) = 3$, so $b^* = 3$. Recall that $v(N) = 10$ (take $M = \{12, 35, 45, 56\}$). *Right:* the reduction to the partitioned matching game $(\overline{N}, \overline{v})$. Note that $|\overline{N}| = 14$ and $c = b^*$.

general due to the computation of the sets $\mathcal{S}_i$ and the exponential number of inequalities in each linear program of the scheme of Maschler, Peleg and Shapley [44]. As shown below, this situation changes for matching games. Note that the nucleolus exists even for b-matching games and partitioned matching games, as the set of imputations for these games is non-empty. We first observe that the least core of matching games with a non-empty core (such as assignment games) has a compact description: for these games, we may restrict the set of constraints to only those for coalitions of size at most 2[3]. This immediately leads to a polynomial-time algorithm for the nucleolus using the ellipsoid method and the scheme of Maschler, Peleg and Shapley [44]. Solymosi and Raghavan [67] gave a faster algorithm, running in $O(n^4)$ time, for computing the nucleolus of assignment games. By using again the duplication technique of Nemhauser and Trotter Jr. [63], Biró, Kern and Paulusma [61] translated this algorithm into an $O(n^4)$-time algorithm for matching games with a non-empty core.

Faigle et al. [68] proved the following result for the nucleon of a matching game; the definition of the *nucleon* is obtained from the definition of the nucleolus by replacing (additive) excesses by multiplicative excesses $e'(S, x) = x(S)/v(S)$ in the definition of the nucleolus. In this result, the *size* of w is the number of bits in its

[3]If x is an allocation in this compact description, then $x(S) \geq \sum_{ij \in M}(x_i + x_j) \geq w(M) + \varepsilon = v(S) + \varepsilon$ for any coalition S (where M is a maximum weight matching of $G[S]$), and hence, the compact description describes the least core.

binary representation.

Theorem 10 ([68]). *The nucleon can be computed in time polynomial in* n *and the size of* w *for matching games.*

Faigle et al. [68] naturally asked if the same holds for the nucleolus. This led to a series of papers, which we discuss below.

Chen, Lu and Zhang [69] proved that the nucleolus of a fractional matching game can be found in polynomial time. Kern and Paulusma [70] gave a polynomial-time algorithm for uniform matching games; in fact this was shown for matching games with edge weights $w(ij)$ that can be expressed as the sum of positive vertex weights $w'(i)+w'(j)$ of the underlying graph G [71]. Afterwards, Farczadi [72] extended this result by relaxing the vertex weight condition. By combining techniques of Kern and Paulusma [70] and Biró, Kern and Paulusma [61], Hardwick [73] gave an $O(n^4)$-time combinatorial algorithm for computing the nucleolus of a uniform matching game that does not rely on the ellipsoid method. Finally, Könemann, Pashkovich and Toth [74] solved the open problem of Faigle et al. [68].

Theorem 11 ([74]). *The nucleolus can be computed in polynomial time for matching games.*

The proof of Theorem 11 is highly involved. For a least core allocation x, a matching M is *x-tight* if the incident vertices of M form a set S with $x(S) = v(S)+\varepsilon_1$ (so (x, ε) is an optimal solution for the least core, see also Section 2.3). A matching M is *universal* if it is x-tight for every least core allocation x. A least core allocation x is *universal* if the set of x-tight matchings is precisely the set of universal matchings; in particular the nucleolus is a universal allocation. To prove Theorem 11, Könemann, Pashkovich and Toth [74] first characterized universal matchings using classical descriptions of matching polyhedra. They used this characterization to obtain a new description of the set of universal allocations. They showed that this description is of a highly symmetric nature with respect to the least core allocations. This symmetric nature enabled them to give a new description of the least core that is sufficiently compact for computing the nucleolus in polynomial time, by using the aforementioned scheme of Maschler, Peleg and Shapley [44] and the ellipsoid method.

We now turn to b-matching games. Könemann, Toth and Zhou [75] proved that computing the nucleolus is NP-hard for uniform b-assignment games with $b \leq 3$. The same authors proved that the nucleolus can be found in polynomial time for two subclasses of b-assignment games with $b \leq 2$, one of which is the natural case where $b \equiv 2$, and the other one is where $b \equiv 2$ on one bipartition class, but $b \equiv 2$ on only a subset of vertices of constant size in the other bipartition class of the underlying graph. These results complement a result of Bateni et al. [76] who showed that the nucleolus of a b-assignment game can be found in polynomial-time if $b \equiv 1$ on one of the two bipartition classes of the underlying bipartite graph. Finally, Könemann and Toth [77] gave a polynomial-time algorithm for computing the nucleolus of uniform b-matching games on graphs of bounded treewidth.

Vazirani [78] proved that every matching game has a non-empty $\frac{2}{3}$-approximate core and that $\alpha = \frac{2}{3}$ is best possible. This implies that any coalition for matching games can gain at most $\frac{3}{2}$ factor by deviating and forming their own coalition. This result was extended to b-matching games by Xiao, Lu and Fang [79]. Polynomial-time algorithms for finding a $\frac{2}{3}$-approximate core allocation were given in both papers, but these allocations might not be the nucleon.

2.7 Complexity Aspects of the Shapley Value

Only a few complexity results are known for computing the Shapley value of matching games. Aziz and Keijzer [80] proved the following:

Theorem 12 ([80]). *Computing the Shapley value is #P-complete for uniform matching games.*

Aziz and Keijzer [80] also proved polynomial-time solvability for uniform matching games on graphs G if G has maximum degree at most 2 or if G allows a modular decomposition into k cliques for some constant k. They also proved the same results for matching games on graphs G with a positive edge weighting w if G allows a modular decomposition into k independent sets (sets consisting of pairwise non-adjacent vertices); for example, if G is a complete k-partite graph.

Moreover, Aziz and Keijzer [80] asked if there exists a (non-trivial) class of graphs that have at least one vertex of degree at least 3 for which the Shapley value of the corresponding matching game can be computed in polynomial time. This question was answered in the affirmative by Bousquet [81] who proved that the Shapley value can be computed in polynomial time for uniform matching games defined on trees.

Bousquet [81] asked the natural question whether his result for uniform matching games could be generalized from trees to graph classes of bounded treewidth. Greco, Lupia and Scarcello [82] showed that this was indeed possible.

CHAPTER 3

Simulation Setup

In this chapter we describe our simulation setup for $\ell = 2$, $\ell = \infty$ and $\ell = 3$ in detail. For consistency and comparisons of simulation results under different ℓs, all simulations for $\ell = 2$, $\ell = \infty$ and $\ell = 3$ are conducted using the same simulation setup. We now start with our simulation goals. For simulations, 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 between $\ell = 2$, $\ell = \infty$ and $\ell = 3$.

For our fourth aim, we are especially interested in the increase in the total number of patients treated when we compare the cases of $\ell = 2$, $\ell = \infty$ and $\ell = 3$. The simulation results are well presented in Chapter 4 for $\ell = 2$, Chapter 5 for $\ell = \infty$ and Chapter 6 for $\ell = 3$. In this chapter, we explain the simulation setup in Sections 3.1–3.4 and evaluation measures in Section 3.5, which is identical across all three different exchange bounds mentioned above.

3.1 The Set Up

To allow a fair comparison we use the same set up across all simulations. Additionally, we also use the same data from [83] for all simulations regarding $\ell = 2$, $\ell = \infty$ and $\ell = 3$. For all of our simulations, we take the same 100 compatibility graphs $G_1, \ldots, G_{100}$, each with roughly 2000 vertices from [83]. As real medical data is unavailable to us for privacy reasons, the data from [83] was obtained using the data generator from [38]. This data generator was used in many papers and is the most realistic synthetic data generator available; see also [84].

We use every G_i ($i \in \{1, \ldots, 100\}$) as the basis to perform simulations for n countries, where we let $n \in \{4, \ldots, 15\}$ for $\ell = 2$, $n \in \{4, \ldots, 10\}$ for $\ell = \infty$ and $n \in \{4, \ldots, 8\}$ for $\ell = 3$, due to Theorem 1. In order to do this, we first consider the case where all countries have equal size. That is, we partition $V(G_i)$ into the same n sets $V_{i,1}, \ldots, V_{i,n}$ 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 $2, \ldots, 24$. So, starting with $G_i^1(n)$, we run an IKEP of 24 rounds in total. This gives us 24 compatibility graphs $G_i^1(n), \ldots, G_i^{24}(n)$. 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. We consider the same 100 compatibility graphs $G_1, \ldots, G_{100}$ as in simulations with equal country sizes; the only difference is that we now partition each $V(G_i)$ into the same n sets $V_{i,1}, \ldots, V_{i,n}$, 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 $G_i^2(n), \ldots, G_i^{24}(n)$

(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 3.2); and

(iv) a choice of type of optimal solution (see Section 3.3).

Our code for obtaining the simulation instances can be found in GitHub repositories [83] for $\ell = 2$ and [85] for $\ell = \infty$ and $\ell = 3$, along with the compatibility graphs data and the seeds for the randomization. The latter two are also available in [86].

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

3.2 The Initial Allocations

For the initial allocations y we use six known solution concepts, along with the Banzhaf* value, all of which are listed below:

- the Shapley value,
- the Banzhaf value,
- the nucleolus,
- the tau value,
- the benefit value
- the contribution value, and
- the Banzhaf* value (which, we recall, is a variant of the Banzhaf value in the credit-adjusted game; see Chapter 2).

These solution concepts are defined in Chapter 2. Each of them prescribes exactly one allocation for an IKEP associated with the compatibility graphs.

For computing these allocations we need the v-values of partitioned matching games, partitioned permutation games and partitioned 3-permutation games. Recall that, we can compute a single value $v(S)$ in polynomial time for both $\ell = 2$ and $\ell = \infty$ by Theorem 1 through solving a maximum matching problem in the former case and in the latter, by a transformation into a bipartite graph and solving a maximum weight perfect matching problem (see Chapter 1). For $\ell = 3$, guided by Theorem 1, we use ILPs (see Chapter 6) to compute a single value $v(S)$.

3.3 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.

By Theorem 1 (in Chapter 1), computing an arbitrary optimal solution is polynomial-time solvable for $\ell = 2$ or $\ell = \infty$ and NP-hard otherwise. Therefore, we do as follows. For computing an arbitrary optimal solution (arbitrary maximum matching) for $\ell = 2$ we use one given to us by the package of [87]. For computing a weakly close optimal solution for $\ell = 2$, it suffices to perform only the first step of `Lex-Min`, which is a polynomial-time algorithm for computing strongly close optimal solutions for $\ell = 2$, as explained in Chapter 4.

For computing arbitrary optimal solutions for $\ell = \infty$, we use the package of [87], while for $\ell = 3$, we compute them by solving ILPs (see Chapter 6). We compute the weakly and strongly close optimal solutions by solving a sequence of ILPs for $\ell = \infty$ and $\ell = 3$, as described in Chapters 5 and 6 respectively. 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. We recall that the values $v(N)$ of the partitioned 3-permutation games and partitioned permutation games used in the ILPs (see Chapters 5 and 6) have already been computed for the initial allocations, and we simply reuse them.

3.4 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 ("d_1") 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:

- *arbitary*,
- *d1*,
- *d1+c*,
- *lexmin*,
- *lexmin+c*;

plus two scenarios for the Banzhaf* value:

- *d1+c*,
- *lexmin+c*,

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 have twelve different country sizes n for $\ell = 2$, seven different ns for $\ell = \infty$ and five different ns for $\ell = 3$. Additionally, we have 100 compatibility graphs G_i for all three different exchange bounds. Hence, the scale of our experimental study is very large. Namely, the total number of 24-round simulation instances is as follows:

$$\ell = 2 : 32 \times 2 \times 12 \times 100 = 76800,$$

$$\ell = \infty : 32 \times 2 \times 7 \times 100 = 44800,$$

$$\ell = 3 : 32 \times 2 \times 5 \times 100 - 2 = 31998.$$

On a side note, as mentioned in Chapter 2, the benefit value and contribution value do not exist when the corresponding denominator happens to be zero. This does not happen for the benefit value in the setting where all countries have the same size, nor does it occur for the contribution value. However, this case occurs only twice out of 16000 for the benefit value in the varying country size for $\ell = 3$.

Simulations for $\ell = 2$ were run on a desktop PC with AMD Ryzen 9 5950X 3.4 GHz CPU and 128 GB of RAM, running on Windows 10 OS and C++ implementation in Visual Studio (the code is available in GitHub repository [83]). For other simulations for $\ell = \infty$ and $\ell = 3$ 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 1 core, 1 thread and 1G temporary disk space.

3.5 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 $\mathcal{C}^*$ be the union of the chosen maximum cycle packings in each of the 24 rounds. We use the *total relative deviation* defined as

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

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*

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

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

CHAPTER 4

Simulations and Results for $\ell = 2$

In this chapter, we present our algorithm for finding a strongly close optimal solution for $\ell = 2$, and show our simulation results.

In Section 4.1 we describe the algorithm, called `Lex-Min` [2], that we used for computing maximum matchings that lexicographically minimize the country deviations from a given target allocation. As we will explain, this algorithm can also be used for computing maximum matchings that minimize only the largest country deviation from a given target allocation. For the correctness proof and a running time analysis of the algorithm we refer to [2]. Using the `Lex-Min` algorithm, all simulations for ℓ are solved to optimality. Unlike cases of $\ell = \infty$ in Chapter 5 and $\ell = 3$ in Chapter 6, no ILPs are required for computing weakly or strongly close optimal solutions for $\ell = 2$.

In Section 4.2 we present the results of our simulations. As mentioned, we conduct simulations for up to as many as 15 countries in contrast to the previous studies [10, 11] for 3–4 countries. Moreover, we do this both for equal and varying country sizes, and across a wide range of solution concepts, significantly more than 2 solution concepts used in previous studies [10, 11]. Namely, our target allocations are prescribed by four hard-to-compute solution concepts: the Shapley value, nucleolus, Banzhaf value and tau value[1] and two easy-to-compute solution concepts: the aforementioned benefit value, which coincides with the Gately point if the latter is unique, and a natural variant of the benefit value, the contribution value. As mentioned, we defined all these concepts in Chapter 2.

Our simulations show that a credit system using lexicographically minimal maximum matchings instead of ones that minimize only the largest country deviation from a given target allocation makes an IKEP up to 54% more balanced, without decreasing the overall number of transplants. The exact improvement depends on which solution concept is used. From our experiments, both the Banzhaf and Shapley value yield the best results, namely, on average, a deviation of up to 0.52% from the target allocation. However, the differences between the different solution concepts are small: all the other solution concepts stay within 1.23% of the target allocation, and the choice for using a certain solution concept will be up to the policy makers of the IKEP.

We finish our simulations by examining a new approach for incorporating credits that has not been proposed in the literature before. Namely, it is also natural to let the solution concepts prescribe an allocation for a *credit-adjusted* game, where the credits are incorporated into the value function of the game directly. As explained in Chapter 2, where we introduce this approach after first describing the original model, only the Banzhaf value may prescribe different allocations. For all the other solution concepts that we consider both the original and new credit system yield exactly the same target allocations. Our simulations show, however, that the Banzhaf* value yields, on average, a relative deviation up to 0.48% from the target allocation. This is a slight improvement over the best results (0.52%) under the original credit system.

In Section 4.3 we evaluate some other aspects of our simulations. First, we show that IKEPs lead to a significantly larger number of total kidney transplants than the total number of transplants of the KEPs of the individual countries. Second, we show that, although theoretically country credits may build up over time as illustrated with an example in Chapter 1, this situation does not happen in any of

[1]In 0.04% of our simulations, the tau value does not exist and in those cases we replace the tau value by the closely related benefit value.

our simulations. Third, we evaluate computational time issues in our simulations, showing that the generation of the partitioned matching games is the most expensive operation in our simulations. Fourth, we evaluate a number of game-theoretic properties: core stability aspects, convexity and quasibalancedness.

4.1 Computing a Lexicographically Minimal Maximum Matching

Let (N, v) be a partitioned matching game ($\ell = 2$) with a set V of patient-donor pairs. Recall that, for computing optimal solutions if $\ell = 2$, the problem can be reduced to finding a maximum matching M in an undirected graph. Let $\mathcal{M}$ be the set of maximum matchings in the corresponding compatibility graph D, and let x be an allocation. In this section we will give our algorithm `Lex-Min` that we use for computing a maximum matching from $\mathcal{M}$ that is lexicographically minimal for x, namely the strongly close optimal solution for partitioned matching games. This algorithm computes for a partitioned matching game (N, v) and allocation x, strictly decreasing values $d_1, \ldots, d_t$ for some integer $t \geq 1$, and returns a matching $M \in \mathcal{M}$ that is lexicographically minimal for x. For computing $d_1, \ldots, d_t$, the algorithm calls the polynomial-time algorithm provided by the following lemma from [2].

Lemma 1 ([2]). *Given a partitioned matching game (N, v) on a graph $G = (V, E)$ with a positive edge weighting w and and partition $\mathcal{V}$ of V, and intervals $I_1, \ldots, I_n$, it is possible in $O(|V|^3)$-time to decide if there exists a matching $M \in \mathcal{M}$ with $s_p(M) \in I_p$ for $p \in \{1, \ldots, n\}$, and to find such a matching (if it exists).*

`Lex-Min`

input : a partitioned matching game (N, v) and an allocation x

output : a matching $M \in \mathcal{M}$ that is lexicographically minimal for x.

Step 1. Compute the smallest number $d_1 \geq 0$ such that there exists a matching $M \in \mathcal{M}$ with $|x_p - s_p(M)| \leq d_1$ for all $p \in N$.

Step 2. Compute a minimal set $N_1 \subseteq N$ (with respect to set inclusion) such that there exists a matching $M \in \mathcal{M}$ with

$$|x_p - s_p(M)| = d_1 \qquad \text{for all } \; p \in N_1$$
$$|x_p - s_p(M)| < d_1 \qquad \text{for all } \; p \in N \setminus N_1.$$

Step 3. Proceed in a similar way for $t \geq 1$:

- **while** $N_1 \cup \cdots \cup N_t \neq N$ **do**
  - $t \leftarrow t + 1$.
  - $d_t \leftarrow$ smallest d such that there exists a matching $M \in \mathcal{M}$ with

$$|x_p - s_p(M)| = d_j \qquad \text{for all } \; p \in N_j, \; j \leq t-1$$
$$|x_p - s_p(M)| \leq d_t \qquad \text{for all } \; p \in N \setminus (N_1 \cup \cdots \cup N_{t-1}).$$

  - $N_t \leftarrow$ inclusion minimal subset of $N \setminus (N_1 \cup \cdots \cup N_{t-1})$ such that there exists a matching $M \in \mathcal{M}$ with

$$|x_p - s_p(M)| = d_j \qquad \text{for all } \; p \in N_j, \; j \leq t-1$$
$$|x_p - s_p(M)| = d_t \qquad \text{for all } \; p \in N_t$$
$$|x_p - s_p(M)| < d_t \qquad \text{for all } \; p \in N \setminus (N_1 \cup \cdots \cup N_t).$$

Step 4. Return a matching $M \in \mathcal{M}$ with $|x_p - s_p(M)| = d_j$ for all $p \in N_j$ and all $j \in \{1, \ldots, t\}$.

We say that the countries in a set $N \setminus (N_1 \cup \cdots \cup N_{t-1})$ are *unfinished* and that a country is *finished* when it is placed in some N_t. Note that `Lex-Min` terminates as soon as all countries are finished. For a correctness proof and running time analysis of Lex-Min we refer to [2].

Theorem 13 ([2]). *The* `Lex-Min` *algorithm is correct and runs in* $O(n|V|^3 \log |V|)$ *time for a partitioned matching game* (N, v) *with an allocation* x.

Note that `Lex-Min` can also be used for computing a maximum matching M that minimizes the maximum deviation d_1 from an allocation x and as such does not need to be lexicographically minimal.

4.2 Simulation Results

4.2.1 Simulation Results for Equal Country Sizes

In Figure 4.1 we display the main results for the situation where all countries have the same size and when we use the average total relative deviation as our evaluation measure. As expected, using an arbitrary maximum matching in each round makes the kidney exchange scheme significantly more unbalanced, with average total relative deviations above 13.8% for all initial allocations y.

From Figure 4.1 we can compute the *relative improvement* of *lexmin+c* over *d1+c*. For example, for $n = 15$, this percentage is $(2.05 - 0.97)/2.05 =$ **52.49**% for the tau value, whereas for the other solution concepts it is 45.5% (nucleolus); 44% (benefit value); 41% (contribution value); 40% (Shapley value); and 40% (Banzhaf value). Considering the average improvement over $n \in \{4, \dots, 15\}$ yields percentages of 37% (tau value); 36% (nucleolus); 31% (benefit value); 30% (Shapley value); 27% (Banzhaf value); and 24% (contribution value).

From Figure 4.1 we can also compare *lexmin+c* with *lexmin*, and *d1+c* with *d1*. We see that using c has a substantial effect. Whilst *lexmin* ensures that allocations stay close to the target allocations, the role of c is to keep the deviations small and to guarantee fairness for the participating countries over a long time period.

Our main conclusion from Figure 4.1 is that using *lexmin+c* yields the lowest average total relative deviation for all six initial allocations y, with larger differences when the number of countries is growing. In Figure 4.2 we displayed the six *lexmin+c* graphs of Figure 4.1 in one plot in order to compare them with each other. As mentioned, the choice for initial allocation is up to the policy makers of the IKEP. However, from Figure 4.2 we see that the Shapley value and the Banzhaf value in the *lexmin+c* scenario consistently provides the smallest deviations from the target allocations (**0.52%** for $n = 15$), while the contribution value for $n \leq 12$ and the

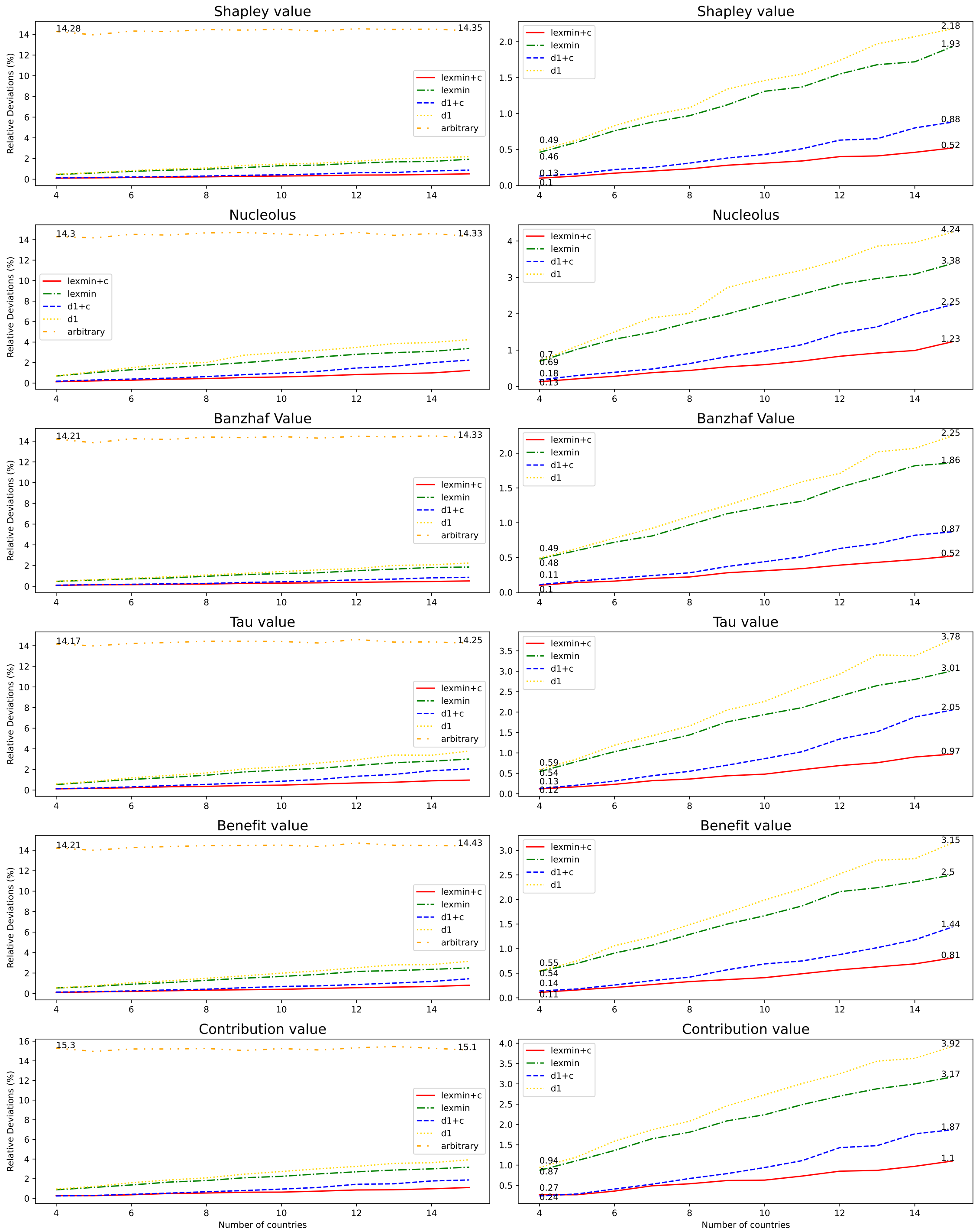

Figure 4.1: Average total relative deviations for the situation where all countries have the same size. The number of countries n is ranging from 4 to 15. The figures on the right side zoom in on the figures from the left side by removing the results for the arbitrary matching scenario.

nucleolus for $n \geq 13$ perform the worst. The latter result is perhaps somewhat surprising given the sophisticated nature of the nucleolus. Figure 4.3 displays the six *d1+c* in one plot, and we could see that the Banzhaf* value and Shapley value consistently yield the lowest average relative deviations, which is in line with the same plot under *lexmin+c* in Figure 4.2.

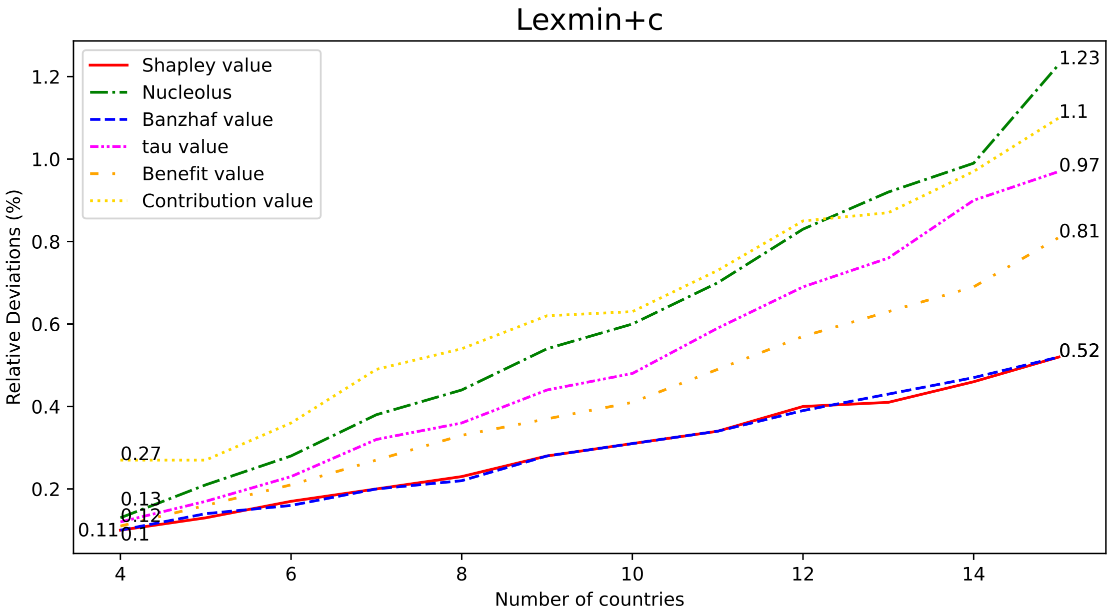

Figure 4.2: Displaying the six *lexmin+c* graphs of Figure 4.1 in one plot (same country sizes).

We now turn to the Banzhaf value for the credit-adjusted games, which we denote as the Banzhaf* value. Recall that choosing any of the other solution concepts as initial allocation will lead to the same results as for the original games, and only for the Banzhaf value the two different credit systems may give different results. Figure 4.4 shows that the latter is indeed the case. It displays the *lexmin+c* graphs for the Shapley value and (original) Banzhaf value from Figure 4.1 and compares them with the *lexmin+c* graph for the Banzhaf* value. Figure 4.5 does the same for the three *d1+c* graphs. Both figures show that the Banzhaf* value behaves better than the Shapley value and Banzhaf value; however the differences are very small (at most 0.04% for lexmin+c and 0.19% for d1+c).

If we use our second evaluation measure, the average maximum relative deviation,

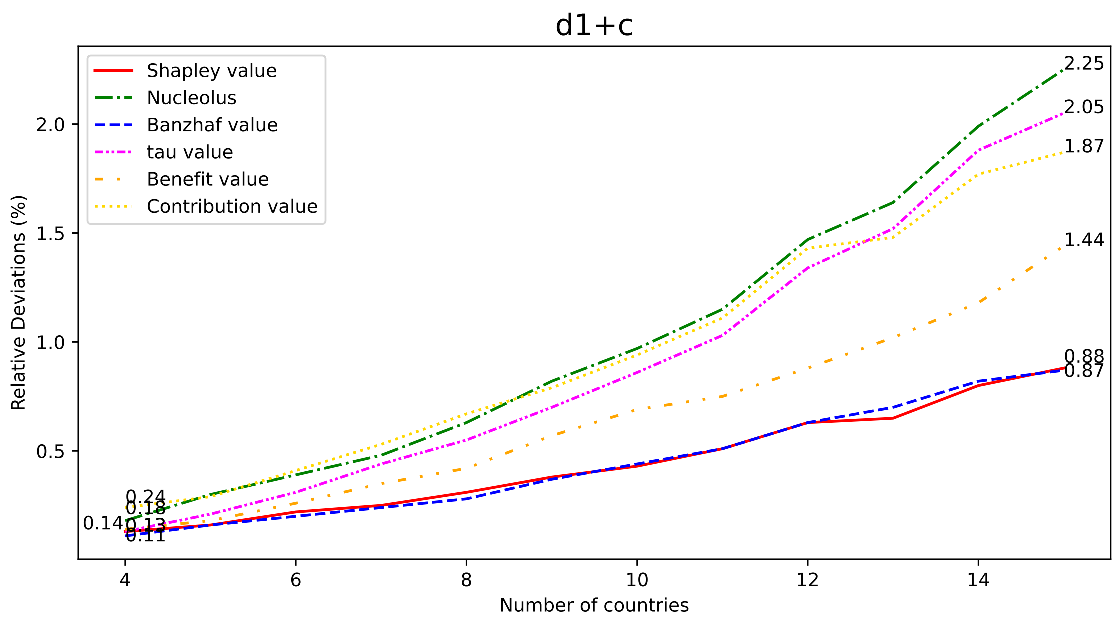

Figure 4.3: Displaying the six *d1+c* graphs of Figure 4.1 in one plot (same country sizes).

then we obtain similar results and can draw the same conclusions; see Figures 4.6–4.10 for equal country sizes.

4.2.2 Simulation Results for Varying Country Sizes

We now turn to the situation of varying country sizes and perform the same simulations as before. We can draw exactly the same conclusions with different percentages and therefore do not perform additional simulations for varying country sizes. That is, from Figure 4.11 we see that using an arbitrary maximum matching in each round makes the kidney exchange scheme significantly more unbalanced, with average total relative deviations above 8.15% for all initial allocations y. Moreover, from Figure 4.11 we can also compute the *relative improvement* of *lexmin+c* over *d1+c*. For example, for $n = 15$, this percentage is $(1.13 - 2.45)/2.45 =$ **54**% for the nucleolus, whereas for the other solution concepts it is 53% (contribution value); 49% (tau value); 48% (benefit value); 46% (Banzhaf value); and 38% (Shapley value). Considering the average improvement over $n = 4, \ldots, 15$ now yields percentages of 44% (contribution value); 41% (nucleolus); 35% (tau value); 32% (benefit value);

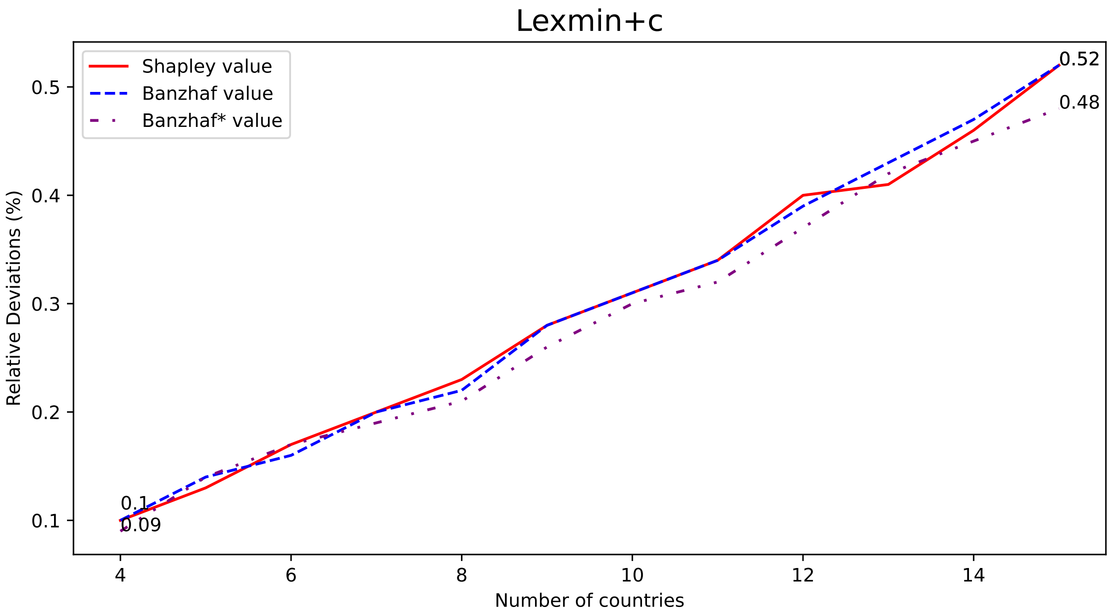

Figure 4.4: Comparing the *lexmin+c* graphs for the Shapley value and Banzhaf value from Figure 4.1 with the one for the Banzhaf* value (same country sizes).

25% (Shapley value); and 25% (Banzhaf value). Compare *lexmin+c* with *lexmin*, and *d1+c* with *d1*, we see again that using c has a substantial effect. Our main conclusion from Figure 4.11 is again that using *lexmin+c* yields the lowest average total relative deviation for all six initial allocations y, with larger differences when the number of countries is growing. However, from Figure 4.12 we see again that the Shapley value and the Banzhaf value in the *lexmin+c* scenario consistently provides the smallest deviations from the target allocations (**0.55%** and **0.54%** for $n = 15$), while the contribution value for $n \leq 13$ and the nucleolus for $n \geq 14$ perform the worst. Figure 4.13 displays the six *d1+c* in one plot, and we could see that the Banzhaf* value and Shapley value consistently yield the lowest maximum relative deviations again, which is in line with the same plot under *lexmin+c* in Figure 4.12.

Turning now to the credit-adjusted games and the Banzhaf* value, the situation with only the Banzhaf value producing different results among the tested allocations, and only in scenarios *lexmin+c* and *d1+c* (since without credits the games are the same), naturally remains the same. However, as shown in Figures 4.14 and 4.15,

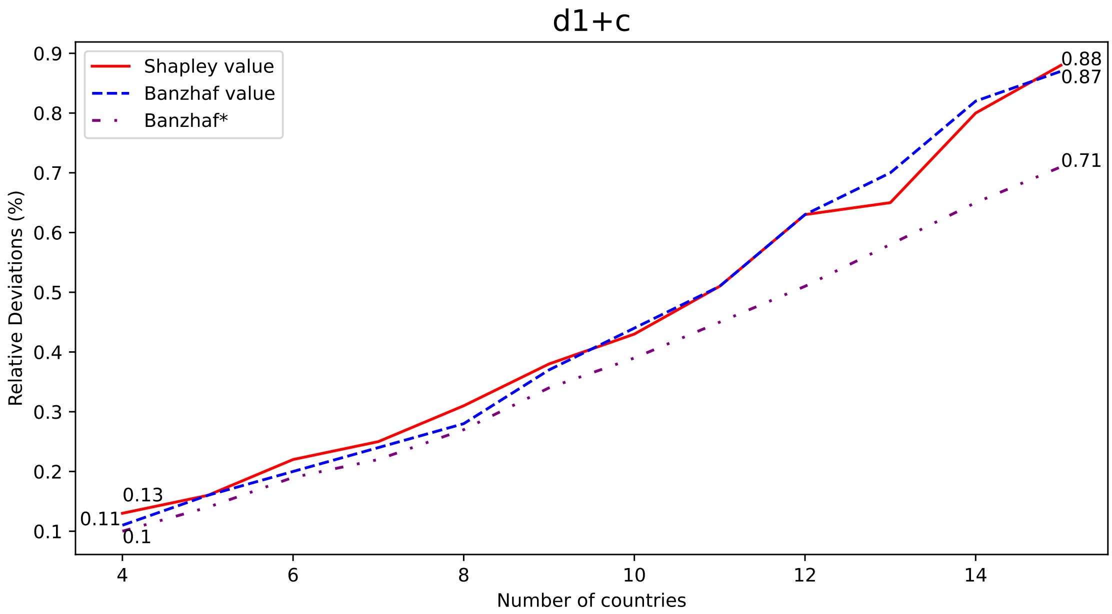

Figure 4.5: Comparing the *d1+c* graphs for the Shapley value and Banzhaf value from Figure 4.1 with the one for the Banzhaf* value (same country sizes).

the behaviour of Banzhaf* value differs under the varying country sizes, in one part performing slightly worse than the Shapley value and the original Banzhaf value for $n \leq 14$ (the differences are again very small: within 0.08% and 0.11% for lexmin+c and d1+c respectively), but outperforming both for $n = 15$, by at most 0.05% for lexmin+c and 0.3% for d1+c. Overall, as shown in Figure 4.16, the behavior of Banzhaf* value in terms of average total relative deviations for the same and varying country sizes are quite consistent across five scenarios.

If we use the average maximum relative deviation instead of the average total relative deviation, then again we obtain similar results and can draw the same conclusions; see Figures 4.17–4.21 for the corresponding figures.

4.3 Evaluation of Further Aspects

In this section we evaluate some other aspects of our simulations. Given that the results of our simulations for varying countries were similar to the results of our simulations for same country sizes, we only evaluated these aspects for the situation, in which all countries have the same size.

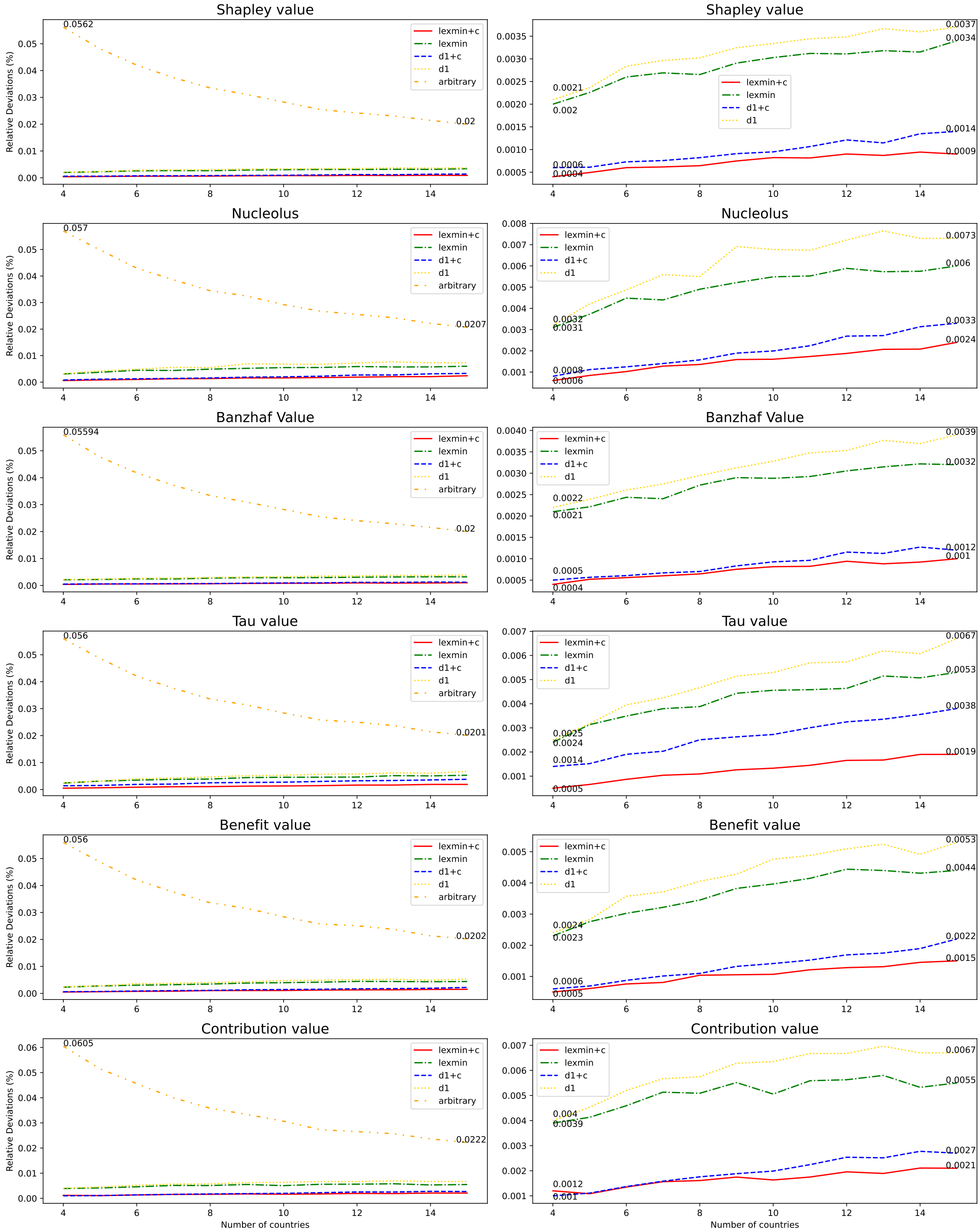

Figure 4.6: Average maximum relative deviations for the situation where all countries have the same size. The number of countries n is ranging from 4 to 15. The figures on the right side zoom in on the figures from the left side by removing the results for the arbitrary matching scenario.

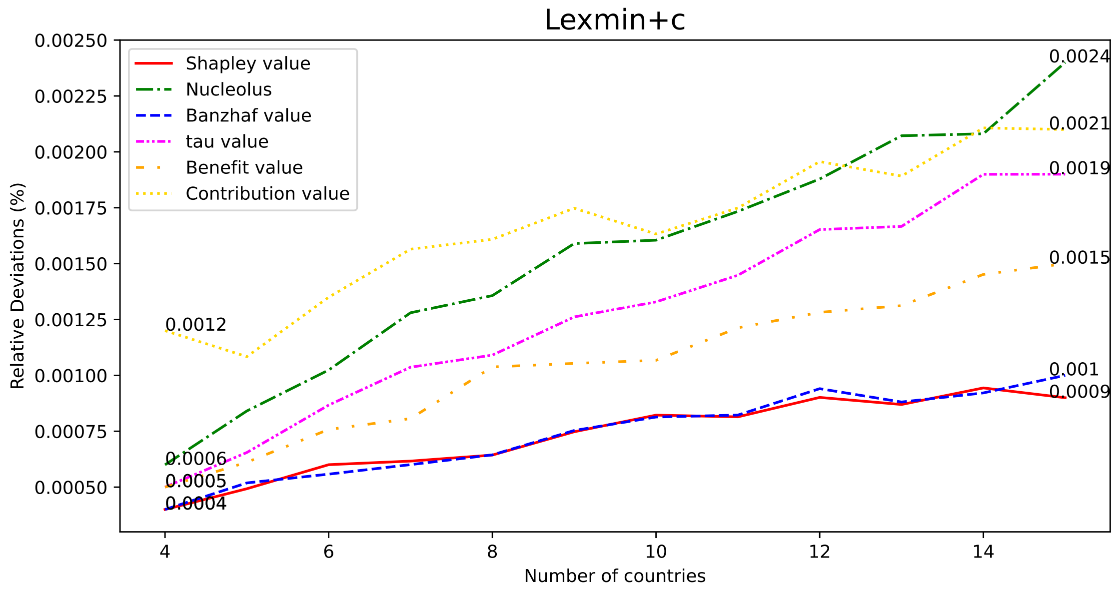

Figure 4.7: Displaying the six *lexmin+c* graphs of Figure 4.6 in one plot (same country sizes).

4.3.1 No Cooperation

It is a natural question to what extent cooperation between countries helps. Table 4.1 shows that cooperation leads to a significantly larger number of total kidney transplants than non-cooperation. This is especially the case when more and more countries are participating in the IKEP. In particular, Table 4.1 shows that a gain of 2.86 times as many kidney transplants can be obtained in the case where the number of countries is 15. Even when $n = 4$, the number of transplants in IKEPs is more than 1.08 times that of the non-cooperation case. In a previous study [11] for $\ell = 3$, a similar gain, 1.08 times the non-cooperation outcome, was observed when $n = 3$. Therefore, Table 4.1 provides strong evidence for forming large IKEPs.

We also note the following. Theoretically, a change in scenario may result in a change in the maximum matching size (total number of kidney transplants). However, Table 4.1 shows that these differences turn out to be negligible (between 0.01% and 0.1% on average).

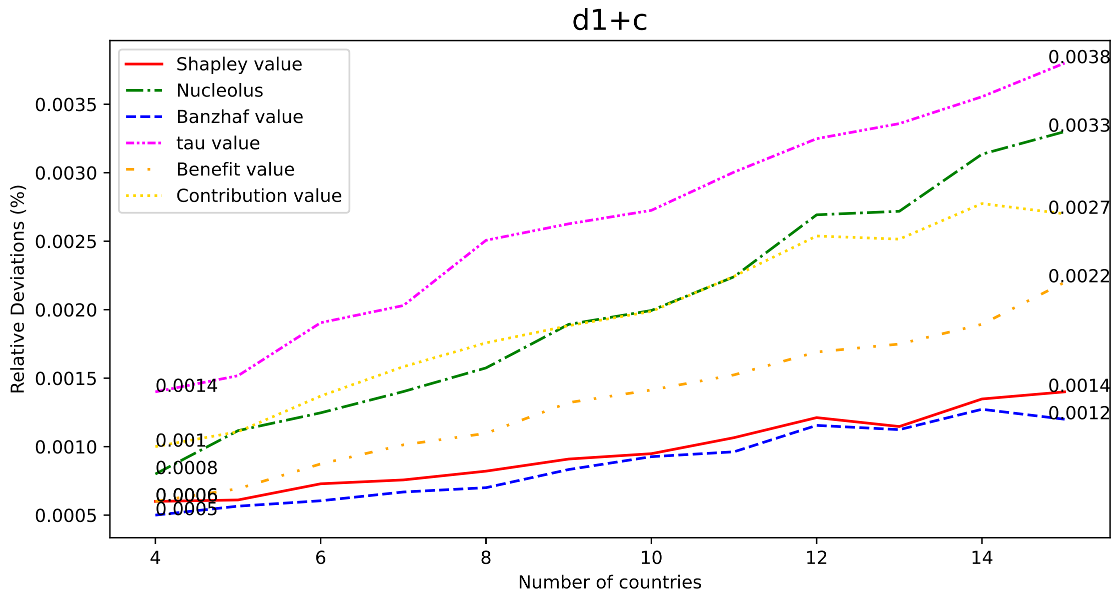

Figure 4.8: Displaying the six *d1+c* graphs of Figure 4.6 in one plot (same country sizes).

4.3.2 Credit Accumulation

In Chapter 1, we gave a theoretical example where credits build up over time for a certain country and are essentially meaningless. However, this behaviour did not happen in any of our 24-round simulations. We performed for every number n of countries with $n \in \{4, \dots, 15\}$, a refined analysis, just to verify if such behaviour could be expected if the number of rounds is larger than 24.

First recall that for a single instance, $c_p^h = x_p^h - y_p^h$ and also that $c_p^h = \sum_{t=1}^{h-1}(y_p^t - s_p(M^t))$ for country p and round $h \geq 2$. That is, credits are the difference between the initial and target allocations in each round, as well as the accumulation of the deviations from the initial allocations. The latter is the accumulation we would like to avoid from happening by using the credit function c. In order to assess credit accumulation over time, we define the *average accumulated deviation* at round h as the average of $\sum_{p \in N} |c_p^h|$ over the 100 instances corresponding to a certain scenario and choice of initial allocation.

In Figure 4.22 we show the results of our analysis. The results displayed are only for $n = 15$, as the figures for $n \in \{4, \dots, 14\}$ turned out to be very similar. As

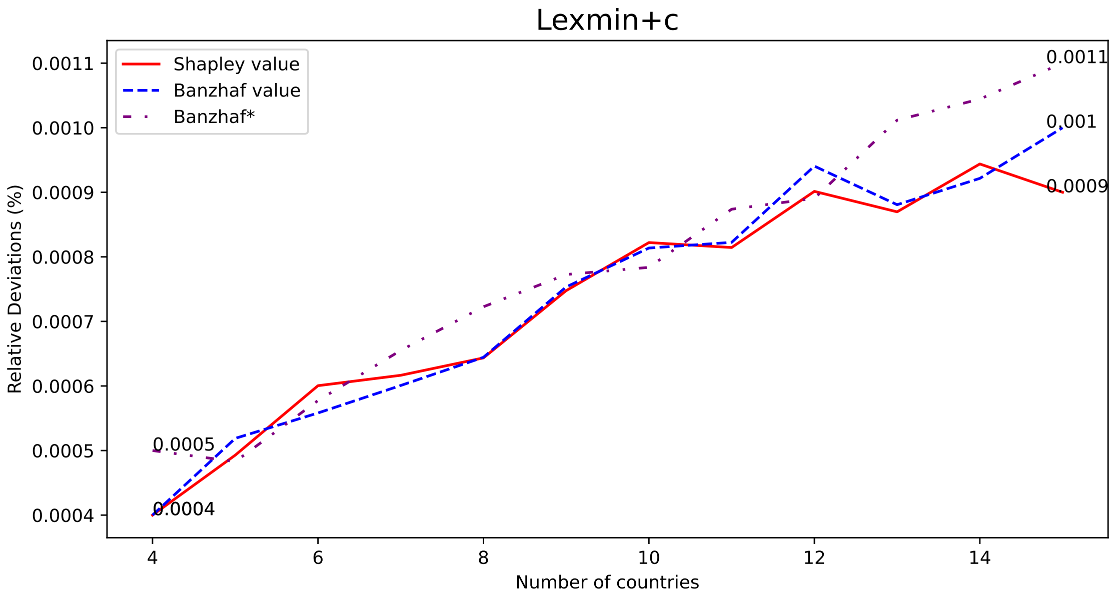

Figure 4.9: Comparing the *lexmin+c* graphs for the Shapley value and Banzhaf value from Figure 4.6 with the one for the Banzhaf* value (same country sizes).

Figure 4.22 shows, the behaviour of credit accumulation is similar for each choice of initial allocation y. Moreover, as expected, the average total deviation is clearly accumulating over time if arbitrary maximum matchings are chosen as solutions (see the left side figures of Figure 4.22). Under *lexmin* and *d1* there is still accumulation. However the credit system is indeed successfully mitigating against this effect, as the plots for *lexmin+c* and *d1+c* show (see the right side figures of Figure 4.22). In particular, there is no indication that this behaviour will change if the number of rounds is larger than 24.

4.3.3 Computational Time

We refer to Table 4.2 for an overview of various computational times by our simulations. We note that `Lex-min` computes at most $n-1$ d_i-values, and in our experiments we actually found instances where d_{n-1} was computed even for $n=10$. However, Table 4.2 shows that using lexicographically minimal maximum matchings instead of ones that only minimize the largest deviation d_1 from the target allocation does not require a significant amount of additional computation time. It

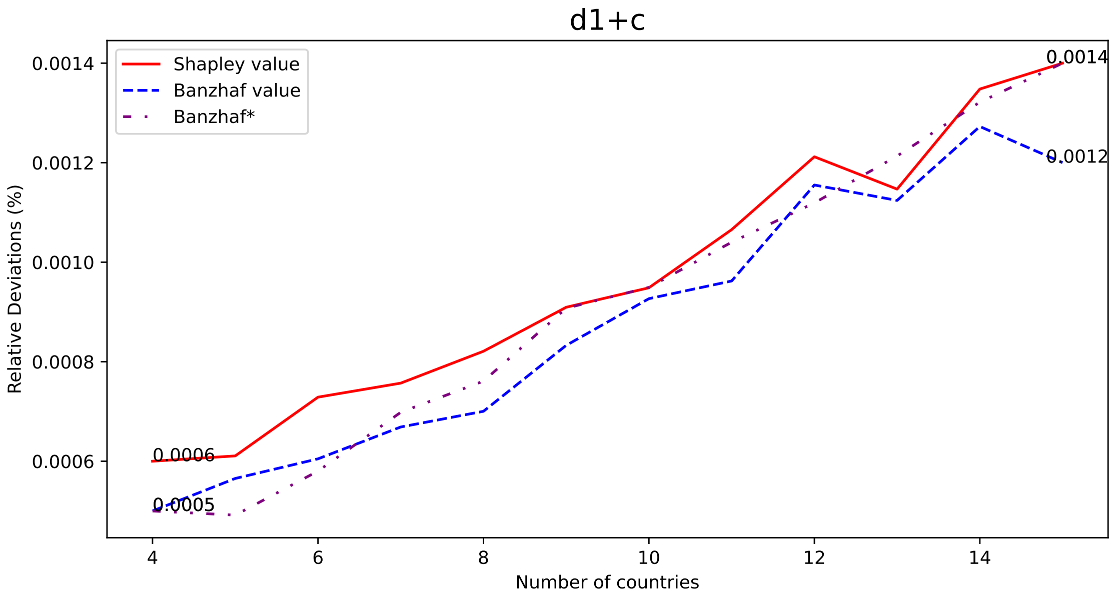

Figure 4.10: Comparing the *d1+c* graphs for the Shapley value and Banzhaf value from Figure 4.6 with the one for the Banzhaf* value (same country sizes).

can also be noticed from Table 4.2 that, as expected, computing the Shapley value and the nucleolus is more expensive than computing the contribution value and the benefit value, especially as the number of countries n is growing. Finally, we see from Table 4.2 that the *game generation*, that is, computing the $2^n - 1$ values $v(S)$, becomes by far the most expensive part when n is growing.

4.3.4 Coalitional Stability

In order to assess the long-term coalitional stability of an IKEP, we turn our focus towards the core of the accumulated partitioned matching games. These games are obtained by summing up the 24 partitioned matching games of each of the 24 rounds of a simulation instance. That is, the *accumulated partitioned matching game* (N, v) is obtained from the partitioned matching games (N, v^h) on compatibility graphs D^h for $h = 1, \ldots, 24$ by setting $v = \sum_{h=1}^{24} v^h$. We define the *accumulated initial allocation* as $y = \sum_{h=1}^{24} y^h$ and the *accumulated solution* as the accumulated number of kidney transplants $s = \sum_{h=1}^{24} s(M^h)$, where M^h is the chosen matching in round h.

All accumulated partitioned matching games in our simulation had a nonempty

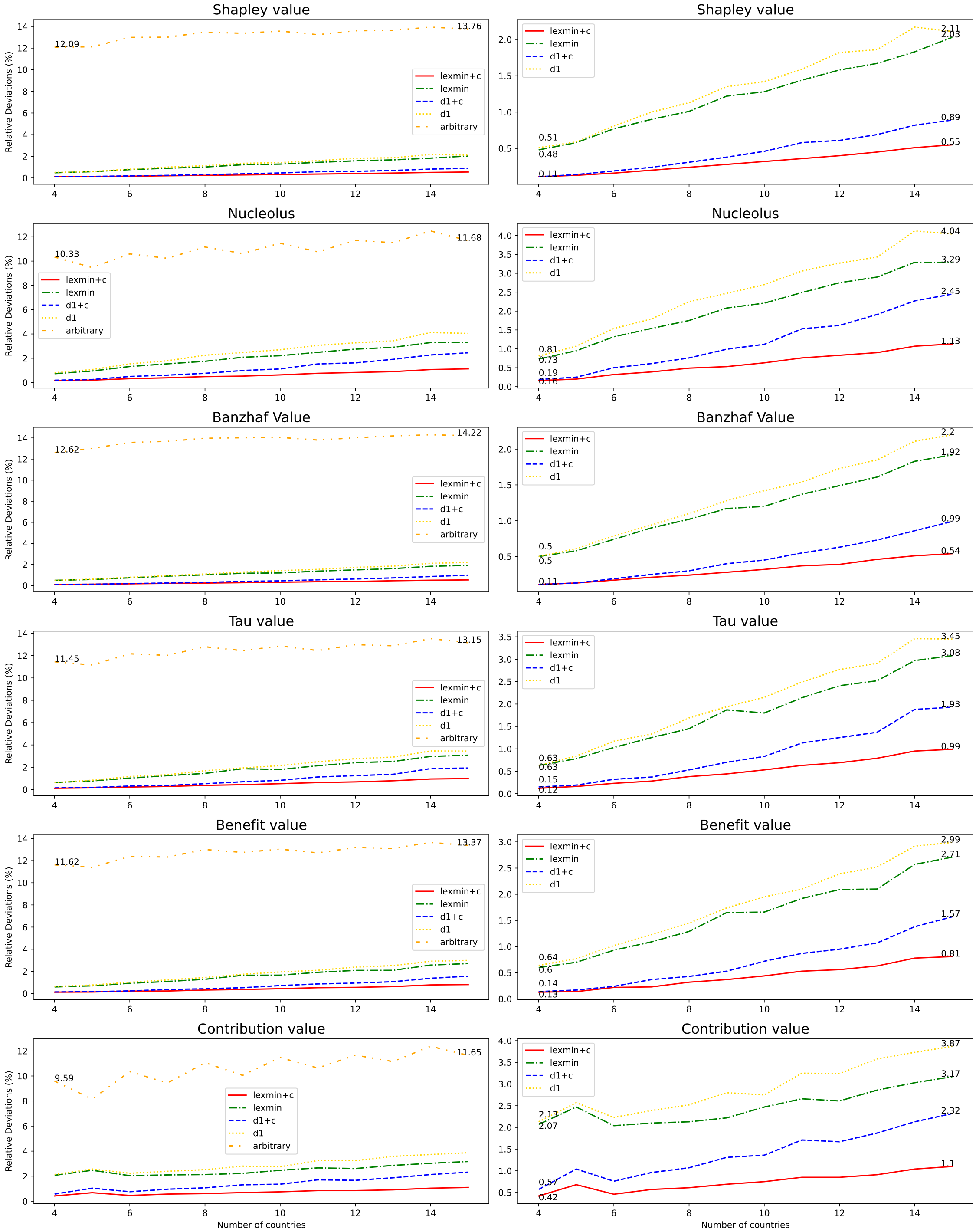

Figure 4.11: Average total relative deviations for the situation where the countries vary in size. The number of countries n is ranging from 4 to 15. The figures on the right side zoom in on the figures from the left side by removing the results for the arbitrary matching scenario.

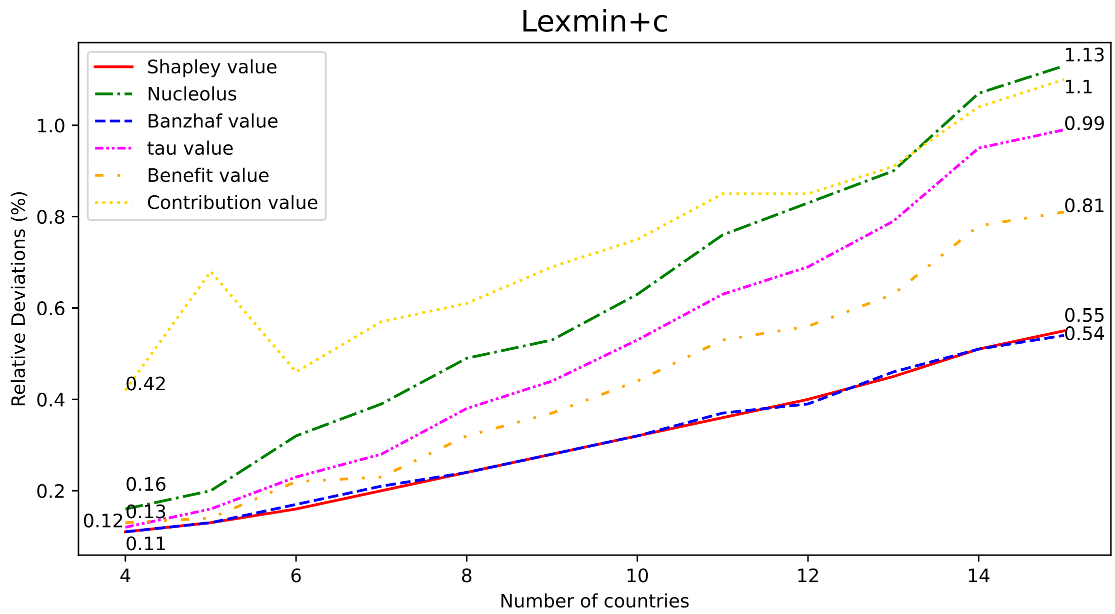

Figure 4.12: Displaying the six *lexmin+c* graphs of Figure 4.11 in one plot (varying country sizes).

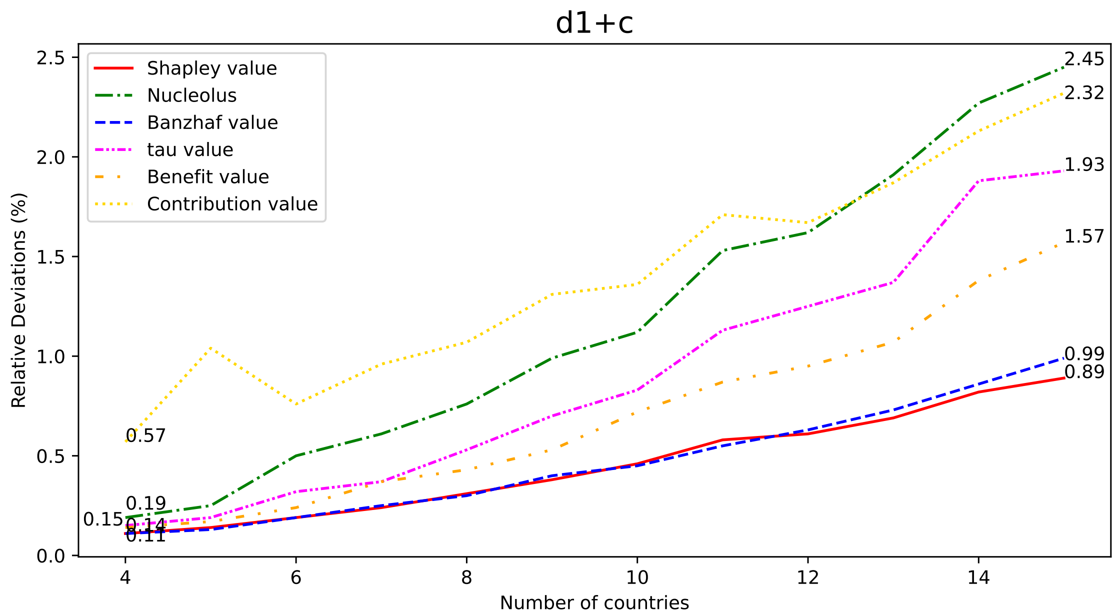

Figure 4.13: Displaying the six *d1+c* graphs of Figure 4.11 in one plot (varying country sizes).

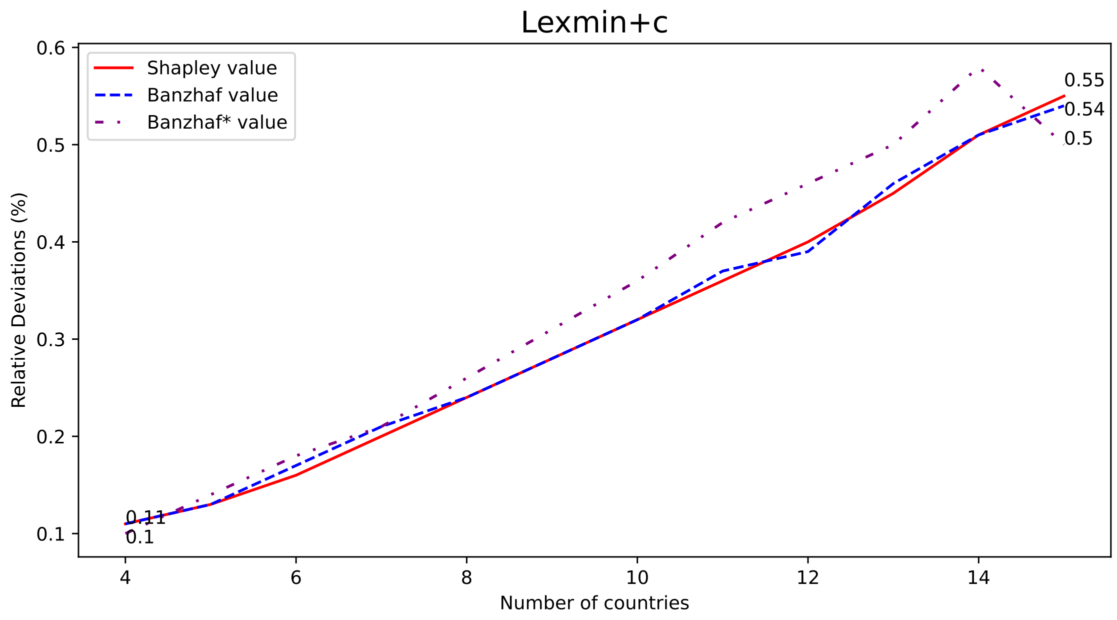

Figure 4.14: Comparing the *lexmin+c* graphs for the Shapley value and Banzhaf value from Figure 4.11 with the one for the Banzhaf* value (varying country sizes).

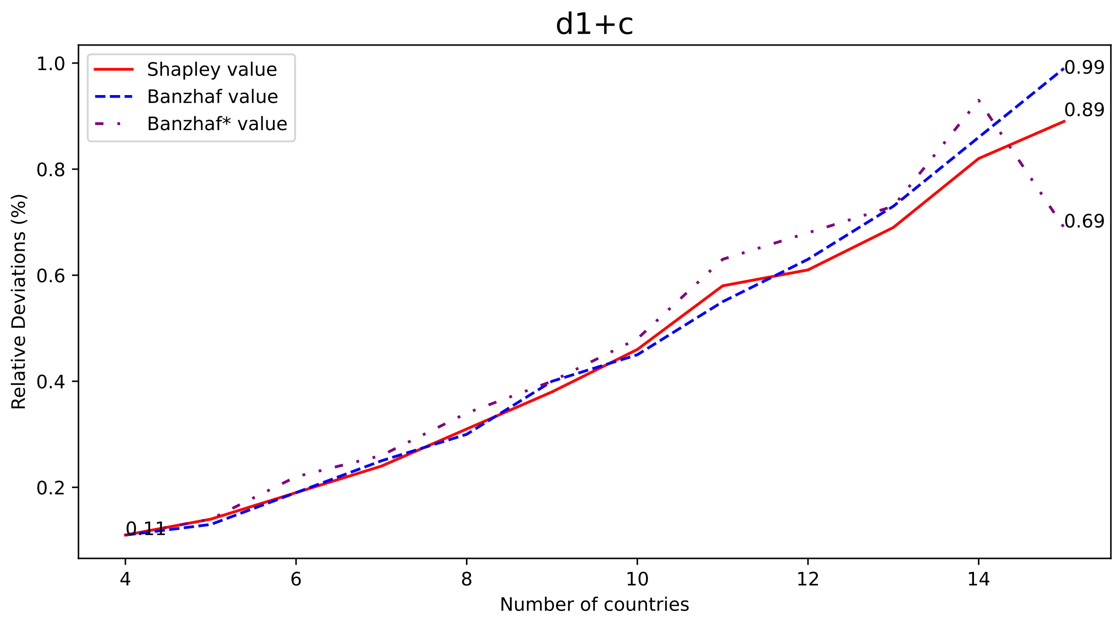

Figure 4.15: Comparing the *d1+c* graphs for the Shapley value and Banzhaf value from Figure 4.11 with the one for the Banzhaf* value (varying country sizes).

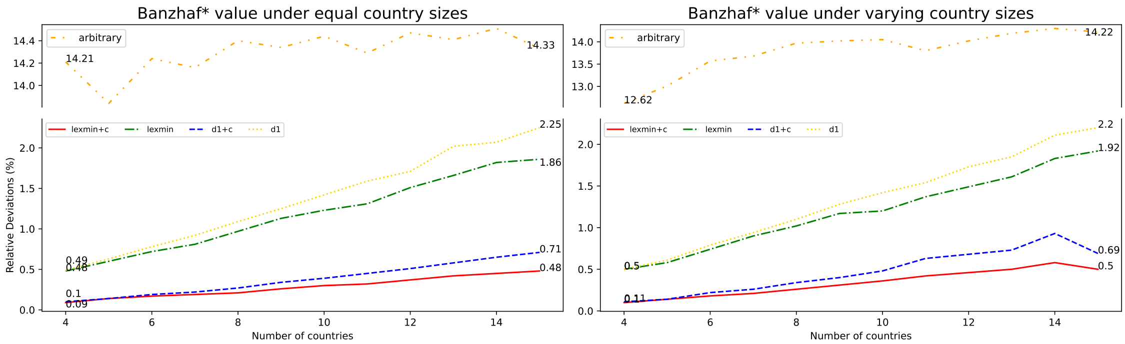

Figure 4.16: Average total relative deviations for the Banzhaf* value across five scenarios under equal and varying country sizes where the number of countries n ranges from 4 to 15.

core. Moreover, all accumulated initial allocations and accumulated solutions turned out to be in the core, except for a few rare cases of accumulated solutions under the arbitrary scenario. For comparison, we evaluate how far away both the accumulated initial allocations and accumulated solutions are from violating a core inequality in the accumulated partitioned matching game. We do this by taking the radius of the largest ball that can be fit into the core with its center being an accumulated initial allocation, or accumulated solution, respectively.[2] Unsurprisingly, this radius is decreasing as the number of countries is increasing. Moreover, the distance of violating a core allocation is practically the same, independently of the chosen scenario. We refer to Tables 4.3 and 4.4 for details and to Table 4.5 for a summary obtained from these two tables by averaging over the number of countries for the *lexmin+c* scenario.

From Table 4.5 we see a high and similar level of stability for all choice of initial allocations. Although the Shapley and Banzhaf values provide consistently the smallest deviations (see Figure 4.2), Table 4.5 shows that the tau value (highest), the benefit value and the nucleolus provide higher levels of coalitional stability not only for the accumulated initial allocations, but also for the accumulated solutions.

[2]The allocations with the largest radius form the *least core* of the game, which we do not attempt to find.

Allocations & scenarios / Countries	4	5	6	7	8	9	10	11	12	13	14	15
Without cooperation	1124.28	974.56	850.60	759.40	687.28	628.92	583.84	538.68	497.72	474.92	440.52	421.88
Shapley value:												
lexmin+c	108.6%	125.3%	143.1%	159.7%	175.8%	191.7%	209.4%	224.0%	240.6%	253.5%	270.4%	286.3%
lexmin	108.6%	125.3%	143.1%	159.7%	175.9%	191.6%	209.5%	224.1%	240.4%	253.3%	270.3%	286.0%
d1+c	108.7%	125.3%	143.0%	159.6%	176.0%	191.4%	209.3%	224.2%	240.6%	253.1%	270.0%	286.3%
d1	108.7%	125.3%	143.1%	159.7%	175.8%	191.6%	209.2%	224.3%	240.5%	253.2%	270.0%	286.0%
arbitrary matching	108.5%	125.2%	142.9%	159.5%	175.7%	191.5%	209.2%	224.0%	240.2%	253.2%	270.1%	285.7%
Nucleolus:												
lexmin+c	108.7%	125.4%	143.1%	159.5%	175.8%	191.6%	209.2%	224.3%	240.6%	253.3%	270.1%	286.2%
lexmin	108.6%	125.4%	143.1%	159.6%	175.9%	191.6%	209.3%	224.3%	240.6%	253.3%	270.2%	286.2%
d1+c	108.7%	125.4%	142.9%	159.7%	175.9%	191.5%	209.3%	224.2%	240.4%	253.6%	270.4%	286.2%
d1	108.6%	125.4%	143.1%	159.7%	175.9%	191.6%	209.4%	224.2%	240.5%	253.6%	270.3%	286.4%
arbitrary matching	108.5%	125.2%	142.9%	159.5%	175.7%	191.5%	209.2%	224.0%	240.2%	253.2%	270.1%	285.7%
Banzhaf value:												
lexmin+c	108.7%	125.3%	143.0%	159.8%	175.8%	191.7%	209.3%	224.5%	240.4%	253.3%	270.1%	286.2%
lexmin	108.6%	125.2%	143.0%	159.7%	175.9%	191.6%	209.1%	224.3%	240.5%	253.0%	270.0%	285.9%
d1+c	108.7%	125.3%	143.0%	159.6%	175.9%	191.9%	209.3%	224.5%	240.6%	253.2%	270.2%	286.3%
d1	108.5%	125.2%	143.0%	159.8%	175.9%	191.7%	209.4%	224.2%	240.5%	253.1%	270.3%	286.1%
arbitrary matching	108.5%	125.2%	142.9%	159.5%	175.7%	191.5%	209.2%	224.0%	240.2%	253.2%	270.1%	285.7%
tau:												
lexmin+c	108.7%	125.4%	142.9%	159.7%	176.0%	191.7%	209.4%	224.2%	240.5%	253.5%	270.5%	286.2%
lexmin	108.6%	125.3%	143.1%	159.7%	175.8%	191.6%	209.7%	224.4%	240.7%	253.5%	270.3%	286.4%
d1+c	108.6%	125.4%	143.1%	159.7%	175.9%	191.7%	209.5%	224.5%	240.7%	253.7%	270.4%	286.3%
d1	108.7%	125.4%	143.0%	159.8%	176.0%	191.8%	209.6%	224.4%	240.7%	253.4%	270.3%	286.2%
arbitrary matching	108.5%	125.2%	142.9%	159.5%	175.7%	191.5%	209.2%	224.0%	240.2%	253.2%	270.1%	285.7%
Benefit value:												
lexmin+c	108.6%	125.4%	143.0%	159.7%	176.0%	191.7%	209.6%	224.1%	240.6%	253.2%	270.4%	286.2%
lexmin	108.6%	125.5%	142.9%	159.7%	176.0%	191.7%	209.4%	224.4%	240.6%	253.7%	270.3%	286.3%
d1+c	108.7%	125.4%	143.1%	159.7%	175.9%	191.9%	209.4%	224.3%	240.6%	253.7%	270.1%	286.2%
d1	108.6%	125.3%	143.1%	159.7%	175.9%	191.6%	209.4%	224.4%	240.7%	253.7%	270.3%	286.1%
arbitrary matching	108.5%	125.2%	142.9%	159.5%	175.7%	191.5%	209.2%	224.0%	240.2%	253.2%	270.1%	285.7%
Contribution value:												
lexmin+c	108.6%	125.4%	143.1%	159.6%	176.0%	191.6%	209.4%	224.2%	240.6%	253.3%	270.3%	286.3%
lexmin	108.6%	125.4%	143.0%	159.7%	175.9%	191.8%	209.5%	224.2%	240.5%	253.4%	270.2%	286.2%
d1+c	108.7%	125.3%	143.1%	159.7%	175.9%	191.6%	209.6%	224.3%	240.5%	253.4%	270.2%	286.1%
d1	108.6%	125.3%	143.0%	159.6%	176.2%	191.8%	209.4%	224.2%	240.5%	253.6%	270.4%	286.0%
arbitrary matching	108.5%	125.2%	142.9%	159.5%	175.7%	191.5%	209.2%	224.0%	240.2%	253.2%	270.1%	285.7%

Table 4.1: For $n \in \{4, \ldots, 15\}$, the improvement on the average number of kidney transplants if cooperation is allowed where all countries have the same size. For example, if $n = 4$, y is the Shapley value and the scenario is *lexmin+c*, then the average number of kidney transplants changes from 1124.28 (no cooperation) to $1.086 \times 1124.28 = 1220.97$.

4.3.5 Convexity and Quasibalancedness

Recall that the tau value is only defined if the game is quasibalanced and that we replaced the tau value by the benefit value if the tau value is not defined. We also recall that tau value and benefit value coincide when the game is convex. Table 4.6 provides justification for this replacement.

CPU times / Countries	4	5	6	7	8	9	10	11	12	13	14	15
Data preparation	0.59	0.64	0.56	0.53	0.53	0.52	0.56	0.54	0.50	0.52	0.51	0.50
Graph building	20.74	20.81	20.82	20.69	20.73	20.67	20.77	20.70	20.57	20.67	20.59	20.66
Game generation	0.06	0.10	0.18	0.34	0.66	1.29	2.59	5.06	9.90	19.96	39.17	79.58
Shapley value	10^{-5}	10^{-5}	10^{-4}	10^{-4}	10^{-4}	10^{-3}	10^{-3}	10^{-3}	*0.01*	*0.02*	*0.04*	*0.08*
Nucleolus	10^{-3}	10^{-3}	10^{-3}	10^{-3}	*0.01*	*0.02*	*0.03*	*0.04*	*0.06*	*0.11*	*0.22*	*0.44*
Banzhaf	10^{-5}	10^{-5}	10^{-4}	10^{-4}	10^{-4}	10^{-3}	10^{-3}	10^{-3}	*0.01*	*0.02*	*0.04*	*0.08*
tau	10^{-5}	10^{-5}	10^{-4}	10^{-4}	10^{-4}	10^{-3}	10^{-3}	10^{-3}	*0.01*	*0.02*	*0.04*	*0.08*
Benefit	10^{-5}	*0.00*	10^{-5}	10^{-6}	10^{-5}	*0.00*	10^{-6}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Contribution	10^{-5}	*0.00*	10^{-5}	*0.00*	10^{-6}	*0.00*	*0.00*	10^{-5}	10^{-5}	10^{-5}	*0.00*	10^{-5}
Total: lexmin+c	21.75	21.95	21.98	22.00	22.38	22.96	24.46	26.82	31.51	41.73	60.93	101.46
Total: d1+c	21.74	21.92	21.92	21.90	22.26	22.80	24.28	26.64	31.33	41.56	60.71	101.32
Total: lexmin	21.78	21.96	21.98	21.99	22.36	22.93	24.42	26.78	31.46	41.68	60.81	101.4
Total: d1	21.77	21.94	21.93	21.91	22.27	22.81	24.29	26.65	31.33	41.53	60.78	101.11
Total: arbitrary	21.28	21.55	21.56	21.57	21.94	22.51	24.00	26.41	31.13	41.35	60.51	101.18

Table 4.2: Computational times for a single instance, broken down to the different computation tasks for *lexmin+c* where all countries have the same size, while the total rows for the different scenarios are averaged over the four initial allocations.

4.4 Concluding Remarks

Our simulations showed that using maximum matchings that are lexicographically minimal with respect to the country deviations from target allocations leads to a significant improvement for IKEPs. Moreover, they showed that this improvement is even more significant when the number of countries is large. This is relevant, as IKEPs, such as Eurotransplant, are under development and others, such as Scandiatransplant, are expected to grow.

Both lexicographically minimal maximum matchings and maximum matchings that only minimize the maximum deviation d_1 can be computed in polynomial time. In practice one might expect that the latter can still be computed faster. However, our simulations showed that computing them instead of maximum matchings that only minimize the maximum deviation indeed does not require any significant additional computational time (see Section 4.3.3).

A challenging aspect was to compute the nucleolus of partitioned matching games consisting of up to fifteen countries. For this we used the state-of-the-art *Lexicographical Descent* method of [88].

Allocations & policies	4	5	6	7	8	9	10	11	12	13	14	15	Total
Shapley:													
lexmin+c	105.25	85.64	70.31	59.65	51.39	44.60	40.43	35.65	32.31	28.78	26.86	24.71	50.46
lexmin	106.08	85.29	69.95	59.25	51.54	44.70	40.99	35.60	32.42	28.75	27.13	24.21	50.49
d1+c	106.12	85.84	69.93	59.29	52.06	45.21	40.04	35.44	32.19	28.72	26.69	24.13	50.47
d1	106.15	85.09	70.03	59.46	51.30	44.58	40.57	36.01	32.01	29.07	26.76	24.35	50.45
Nucleolus:													
lexmin+c	108.84	88.40	73.63	62.84	54.48	47.92	43.85	38.55	34.49	31.22	29.01	26.81	53.34
lexmin	108.40	88.39	73.81	63.19	54.84	48.03	43.43	38.52	34.37	31.34	29.31	26.64	53.36
d1+c	108.68	88.79	73.76	63.03	54.00	48.44	43.55	38.30	34.26	31.42	29.06	26.80	53.34
d1	108.34	88.33	73.96	63.52	54.57	47.86	43.77	38.37	34.12	31.43	29.00	26.80	53.34
Banzhaf:													
lexmin+c	105.08	84.88	68.95	59.31	50.22	43.86	39.70	35.09	30.64	27.49	26.05	23.53	50.08
lexmin	105.38	84.75	69.68	58.83	50.50	43.49	38.93	35.47	31.04	27.88	25.74	23.03	50.11
d1+c	104.72	83.75	69.43	58.87	50.78	43.74	39.07	34.94	31.53	27.54	26.02	23.14	49.98
d1	105.43	84.72	69.67	59.21	50.93	43.79	39.62	35.08	30.87	27.58	26.03	23.15	50.21
tau:													
lexmin+c	107.78	87.52	72.96	61.84	54.17	48.15	43.38	38.77	34.81	32.06	29.54	26.97	53.62
lexmin	107.21	87.48	73.17	62.18	54.29	48.01	43.53	38.62	34.73	31.89	29.61	27.21	53.63
d1+c	107.34	87.61	72.84	62.15	54.21	48.06	43.56	38.46	34.90	31.67	29.50	27.16	53.58
d1	107.08	87.42	73.12	62.30	54.30	48.20	43.57	38.83	34.76	31.72	29.67	26.85	53.60
Benefit:													
lexmin+c	107.45	87.47	73.18	62.11	54.62	48.21	43.66	38.87	35.25	32.07	30.03	27.92	53.40
lexmin	107.43	87.32	72.52	61.57	54.11	47.93	43.21	38.65	35.06	32.29	30.01	27.71	53.15
d1+c	107.37	87.55	72.60	61.90	54.67	47.87	43.36	38.83	35.19	32.46	29.86	27.79	53.29
d1	106.95	87.52	72.56	61.40	54.58	48.28	43.13	38.92	34.95	32.30	29.81	27.71	53.18
Contribution:													
lexmin+c	101.26	81.72	67.48	55.60	48.95	42.44	38.69	33.61	31.07	27.72	26.09	23.13	48.15
lexmin	101.69	81.48	66.79	56.16	49.17	42.85	38.47	33.69	30.90	28.04	25.88	23.43	48.21
d1+c	101.43	81.10	66.52	55.89	49.00	42.30	38.49	33.66	30.83	27.62	26.06	23.16	48.01
d1	101.98	81.12	66.53	56.31	49.40	42.66	38.61	33.76	30.94	27.72	26.22	23.21	48.21

Table 4.3: Average distance of accumulated initial allocations from violating a core inequality of the accumulated partitioned matching games for the six initial allocations, the four scenarios and the twelve country set sizes as well as total average over all country set sizes where all countries have the same size.

Allocations & policies	4	5	6	7	8	9	10	11	12	13	14	15	Total
Shapley:													
lexmin+c	105.24	85.58	70.22	59.57	51.37	44.55	40.39	35.55	32.19	28.74	26.68	24.60	50.39
lexmin	105.79	84.85	69.60	58.51	50.77	43.77	39.89	34.58	31.43	27.68	25.99	23.10	49.66
d1+c	106.17	85.84	69.98	59.32	51.98	45.06	39.90	35.33	32.08	28.58	26.48	24.07	50.40
d1	105.83	84.61	69.69	58.88	50.36	43.22	39.53	34.76	30.70	27.50	25.27	22.93	49.44
arbitrary	47.92	35.98	29.64	22.49	19.99	14.50	13.81	11.19	8.11	7.23	7.52	6.79	18.76
Nucleolus:													
lexmin+c	108.75	88.27	73.37	62.59	54.28	47.60	43.65	38.22	34.28	31.06	28.79	26.36	53.10
lexmin	107.19	86.75	72.14	61.53	52.39	46.01	41.46	36.29	32.43	28.96	27.07	24.37	51.38
d1+c	108.64	88.66	73.72	62.80	53.76	48.08	43.01	37.65	33.48	30.56	28.13	25.69	52.85
d1	106.94	86.64	71.98	60.43	52.31	44.02	40.48	34.94	30.74	27.60	25.14	23.40	50.38
arbitrary	47.92	35.98	29.64	22.49	19.99	14.50	13.81	11.19	8.11	7.23	7.52	6.79	18.76
Banzhaf:													
lexmin+c	105.02	84.82	68.91	59.24	50.16	43.8	39.66	35.04	30.57	27.45	25.99	23.49	50.03
lexmin	105.01	84.43	69.20	58.21	49.89	42.78	38.17	34.96	30.43	27.03	24.84	22.4	49.51
d1+c	104.82	83.82	69.33	58.85	50.76	43.61	38.99	34.94	31.43	27.35	25.89	23.15	49.94
d1	105.02	84.30	69.05	58.52	50.19	43.12	38.64	34.02	29.8	26.24	24.74	22.17	49.34
arbitrary	47.92	35.98	29.65	22.51	20.09	14.81	14.47	11.46	8.80	8.00	7.96	7.29	19.47
tau:													
lexmin+c	107.7	87.55	72.8	61.59	54.02	47.76	43.3	38.53	34.47	31.84	29.21	26.73	53.43
lexmin	106.62	86.35	71.97	60.87	52.81	46.09	41.65	36.61	32.96	29.66	27.6	25	52.01
d1+c	107.35	87.51	72.77	62.03	53.87	47.82	42.91	37.85	34.14	30.64	28.76	26.24	53.10
d1	106.52	86.2	71.66	60.39	52.06	45.62	40.47	35.39	31.73	28.31	26.29	22.99	51.09
arbitrary	47.92	35.98	29.65	22.51	20.09	14.81	14.47	11.46	8.80	8.00	7.96	7.29	19.47
Benefit:													
lexmin+c	107.48	87.39	73.02	61.88	54.39	47.99	43.45	38.59	34.94	31.82	29.69	27.58	53.19
lexmin	107.03	86.29	71.51	60.26	52.36	46.17	41.22	36.78	32.63	30.09	27.68	25.14	51.43
d1+c	107.32	87.50	72.55	61.76	54.53	47.52	42.91	38.54	34.56	31.85	29.26	26.85	52.93
d1	106.37	86.66	71.23	59.91	52.67	46.03	40.33	36.12	31.98	28.82	26.68	24.39	50.93
arbitrary	47.92	35.98	29.64	22.49	19.99	14.50	13.81	11.19	8.11	7.23	7.52	6.79	18.76
Contribution:													
lexmin+c	101.15	81.79	67.55	55.62	48.85	42.31	38.71	33.73	31.11	27.58	25.98	23.18	48.13
lexmin	102.06	82.25	66.94	56.30	49.76	42.49	38.57	33.79	30.45	27.63	25.57	23.24	48.25
d1+c	101.57	81.20	66.79	55.84	49.25	42.33	38.40	33.48	30.41	27.44	25.95	23.15	47.98
d1	102.38	81.24	66.72	56.17	49.50	42.19	37.57	33.59	29.87	26.90	24.57	22.33	47.75
arbitrary	47.92	35.98	29.64	22.49	19.99	14.50	13.81	11.19	8.11	7.23	7.52	6.79	18.76

Table 4.4: Average distance of accumulated solutions from violating a core inequality of the accumulated partitioned matching games for the six initial allocations, the four scenarios and the sanity check of arbitrary matching, for the 12 country set sizes as well as total average over all country set sizes where all countries have the same size.

	Shapley	nucleolus	Banzhaf	tau	benefit	contribution	Banzhaf*
Accumulated initial allocation	50.46	53.34	50.08	53.62	53.40	48.15	49.74
Accumulated solution	50.39	53.10	50.03	53.43	53.19	48.13	48.39

Table 4.5: Average distances, over n ranging from 4 to 15, of accumulated initial allocations (first row) and accumulated solutions from violating a core inequality of the accumulated partitioned matching games under the *lexmin+c* scenario where all countries have the same size. For example, by using the Shapley value as the initial allocation, every coalition of countries on average has at least 50.39 more kidney transplants by participating in the IKEP than they would be able to achieve on their own.

n	not quasi-balanced	convex	not convex, tau = benefit
4	0%	36.7%	55.3%
5	0.0167%	9%	60%
6	0%	2.2%	51.5%
7	0%	0.9%	45%
8	0%	0.4%	38.9%
9	0.025%	0.3%	32.2%
10	0.008%	0.1%	27.9%
11	0.05%	0%	22.9%
12	0.083%	0.1%	18.3%
13	0.15%	0.1%	14.8%
14	0.05%	0%	12.9%
15	0.1%	0%	11.2%
Total	0.04%	4.14%	31.6%

Table 4.6: The first column refers to number of countries. The second, third and fourth columns give, respectively, the percentage of non-quasibalanced games; percentage of convex games; and percentage of non-convex games with tau and benefit values for the initial allocations coinciding. The percentages of games are taken over all rounds, all initial allocations and all scenarios where all countries have the same size. The percentages of games in the last row are taken over all rounds, all initial allocations, all scenarios and all numbers of countries where all countries have the same size.

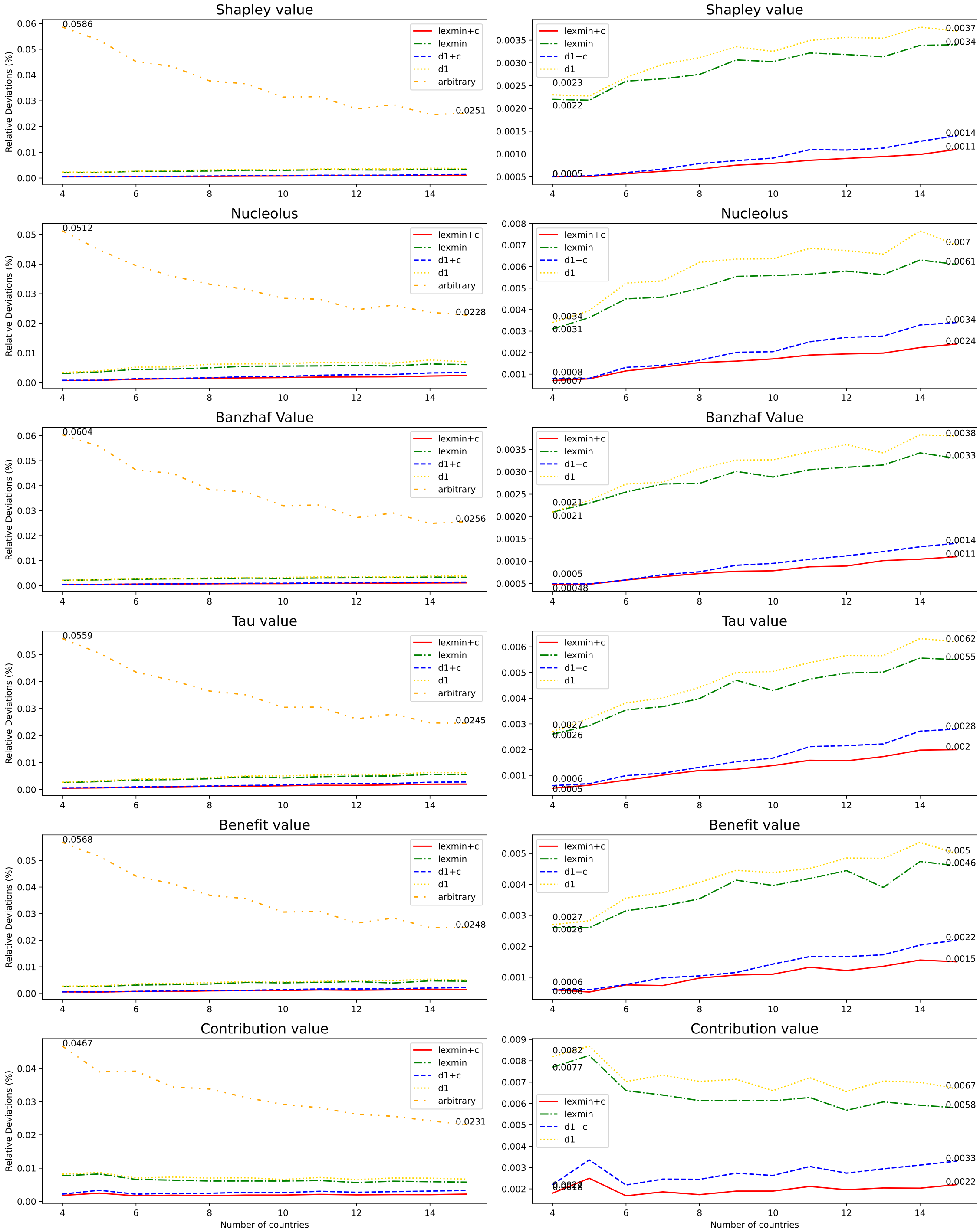

Figure 4.17: Average maximum relative deviations for the situation where the countries vary in size. The number of countries n is ranging from 4 to 15. The figures on the right side zoom in on the figures from the left side by removing the results for the arbitrary matching scenario.

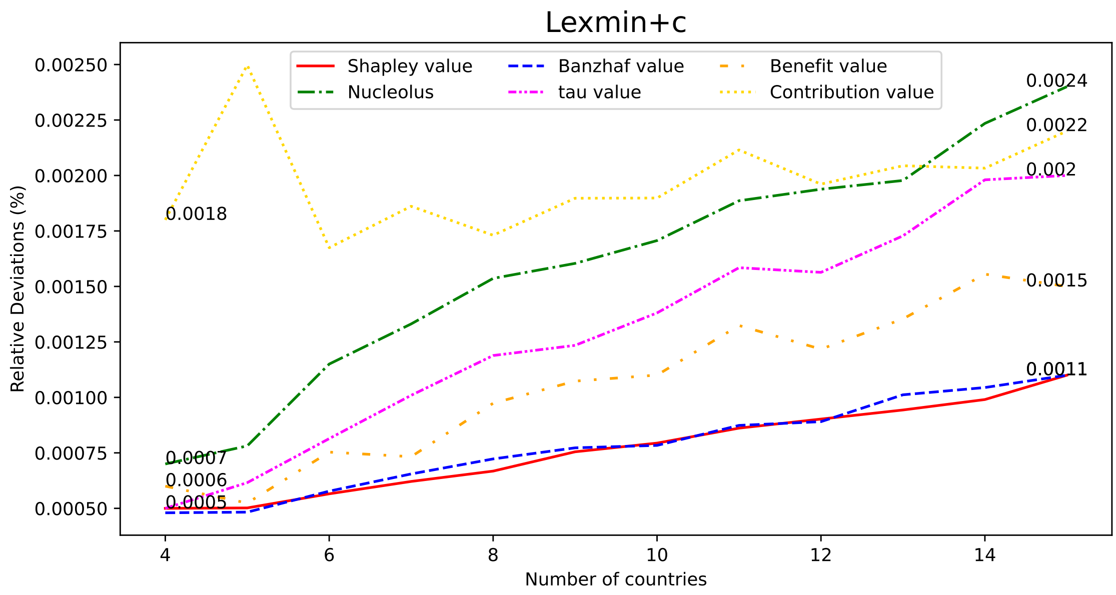

Figure 4.18: Displaying the six *lexmin+c* graphs of Figure 4.17 in one plot (varying country sizes).

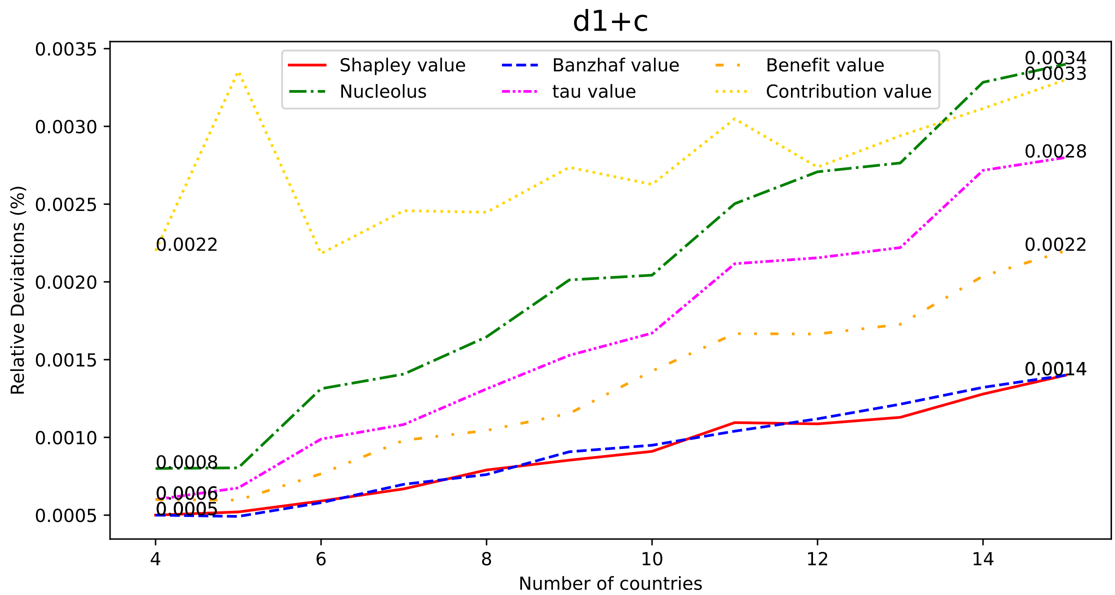

Figure 4.19: Displaying the six *d1+c* graphs of Figure 4.17 in one plot (varying country sizes).

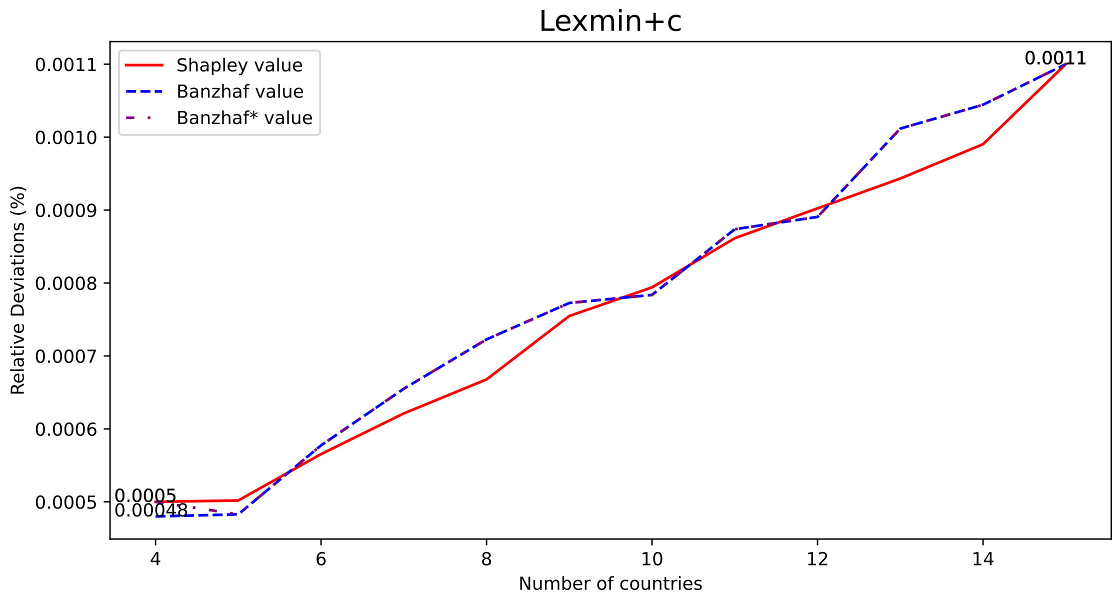

Figure 4.20: Comparing the *lexmin+c* graphs for the Shapley value and Banzhaf value from Figure 4.17 with the one for the Banzhaf* value (varying country sizes).

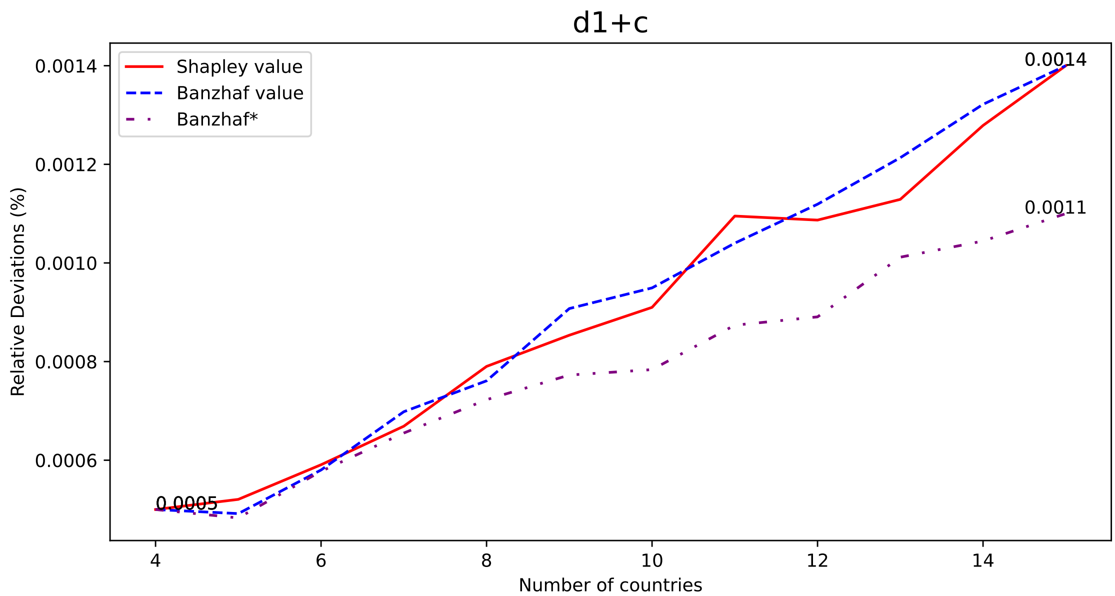

Figure 4.21: Comparing the *d1+c* graphs for the Shapley value and Banzhaf value from Figure 4.17 with the one for the Banzhaf* value (varying country sizes).

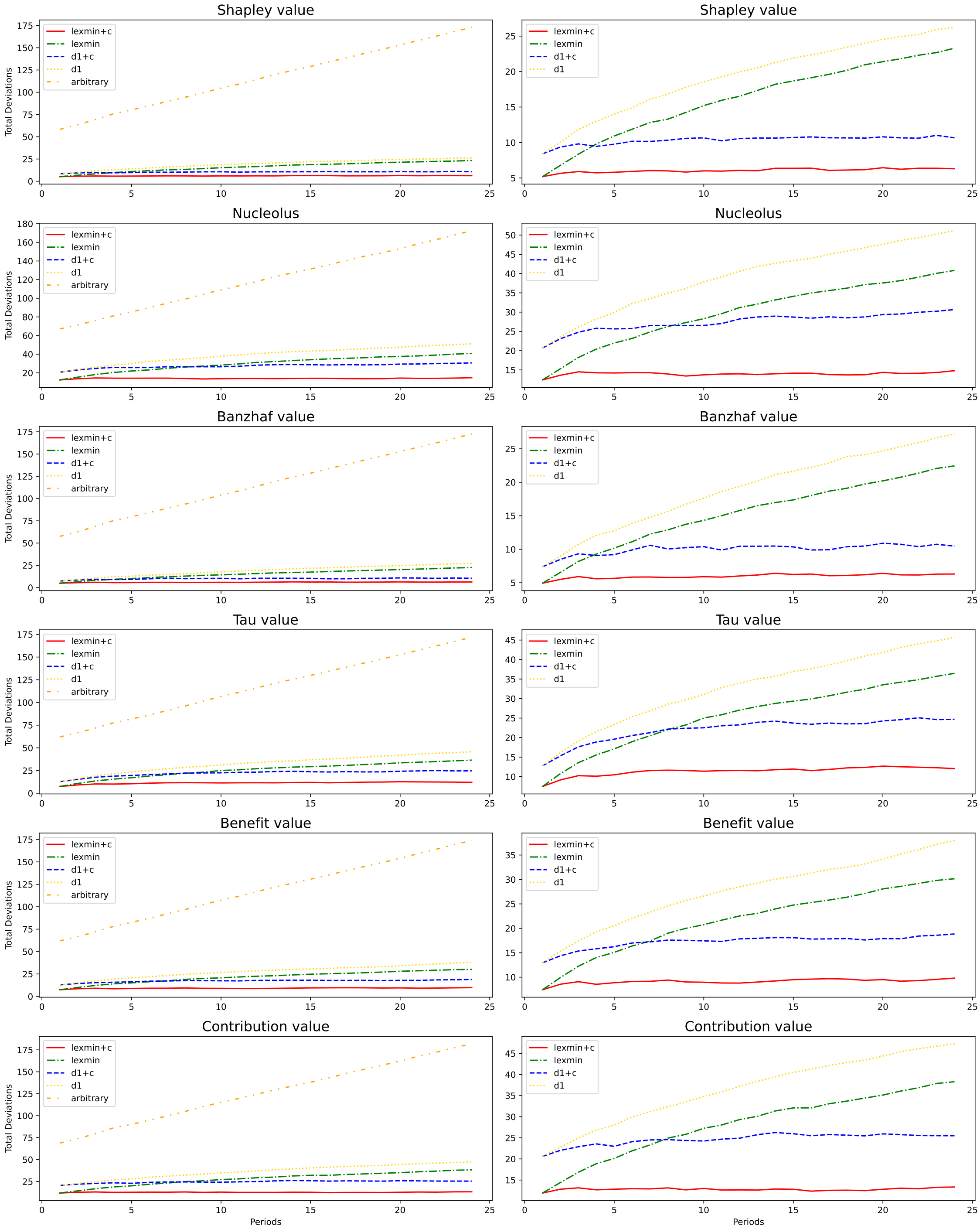

Figure 4.22: The average accumulated deviation over the 24 rounds when the number of countries $n = 15$ where all countries have the same size. The right side of the figure is taken from the left side after omitting the additional scenario where arbitrary maximum matchings are chosen.

CHAPTER 5

Simulations and Results for $\ell = \infty$

In this chapter, we start with our theoretical results (Section 5.1). Permutation games, i.e. partitioned permutation games of width 1, have a nonempty core [89], and a core allocation can be found in polynomial time [90]. We generalize these two results to partitioned permutation games of any width c.

Theorem 14. *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 (see Chapter 1), we cannot hope to generalize Theorem 2 (see Chapter 1) 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 $\mathsf{P} \neq \mathsf{NP}$).

Theorem 15. *The problem of finding an optimal solution that is weakly close to a given target allocation is NP-hard even for partitioned permutation games of width $c = 2$.*

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 [91] in 1982.

Theorem 16. *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 16 is not practical for instances of realistic size (which we aim for). Moreover, it is a randomized algorithm. Hence, it it not acceptable either in the setting of kidney exchange, as policy makers would only use a solution yielding the maximum number of kidney transplants. Therefore and also due to Theorem 15, we formulate the problems of computing a weakly or strongly close optimal solution as integer linear programs, as described in Section 5.2. 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 5.3, simulations for IKEPs up to *ten* countries, so more than the four countries in the simulations for $\ell = 3$ [10, 11] and the eight countries for $\ell = 3$ in Chapter 6, but less than the fifteen countries in the simulations for $\ell = 2$ (see Chapter 4).

Our simulations show, as for $\ell = 2$ in Chapter 4, that a credit system using strongly close optimal solutions instead of weakly optimal solutions makes an IKEP up to 56% more balanced, without decreasing the overall number of transplants. The exact improvement is determined by the choice of solution concept. Our simulations also indicate that out of the seven solution concepts, using the Banzhaf* value leads to the lowest deviations, namely on average, a deviation of at most 0.90% from the initial allocation (but the differences between the Banzhaf* value with the Banzhaf value or 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 [38]). However, the exchange cycles may be very large, in particular in the starting round.

5.1 Theoretical Results

We start with Theorem 14. Recall that the width c of a partitioned permutation game (N, v) defined on a directed graph $G = (V, A)$ with vertex partition $(V_1, \ldots, V_n)$ is the maximum size of a set V_i.

Theorem 14 (restated) *The core of every partitioned permutation game is non-empty, 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 [90] (and such a core allocation always exists [89]), the same holds for the former.

Let (N, v) be a partitioned permutation game on a graph $G = (V, A)$ with partition $(V_1, \ldots, V_n)$ 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 set $i \in N$, 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 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 $\mathcal{C} = \{C_1, C_2, .., C_k\}$ 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 [92, 93].

Exact 3-Cover	
Instance:	A family of 3-element subsets of $[3n]$, $\mathcal{S} = \{S_1, \dots, S_{3n}\}$, where each element belongs to exactly three sets
Question:	Is there an *exact* 3-*cover* for $\mathcal{S}$, that is, a subset $\mathcal{S}' \subset \mathcal{S}$ such that each element appears in exactly one of the sets of $\mathcal{S}'$?

Given an instance I of Exact 3-Cover, we construct a partitioned permutation game (N, v) as follows (see Figure 5.1 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 $x_{6n+1} := x_1$.
- 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.

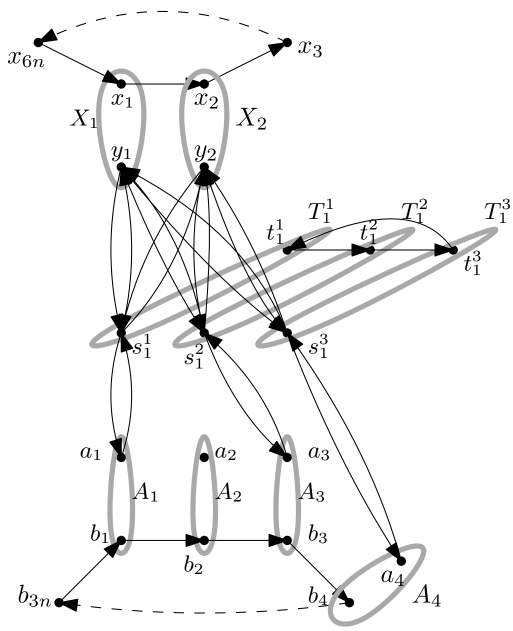

Figure 5.1: An illustration for Theorem 14. The figure shows the part of the construction for a set $S_1 = \{1, 3, 4\}$. Dashed arcs denote that there is a cycle through the given vertex whose vertices are not included in the picture. Dark grey bold ellipses denote the players.

If we sum up all allocation values we find that

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

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.

"⇒" First suppose $\{S_{k_1}, \ldots, S_{k_n}\}$ is an exact 3-cover in I. We claim that $\mathcal{P} = \{A_i \mid i \in [3n]\} \cup \{T^l_{k_i} \mid i \in [n], l \in [3]\}$ is a blocking coalition. We first show that $v(\mathcal{P}) = 12n$, which can be seen as follows. First, the s^l_j 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^l_j vertices can be covered too, as for each $j \in \{k_1, \ldots, k_n\}$ all of T_j^1, T_j^2, T_j^3 belong to $\mathcal{P}$. Then, x is not in the core, as

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

"$\Leftarrow$" 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

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

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

First suppose that $\mathcal{A} \cup \mathcal{X} \subseteq \mathcal{P}$. 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 $\mathcal{X} \subseteq \mathcal{P}$. 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,

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

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 $\mathcal{A} \subseteq \mathcal{P}$, 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,

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

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. □

Before we prove Theorem 15, 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 $\mathcal{C}$ of G, we let $s_p(\mathcal{C})$ denote the number of arcs (u, v) with $v \in V_p$ that belong to some directed cycle of $\mathcal{C}$. We say that $\mathcal{C}$ *satisfies* a set of intervals $\{I_1, \dots, I_n\}$ if $s_p(\mathcal{C}) \in I_p$ for every $p \in \{1, \dots, n\}$.

Lemma 2. *For instances $(G, \mathcal{V}, \mathcal{I})$, 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 $\mathcal{I} = \{I_1, \dots, I_n\}$ is a set of intervals, the problem of finding a maximum cycle packing of G satisfying $\mathcal{I}$ 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 $\mathcal{I}$ 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 $v \in V \setminus U$ and its copy $v' \in V' \setminus U'$, we add the edge uu' 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 uu' 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 14, 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 Figure 5.2):

– 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 $x_1, \ldots, x_{2n}$ and $2n$ sink vertices $y_1, \ldots, y_{2n}$.

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.

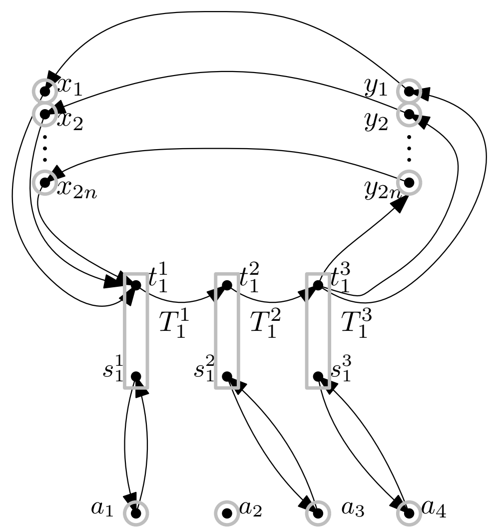

Figure 5.2: Illustration for Lemma 2, showing the construction for a set $S_1 = \{1, 2, 4\}$. Grey bold lines mark the players.

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.

"⇒" First suppose I has an exact 3-cover $S_{l_1}, \ldots, S_{l_n}$. We create a cycle packing $\mathcal{C}$ of G. For each $j \in \{l_1, \ldots, l_n\}$, 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 $j \notin \{l_1, \ldots, l_n\}$ 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 $[3n] \setminus \{l_1, \ldots, l_n\}$.

Clearly, $\mathcal{C}$ is a cycle packing. Each A_i has an incoming arc, as $S_{l_1}, \ldots, S_{l_n}$ was 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 $\{(y_i, x_i), (x_i, t_{j_i}^1), (t_{j_i}^1, t_{j_i}^2), (t_{j_i}^3, y_i)\}$, and so did each X_i and each Y_i. Hence, all lower bounds are satisfied.

"⇐" 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 $\mathcal{C}$.

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^i_j must have one incoming arc, it follows that for all these j, the cycles $\{(a_{j_1}, s^1_j), (s^1_j, a_{j_1})\}$, $\{(a_{j_2}, s^2_j), (s^2_j, a_{j_2})\}$, $\{(a_{j_3}, s^3_j), (s^3_j, a_{j_3})\}$ are included in $\mathcal{C}$, so they are vertex disjoint. Hence, the corresponding sets must form an exact 3-cover. □

Theorem 15 (restated). *The problem of finding an optimal solution that is weakly close to a given target allocation is* NP*-hard even for partitioned permutation games of width $c = 2$.*

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 $\mathcal{C}$. Letting each $I_p = [x_p, x_p]$ and applying Lemma 2, we see that finding a cycle packing where each $s_p(\mathcal{C})$ 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 $d_1(\mathcal{C}) = \max_{p \in N}\{|x_p - s_p(\mathcal{C})|\}$; 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. □

In the remainder of this chapter, the following problem plays an important role:

> q-EXACT PERFECT MATCHING
>
> *Instance:* An undirected bipartite graph $B = (U, W; E)$, where each edge is coloured with one of $\{1, \ldots, q\}$, and numbers $k_1, \ldots, k_q$.
>
> *Question:* Is there a perfect matching in B consisting of k_q edges of each colour q?

For $q = 2$, this problem is also known as EXACT PERFECT MATCHING, which, as mentioned, was introduced by Papadimitriou and Yannakakis [91] 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 17), from which Theorem 16 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 $\mathcal{C}$, $d'(\mathcal{C}) = (|x_{p_1} - s_{p_1}(\mathcal{C})|, \ldots, |x_{p_n} - s_{p_n}(\mathcal{C})|)$ is the *unordered deviation vector* of $\mathcal{C}$.

Theorem 17. *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, where $n = |N|$.*

Proof. Let (N, v) be a partitioned permutation game with n players, defined on a directed graph $G = (V, A)$ with vertex partition $V_1, \ldots, V_n$. As mentioned, we reduce to q-Exact Perfect Matching for an appropriate value of q. From (N, v) and a vector $d' = (d'_1, \ldots, d'_n)$ 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 $p \in \{1, \ldots, n\}$, let $k_p = d'_p$.

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

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 [94]. □

We cannot hope to derandomize the algorithm from Theorem 17 without first solving 2-Exact Perfect Matching problem in polynomial time, as we now show.

Theorem 18. *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 v_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 $V_2 = V(G) \setminus V_1$. 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 $\mathcal{C}$ with $s_1(\mathcal{C}) = k_1$ and $s_2(\mathcal{C}) = 3n - k_1$. 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 $s_1(\mathcal{C}) = k_1$ and $s_2(\mathcal{C}) = 3n - k_1$. 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 $\mathcal{C}$. Therefore, these edges induce a perfect matching in B with exactly k_1 red and k_2 blue edges. □

5.2 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 $V_1, \ldots, V_n$ for some $n \geq 1$. Recall that for a maximum cycle packing $\mathcal{C}$ of G, we let $s_p(\mathcal{C})$ denote the number of arcs (u, v) with $v \in V_p$ that belong to some directed cycle of $\mathcal{C}$. Recall also that $|x_p - s_p(\mathcal{C})|$ is the deviation of country $p \in N$ from its target x_p if $\mathcal{C}$ is chosen as optimal solution. Moreover, in the vector $d(\mathcal{C}) = (|x_{p_1} - s_{p_1}(\mathcal{C})|, \ldots, |x_{p_n} - s_{p_n}(\mathcal{C})|)$, the deviations $|x_p - s_p(\mathcal{C})|$ are ordered non-increasingly. Finally, we recall that $\mathcal{C}$ is strongly close to x if $d(\mathcal{C})$ 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. [1]). 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}.

$$M^* := \max_{e} \sum_{ij \in A} e_{ij} \quad \text{s.t.} \tag{edge-formulation}$$

$$\sum_{j:ji \in A} e_{ji} = \sum_{j:ij \in A} e_{ij} \quad \forall i \in V \tag{5.1}$$

$$\sum_{j:ji \in A} e_{ji} \leq \quad 1 \qquad \forall i \in V \tag{5.2}$$

Constraint 5.1 represents the (well-known) Kirchoff law. Constraint 5.2 ensures that every node is covered by at most one cycle. The objective function provides a maximum cycle packing (of size M^*).

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 M^* we solve the following ILP to find d_1^*:

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

$$\sum_{j:ji\in A} e_{ji} = \sum_{j:ij\in A} e_{ij} \quad \forall i \in V \tag{5.1}$$

$$\sum_{j:ji\in A} e_{ji} \leq 1 \quad \forall i \in V \tag{5.2}$$

$$\sum_{ij\in A} e_{ij} = M^* \tag{5.3}$$

$$\sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} - x_p \leq d_1 \quad \forall p \in N \tag{5.4}$$

$$x_p - \sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} \leq d_1 \quad \forall p \in N \tag{5.5}$$

Constraints 5.1-5.3 guarantee that all solutions are, in fact, maximum cycle packings. These constraints will be part of the formulation throughout the entire ILP-series. Constraints 5.4 and 5.5, 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

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

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

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. Let ε be a small positive constant that is weakly smaller than the minimum possible positive difference between any two countries. Note that there is a special case where minimum possible positive difference does not exist. This occurs when $\text{frac}(p) = 0.5$ for every $p \in N$, leading to four identical values, namely $\text{frac}(x_p) = 1 - \text{frac}(x_p) = \text{frac}(x_q) = 1 - \text{frac}(x_q) = 0.5$, and hence four pairwise differences being 0. For this special case, we let ε be a small constant satisfying $\varepsilon \in (0, 1]$.

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 $p \in N_t$.

$$\min_{z^1,e} \sum_{p \in N} z_p^1 \quad \text{s.t.} \tag{ILP$_{N_1}$}$$

$$\sum_{j:ji\in A} e_{ji} = \sum_{j:ij\in A} e_{ij} \qquad \forall i \in V \tag{5.1}$$

$$\sum_{j:ji\in A} e_{ji} \leq \quad 1 \qquad \forall i \in V \tag{5.2}$$

$$\sum_{ij\in A} e_{ij} = \quad M^* \tag{5.3}$$

$$\sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} - x_p \leq \quad d_1^* - \varepsilon(1 - z_p^1) \quad \forall p \in N \tag{5.6}$$

$$x_p - \sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} \leq \quad d_1^* - \varepsilon(1 - z_p^1) \quad \forall p \in N \tag{5.7}$$

As discussed, for each country p, the left hand side of either constraint 5.6 or constraint 5.7 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 constraint 5.6 or constraint 5.7 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:

$$\min_{d_2,e,z^1} d_2 \quad \text{s.t.} \tag{ILP$_{d_2}$}$$

$$
\begin{aligned}
\sum_{j:ji\in A} e_{ji} &= \sum_{j:ij\in A} e_{ij} && \forall i \in V && (5.1)\\
\sum_{j:ji\in A} e_{ji} &\leq \quad 1 && \forall i \in V && (5.2)\\
\sum_{ij\in A} e_{ij} &= \quad M^* && && (5.3)\\
\sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} - x_p &\leq \quad d_1^* && \forall p \in N && (5.8)\\
x_p - \sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} &\leq \quad d_1^* && \forall p \in N && (5.9)\\
\sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} - x_p &\leq \quad d_2 + z_p^1 d_1^* && \forall p \in N && (5.10)\\
x_p - \sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} &\leq \quad d_2 + z_p^1 d_1^* && \forall p \in N && (5.11)\\
\sum_{p\in N} z_p^1 &= n_1^* && && (5.12)
\end{aligned}
$$

(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 $L \geq 2(d_1^* - d_2^*)$:

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

$$\sum_{j:ji\in A} e_{ji} = \sum_{j:ij\in A} e_{ij} \qquad \forall i \in V \tag{5.1}$$

$$\sum_{j:ji\in A} e_{ji} \leq 1 \qquad \forall i \in V \tag{5.2}$$

$$\sum_{ij\in A} e_{ij} = M^* \tag{5.3}$$

$$\sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} - x_p \leq d_2^* - \varepsilon(1 - z_p^2) + z_p^1 L \quad \forall p \in N \tag{5.13}$$

$$x_p - \sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} \leq d_2^* - \varepsilon(1 - z_p^2) + z_p^1 L \quad \forall p \in N \tag{5.14}$$

$$\sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} - x_p \leq d_1^* \qquad \forall p \in N \tag{5.8}$$

$$x_p - \sum_{j\in V_p}\sum_{i:ij\in A} e_{ij} \leq d_1^* \qquad \forall p \in N \tag{5.9}$$

$$\sum_{p\in N} z_p^1 = n_1^* \tag{5.12}$$

$$z_p^1 + z_p^2 \leq 1 \quad \forall p \in N \tag{5.15}$$

Subsequently we follow a similar approach for all $t \geq 3$, until either $|N| = n_1^* + n_2^* + \cdots + n_t^*$ 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. $L = d_{t-1}^*$.

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

$$\sum_{j:ji\in A} e_{ji} = \sum_{j:ij\in A} e_{ij} \qquad \forall i \in V \tag{5.1}$$

$$\sum_{j:ji\in A} e_{ji} \leq \quad 1 \qquad \forall i \in V \tag{5.2}$$

$$\sum_{ij\in A} e_{ij} = \quad M^* \tag{5.3}$$

$$\sum_{i=1}^{t-1} z_p^i \leq \quad 1 \qquad \forall p \in N \tag{5.16}$$

$$\sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} - x_p \leq \quad d_t + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \tag{5.17}$$

$$x_p - \sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} \leq \quad d_t + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \tag{5.18}$$

$$\sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} - x_p \leq \quad \sum_{i=1}^{t-1} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t-1} z_p^i\right) L \qquad \forall p \in N \tag{5.19}$$

$$x_p - \sum_{j\in V_p} \sum_{i:ij\in A} e_{ij} \leq \quad \sum_{i=1}^{t-1} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t-1} z_p^i\right) L \qquad \forall p \in N \tag{5.20}$$

$$\sum_{p\in N} z_p^i = \quad n_i^* \qquad \forall i \in \{1, \ldots, t-1\} \tag{5.21}$$

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

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

$$\sum_{j:ji\in A} e_{ji} = \sum_{j:ij\in A} e_{ij} \qquad \forall i \in V \tag{5.1}$$

$$\sum_{j:ji\in A} e_{ji} \leq \quad 1 \qquad \forall i \in V \tag{5.2}$$

$$\sum_{ij\in A} e_{ij} = \quad M^* \tag{5.3}$$

$$\sum_{i=1}^{t-1} z_p^i \leq \quad 1 \qquad \forall p \in N \tag{5.16}$$

$$\sum_{j\in V_p} e_{ij} - x_p \leq \quad d_t^* - \varepsilon(1 - z_p^t) + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \tag{5.22}$$

$$x_p - \sum_{j\in V_p} e_{ij} \leq \quad d_t^* - \varepsilon(1 - z_p^t) + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \tag{5.23}$$

$$\sum_{j\in V_p} e_{ij} - x_p \leq \quad \sum_{i=1}^{t} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t} z_p^i\right) L' \qquad \forall p \in N \tag{5.24}$$

$$x_p - \sum_{j\in V_p} e_{ij} \leq \quad \sum_{i=1}^{t} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t} z_p^i\right) L' \qquad \forall p \in N \tag{5.25}$$

$$\sum_{p\in N} z_p^i = \quad n_i^* \qquad \forall i \in \{1, \ldots, t-1\} \tag{5.21}$$

From the above, we conclude that the following theorem holds.

Theorem 19. *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.

5.3 Simulation Results

5.3.1 Comparison between $\ell = \infty$ and $\ell = 2$

In Figure 5.3 we display our main results for *equal country sizes*, that is, when all countries are of the same size (see also Chapter 2). In Figure 5.3, we compare different solution concepts under different scenarios for $\ell = \infty$. Figure 5.3 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. As expected, using an arbitrary maximum cycle packing in each round makes the kidney exchange scheme significantly more unbalanced, with average total relative deviations over 4% for all initial allocations y. 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.

The above observations are in line with the results under the setting where $\ell = 2$ in Chapter 4. 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. From Figure 5.3 we see that the Banzhaf* value in the *lexmin+c* scenario provide the smallest deviations from the target allocations (as when $\ell = 2$ in Chapter 4). 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.

Figure 5.4 shows the same kind of results as Figure 5.3 but now for *varying country sizes*, that is, when we have small, medium and large countries, divided exactly as in Chapter 4 for $\ell = 2$ (see also Chapter 2). Note that subject to minor fluctuations we can draw the same conclusions from Figure 5.4 for varying country sizes as we did from Figure 5.3 for equal country sizes.

Figures 5.3 and 5.4 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 Figures 5.3 and 5.4 in Figure 5.5 and all the *d1+c* plots from Figures 5.3 and 5.4 in Figure 5.6. Figures 5.5 and 5.6 both show an ordering from the Banzhaf* value, which has the best performance, to the contribution value which has the worst in

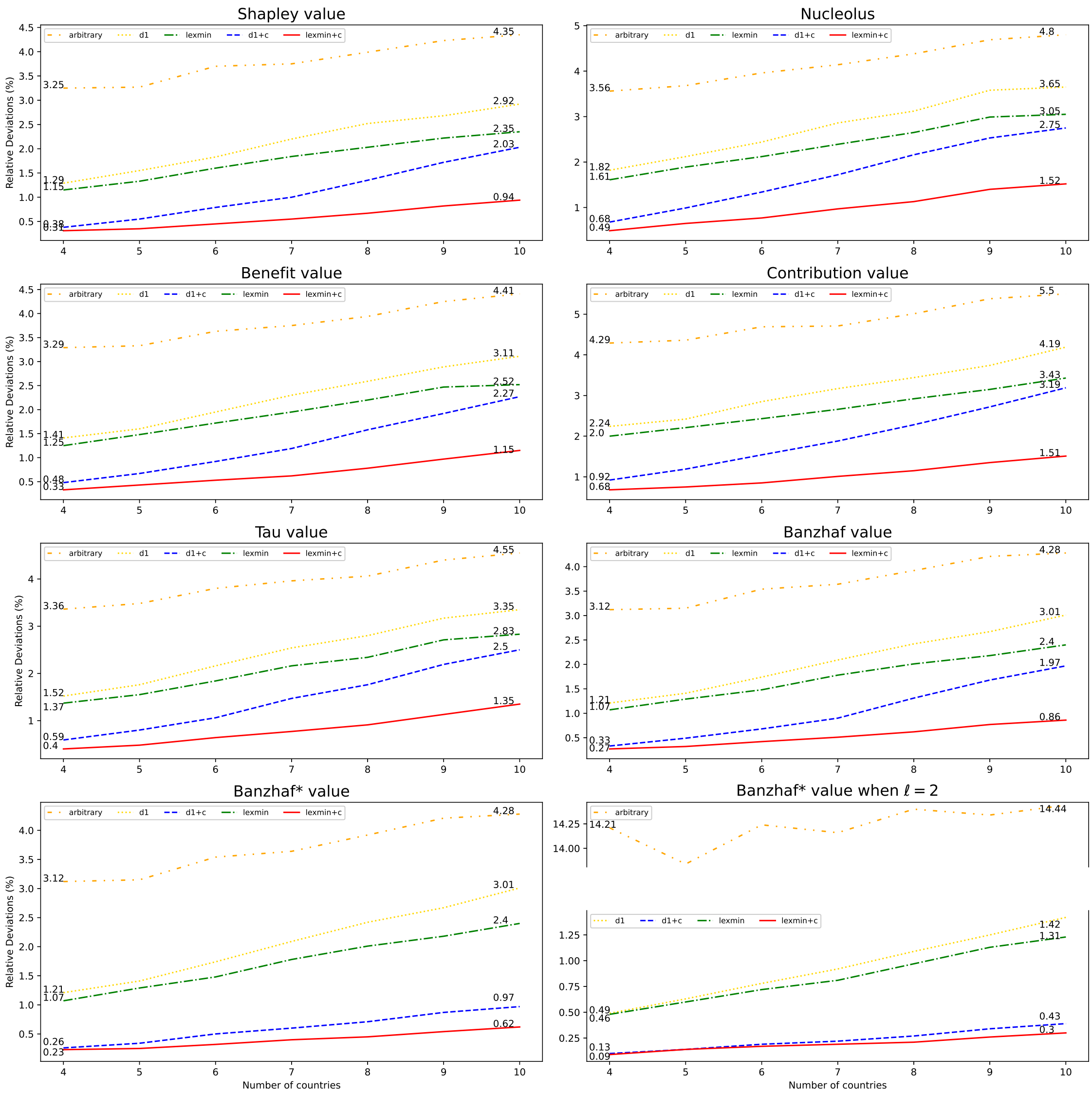

Figure 5.3: Average total relative deviations for each of the seven solution concepts under the five different scenarios for **equal** country sizes, where the number of countries n is ranging from 4 to 10. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely 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.

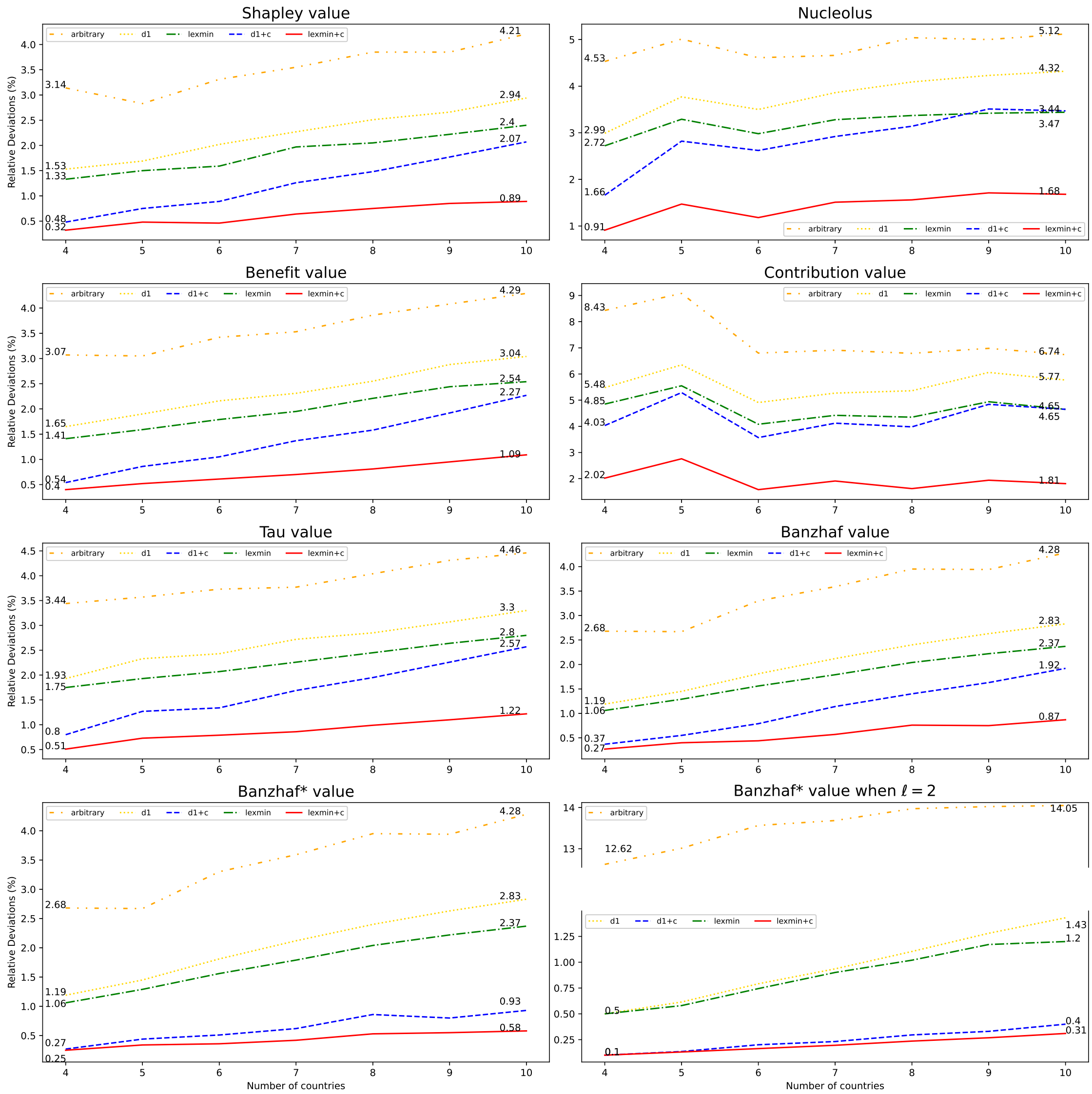

Figure 5.4: Average total relative deviations for each of the seven solution concepts under the five different scenarios for **varying** country sizes, where the number of countries n is ranging from 4 to 10. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which behaved best for $\ell = 2$.

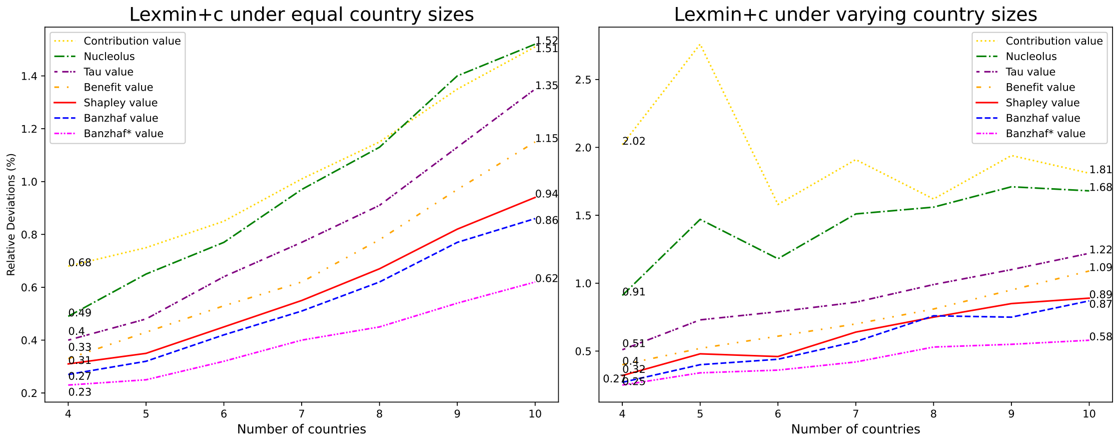

Figure 5.5: Average total relative deviations for all solution concepts in the *lexmin+c* scenario, where the number of countries n ranges from 4 to 10.

almost all cases for the two most important scenarios *lexmin+c* and *d1+c*.

We note that Figures 5.5 and 5.6 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. We refer to Figures 5.7–5.10 for the analogs of Figures 5.3–5.6 if we use the average maximum relative deviation as our evaluation measure.

Incomplete Instances. In the simulations for $\ell = 2$ in Chapter 4, 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-hour 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-hour 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

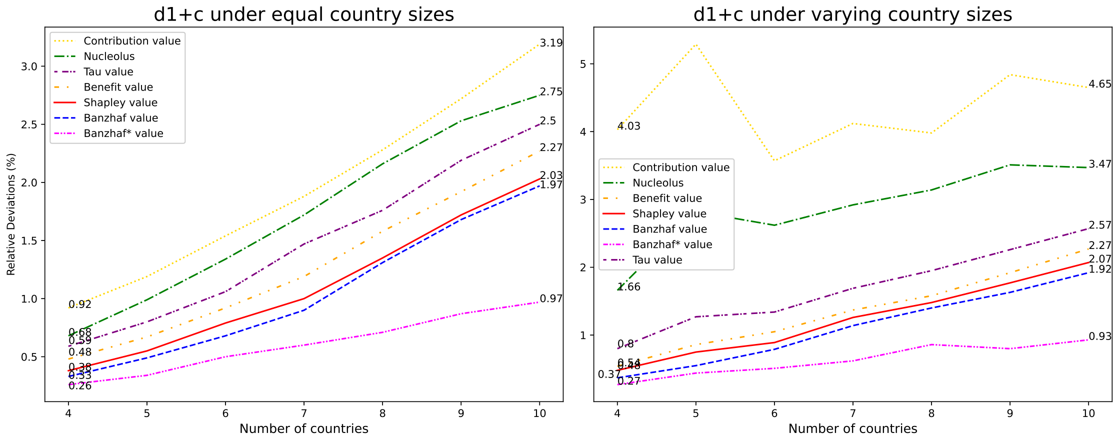

Figure 5.6: Average total relative deviations for all solution concepts in the *d1+c* scenario, where the number of countries n ranges from 4 to 10.

simulation instance occurred at *d1* / *d1+c* (for nine countries, using the tau value). Tables 5.1 and 5.2 summarize the distribution of incomplete simulation instances aggregated over different choices of solution concepts or scenarios. We refer to Tables 5.3–5.4 for a complete breakdown of the averages in both these tables.

Given 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 Figures 5.3–5.10 and Tables 5.5–5.8. We believe this was justified after doing some additional research. That is, we also constructed the same figures as Figures 5.3 and 5.4 but *without* the incomplete simulation instances; see Figures 5.11–5.12. It turned out that the largest percentage points difference in Figure 5.3 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 versus Cycle Length. When $\ell = \infty$ instead of $\ell = 2$, we expect more kidney transplants as exchange cycles may now have any size. In Tables 5.5–5.8 and Figure 5.13 we quantify this. From Tables 5.7–5.8 we see that when $\ell = \infty$ instead of $\ell = 2$, on average, it is possible to achieve 45%

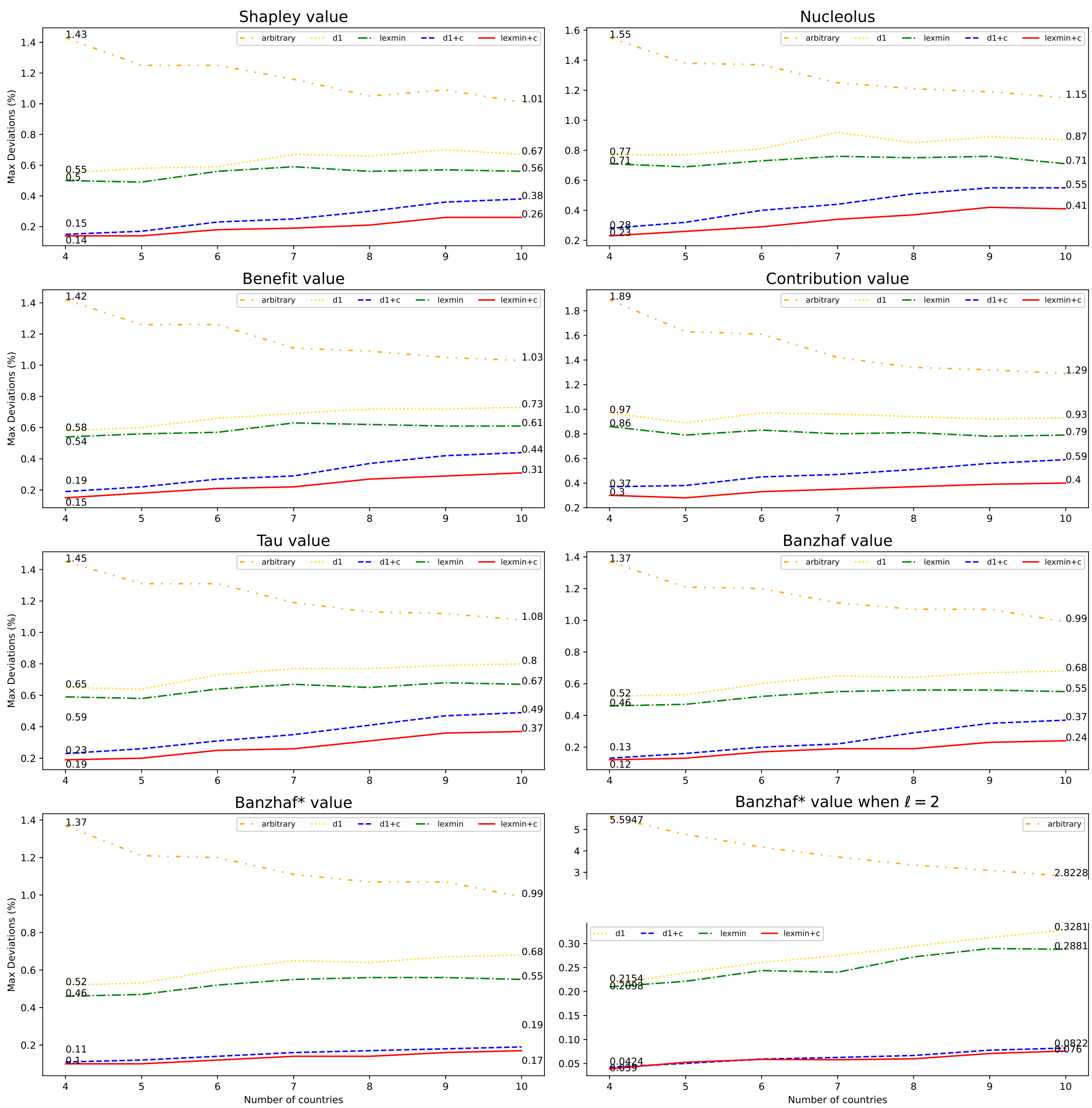

Figure 5.7: Average maximum relative deviations for each of the seven solution concepts under the five different scenarios for **equal** country sizes, where the number of countries n is ranging from 4 to 10. For comparison, the lower right figure displays a result from [3] for $\ell = 2$, namely for the Banzhaf* value, which behaved best for $\ell = 2$.

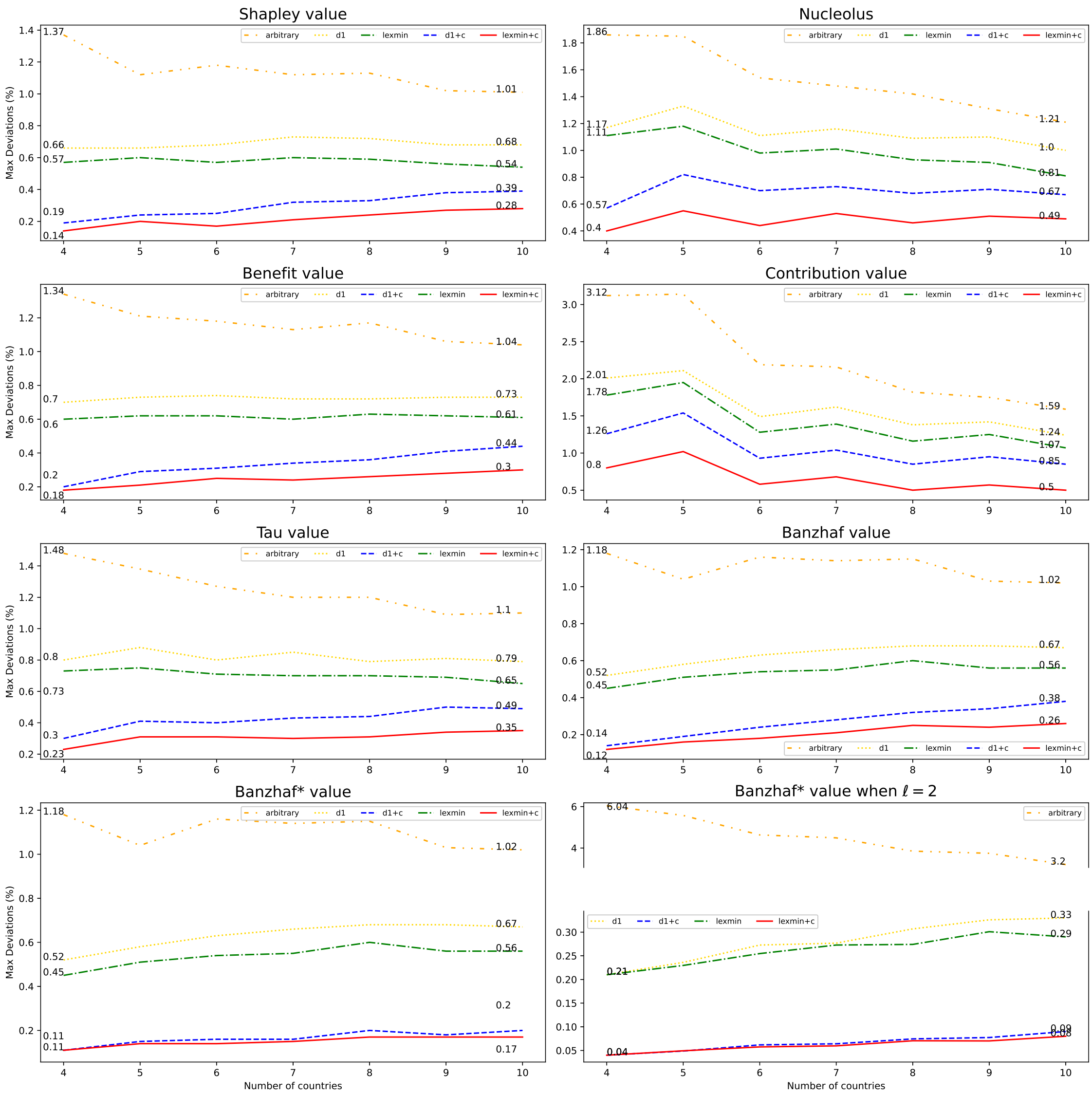

Figure 5.8: Average maximum relative deviations for each of the seven solution concepts under the five different scenarios for **varying** country sizes, where the number of countries n is ranging from 4 to 10. For comparison, the lower right figure displays a result from [3] for $\ell = 2$, namely for the Banzhaf* value, which behaved best for $\ell = 2$.

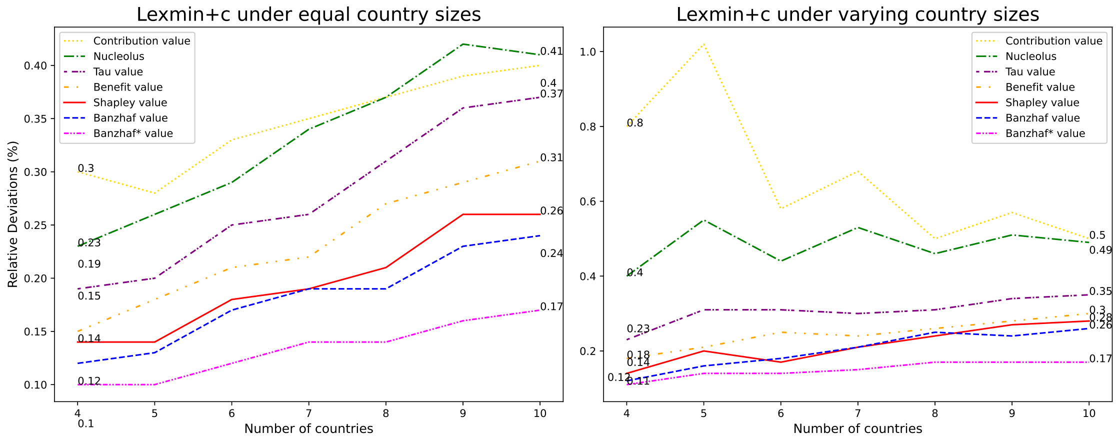

Figure 5.9: Average maximum relative deviations for all solution concepts in the *lexmin+c* scenario, where the number of countries n ranges from 4 to 10.

more kidney transplants when $n = 10$ across 7 solution concepts and 5 scenarios. This increase holds for both equal and varying country sizes, with the number of countries ranging from 4 to 10. This is irrespective of the chosen scenario, as the total number of kidney transplants is nearly identical throughout the five scenarios (as for $\ell = 2$). The $\ell = \infty$ setting is not realistic of course. Namely, Figure 5.13 show that long cycles, with even more than 400 vertices, may occur. Figure 5.13 also shows that these long cycles all happen in the first round. We note that before our experiments, the relation between increase in transplants versus increase in cycle length had not been researched by any simulations. Finding out about this relation through an extensive set of simulations was one of our main motivations.

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 Chapter 1 for a theoretical example of this behaviour when $\ell = 2$).

In Figures 5.14 and 5.15 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

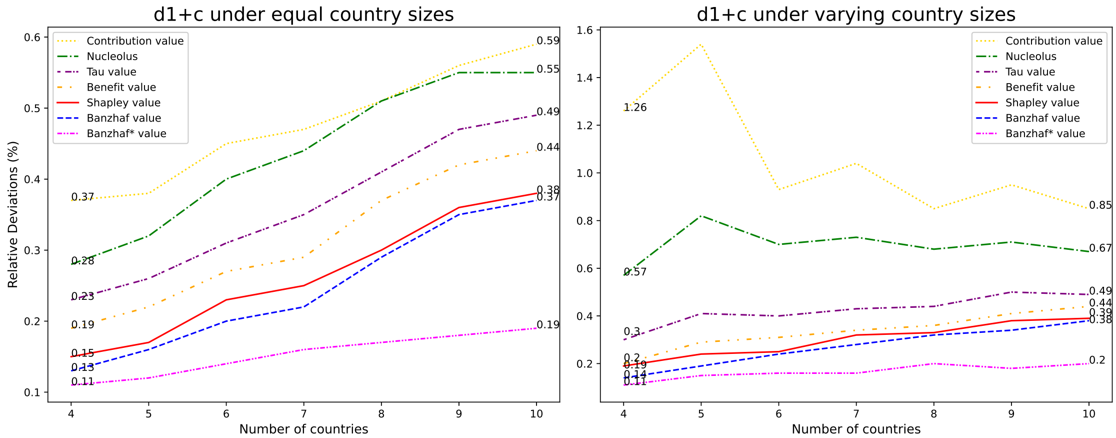

Figure 5.10: Average maximum relative deviations for all solution concepts in the *d1+c* scenario, where the number of countries n ranges from 4 to 10.

three scenarios *arbitary*, *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 Figures 5.3 and 5.4). 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 $\ell = 2$ in Chapter 4. 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 5.9 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 Chapter 4). We note that the relative ratio is slightly better for $\ell = 2$. This extends to varying countries as well; see Table 5.10. We also included the tables for the other scenarios as well for both equal and varying coutry sizes; see Tables 5.11–5.18. From these figures, we note

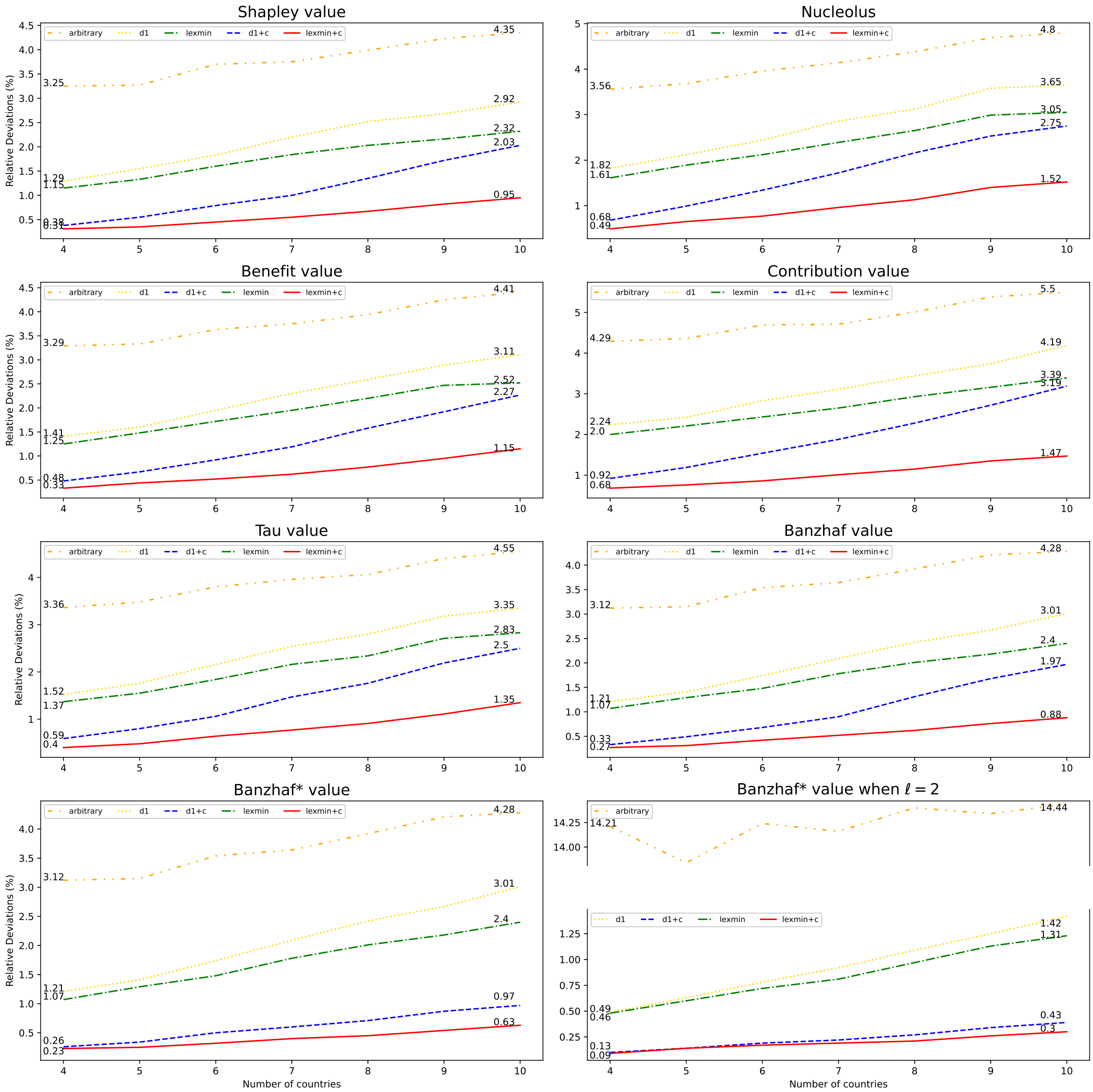

Figure 5.11: Average total relative deviations, leaving out the incomplete simulation instances under the five different scenarios for **equal** country sizes, where the number of countries ranges from 4 to 10. We note that this figure is identical to Figure 5.3 that includes all simulation instances.

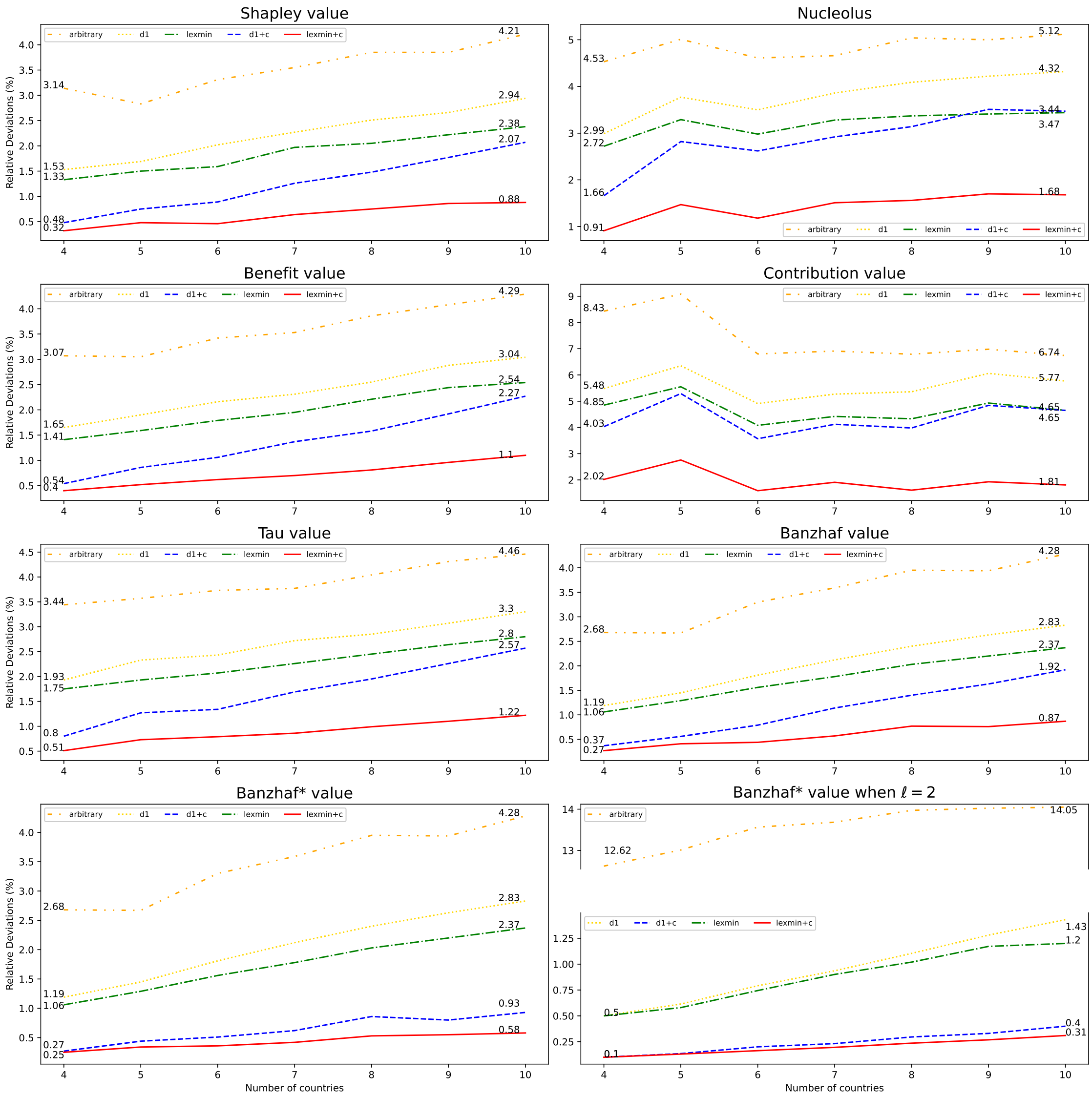

Figure 5.12: Average total relative deviations, leaving out the incomplete simulation instances under the five different scenarios for **varying** country sizes, where the number of countries ranges from 4 to 10. We note that this figure is identical to Figure 5.4 that includes all simulation instances.

#incomplete / 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 5.1: Average number of incomplete simulation instances for **equal** country sizes.

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.

5.3.2 Computation Time

Table 5.19 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 5.19 we also see that the computation time for the *arbitrary* scenario, which does not require selecting optimal solutions, 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*). Table 5.19 clearly shows that

#incomplete / 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 5.2: Average number of incomplete simulation instances for **varying** country sizes.

computing the ILP series for the *lexmin* and *lexmin+c* scenarios requires significantly more CPU time compared to the *d1* and *d1+c* scenarios. This is because the *lexmin* and *lexmin+c* scenarios involve longer series of ILPs, whereas for the *d1* and *d1+c* scenarios, only a single ILP_{d_1} needs to be solved. Given CPU time increases significantly, nearly doubling as n grows, we are only able to perform simulations for up to 10 countries, particularly due to the computational challenges of the hard-to-compute solution concepts and selecting weakly or strongly optimal solutions when n is large.

On a side note, we observe from Table 5.19 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_{d_1}) far more easily for $n = 6$ than for $n = 5$. We have no explanation for this; it may well be due to the nature of Gurobi and the computing node on which our simulation was executed, in which we have no insights.

5.4 Concluding Remarks

We formalized the IKEP when $\ell = \infty$ as the class of cooperative game theory called partitioned permutation games and proved a number of complexity results that contrast known results for partitioned matching games.

#incomplete / n	4	5	6	7	8	9	10	Total
Shapley value								
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	0	0	0	0
lexmin	0	0	0	3	2	14	7	26
lexmin+c	0	0	0	3	5	13	7	28
Nucleolus								
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	0	0	0	0
lexmin	0	0	0	0	0	0	0	0
lexmin+c	0	0	0	2	0	0	1	3
Benefit value								
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	0	0	0	0
lexmin	0	0	0	0	1	0	0	1
lexmin+c	1	1	5	1	2	3	0	13
Contribution value								
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	0	0	0	0
lexmin	1	1	2	1	1	2	4	12
lexmin+c	2	2	5	3	5	0	5	22
Tau value								
d1	0	0	0	0	0	1	0	1
d1+c	0	0	0	0	0	1	0	1
lexmin	0	0	0	0	0	1	0	1
lexmin+c	0	1	0	0	0	1	0	2
Banzhaf value								
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	0	0	0	0
lexmin	0	1	1	4	1	8	9	24
lexmin+c	0	2	5	3	1	9	9	29
Banzhaf* value								
d1+c	0	0	0	0	0	0	0	0
lexmin+c	0	0	0	0	0	0	1	1

Table 5.3: Complete breakdown of the number of incomplete simulation instances (out of 100) for the seven different solution concepts and the four different scenarios (excluding the arbitrary matching scenario, where no ILPs were used) for **equal** country sizes.

#incomplete / n	4	5	6	7	8	9	10	Total
				Shapley value				
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	0	0	0	0
lexmin	0	0	0	1	0	0	6	7
lexmin+c	0	0	0	1	0	2	2	5
				Nucleolus				
d1	0	0	0	0	0	1	0	1
d1+c	0	0	0	0	0	0	0	0
lexmin	0	0	0	0	0	1	0	1
lexmin+c	0	0	0	0	0	1	0	1
				Benefit value				
d1	0	0	0	0	0	0	0	0
d1+c	0	0	1	0	0	0	0	1
lexmin	0	0	0	0	0	0	0	0
lexmin+c	1	0	2	0	0	1	3	7
				Contribution value				
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	0	0	0	0
lexmin	0	0	1	0	1	1	0	3
lexmin+c	0	0	1	0	1	1	1	4
				Tau value				
d1	0	0	0	0	0	0	0	0
d1+c	0	0	0	0	1	0	0	1
lexmin	0	0	0	0	0	0	0	0
lexmin+c	0	0	0	0	0	0	0	0
				Banzhaf value				
d1	0	1	0	0	0	0	0	1
d1+c	0	1	0	0	0	0	0	1
lexmin	0	1	1	1	2	1	4	10
lexmin+c	0	1	0	0	3	1	4	9
				Banzhaf* value				
d1+c	0	1	0	0	0	0	0	1
lexmin+c	0	1	0	0	1	1	4	7

Table 5.4: Complete breakdown of the number of incomplete simulation instances (out of 100) for the seven different solution concepts and the four different scenarios (excluding the arbitrary matching scenario, where no ILPs were used) for **varying** country sizes.

Solution concepts/n		4	5	6	7	8	9	10
benefit value	*arbitrary*	1781.89	1780.60	1775.74	1771.30	1767.59	1763.44	1781.58
	d1	1782.15	1781.01	1775.31	1771.21	1767.38	1763.04	1781.65
	d1+c	1781.52	1780.01	1774.51	1770.22	1766.34	1762.37	1781.46
	lexmin	1782.25	1781.25	1775.47	1771.56	1767.87	1763.19	1782.58
	lexmin+c	1781.69	1780.12	1774.99	1770.71	1766.30	1761.76	1780.31
contribution value	*arbitrary*	1781.89	1780.60	1775.74	1771.30	1767.59	1763.44	1781.58
	d1	1782.70	1781.30	1775.88	1772.07	1768.31	1763.31	1781.52
	d1+c	1781.43	1780.41	1775.43	1770.90	1766.73	1762.68	1780.50
	lexmin	1782.20	1781.42	1775.86	1771.93	1768.04	1763.31	1782.52
	lexmin+c	1781.80	1780.58	1774.87	1770.21	1766.24	1762.30	1780.05
Nucleolus	*arbitrary*	1781.89	1780.60	1775.67	1771.43	1767.64	1763.17	1781.58
	d1	1781.79	1780.99	1775.76	1771.53	1767.26	1762.97	1781.80
	d1+c	1781.24	1779.79	1775.30	1770.68	1767.00	1762.30	1780.45
	lexmin	1782.02	1781.11	1775.30	1771.99	1767.42	1763.19	1781.98
	lexmin+c	1781.43	1779.96	1774.86	1770.74	1765.81	1761.97	1780.12
Shapley value	*arbitrary*	1781.89	1780.60	1775.78	1771.38	1767.57	1763.14	1781.58
	d1	1782.25	1781.22	1775.81	1771.75	1767.06	1763.34	1781.54
	d1+c	1781.67	1780.88	1775.45	1771.11	1766.68	1762.22	1780.92
	lexmin	1782.43	1781.90	1775.38	1771.73	1767.90	1763.37	1782.02
	lexmin+c	1781.61	1780.79	1775.50	1771.13	1767.13	1762.65	1780.57
Banzhaf value	*arbitrary*	1781.89	1780.60	1775.71	1771.46	1767.50	1763.31	1781.58
	d1	1782.14	1781.14	1775.34	1771.91	1767.57	1763.15	1781.83
	d1+c	1781.67	1780.33	1774.87	1771.37	1766.51	1762.34	1780.38
	lexmin	1782.15	1781.15	1775.62	1771.97	1767.99	1763.16	1782.46
	lexmin+c	1781.96	1780.58	1774.87	1771.08	1766.94	1762.79	1780.66
Banzhaf* value	*arbitrary*	1781.89	1780.60	1775.71	1771.46	1767.50	1763.31	1781.58
	d1	1782.14	1781.14	1775.34	1771.91	1767.57	1763.15	1781.83
	d1+c	1781.77	1780.03	1774.83	1771.38	1767.00	1762.71	1781.08
	lexmin	1782.15	1781.15	1775.62	1771.97	1767.99	1763.16	1782.46
	lexmin+c	1782.04	1781.03	1774.78	1771.32	1767.24	1762.70	1781.24
Tau value	*arbitrary*	1781.89	1780.60	1775.65	1771.36	1767.34	1763.62	1781.58
	d1	1781.87	1780.67	1775.14	1771.22	1767.35	1762.70	1782.22
	d1+c	1781.80	1780.08	1774.28	1770.34	1766.54	1761.85	1780.90
	lexmin	1782.41	1780.63	1775.40	1771.60	1767.60	1763.42	1782.12
	lexmin+c	1781.13	1780.01	1774.42	1770.97	1766.31	1761.81	1780.45

Table 5.5: Average number of kidney transplants for 6 solution concepts along with the Banzhaf* value across 5 scenarios where $\ell = \infty$ with the number of **equal-sized** countries ranging from 4 to 10. See Chapter 4 for full results in the case where $\ell = 2$.

Solution concepts/n		4	5	6	7	8	9	10
benefit value	*arbitrary*	1775.56	1780.86	1765.16	1745.57	1754.08	1731.64	1763.57
	d1	1775.07	1781.09	1765.15	1745.78	1753.74	1731.19	1763.39
	d1+c	1774.24	1780.35	1763.89	1745.75	1753.46	1731.12	1763.37
	lexmin	1775.51	1781.39	1764.70	1745.66	1753.83	1731.51	1763.73
	lexmin+c	1774.38	1780.40	1763.87	1744.95	1752.83	1730.72	1762.45
contribution value	*arbitrary*	1775.56	1780.86	1765.16	1745.57	1754.08	1731.64	1763.57
	d1	1775.17	1780.39	1765.10	1746.20	1753.91	1731.31	1763.45
	d1+c	1774.09	1780.19	1763.82	1745.31	1753.04	1730.92	1762.90
	lexmin	1775.52	1780.88	1765.52	1746.61	1754.36	1732.38	1763.80
	lexmin+c	1773.54	1778.83	1763.59	1744.50	1752.82	1730.44	1761.84
Nucleolus	*arbitrary*	1774.12	1781.25	1764.39	1746.27	1753.55	1731.72	1763.66
	d1	1774.18	1780.85	1764.08	1746.64	1753.38	1731.13	1764.02
	d1+c	1773.08	1779.98	1763.28	1745.61	1752.75	1730.71	1762.79
	lexmin	1774.23	1781.34	1764.32	1746.04	1753.41	1731.24	1763.74
	lexmin+c	1772.92	1780.06	1762.92	1745.32	1751.82	1730.22	1762.19
Shapley value	*arbitrary*	1774.02	1781.25	1764.42	1746.17	1753.44	1731.64	1764.04
	d1	1774.38	1781.17	1763.92	1746.27	1753.60	1731.43	1763.09
	d1+c	1773.47	1779.94	1763.70	1745.30	1753.14	1731.43	1762.99
	lexmin	1774.24	1781.50	1764.67	1745.89	1753.84	1731.73	1763.71
	lexmin+c	1773.51	1780.54	1764.09	1745.53	1752.84	1730.54	1762.60
Banzhaf value	*arbitrary*	1775.90	1780.86	1765.00	1745.64	1754.25	1731.39	1763.24
	d1	1775.17	1781.00	1764.38	1746.30	1754.43	1731.38	1763.34
	d1+c	1775.25	1780.41	1764.60	1745.27	1753.50	1731.05	1762.85
	lexmin	1775.43	1781.02	1765.33	1745.74	1754.33	1732.17	1763.47
	lexmin+c	1774.90	1780.01	1764.23	1745.56	1753.40	1731.02	1762.85
Banzhaf* value	*arbitrary*	1775.90	1780.86	1765.00	1745.64	1754.25	1731.39	1763.24
	d1	1775.17	1781.00	1764.38	1746.30	1754.43	1731.38	1763.34
	d1+c	1774.62	1780.69	1764.32	1745.91	1753.87	1731.27	1763.09
	lexmin	1775.43	1781.02	1765.33	1745.74	1754.33	1732.17	1763.47
	lexmin+c	1775.02	1780.43	1764.81	1745.63	1753.82	1731.49	1762.97
Tau value	*arbitrary*	1775.46	1780.86	1765.12	1745.75	1754.50	1731.55	1763.27
	d1	1774.72	1780.85	1764.86	1745.48	1753.78	1731.42	1763.54
	d1+c	1774.50	1779.87	1763.82	1746.00	1753.06	1731.11	1762.99
	lexmin	1774.90	1781.11	1764.47	1745.77	1754.02	1731.48	1763.62
	lexmin+c	1774.46	1779.77	1763.52	1745.32	1753.29	1730.23	1761.84

Table 5.6: Average number of kidney transplants for 6 solution concepts along with the Banzhaf* value across 5 scenarios where $\ell = \infty$ with the number of **varying-sized** countries ranging from 4 to 10.

Solution concepts/scenarios		*arbitrary*	*d1*	*d1+c*	*lemxin*	*lexmin+c*
	benefit value	1781.58	1781.65	1781.46	1782.58	1780.31
	contribution value	1781.58	1781.52	1780.50	1782.52	1780.05
	Nucleolus	1781.58	1781.80	1780.45	1781.98	1780.12
$\ell = \infty$	Shapley value	1781.58	1781.54	1780.92	1782.02	1780.57
	Banzhaf value	1781.58	1781.83	1780.38	1782.46	1780.66
	Banzhaf* value	1781.58	1781.83	1781.08	1782.46	1781.24
	Tau value	1781.58	1782.22	1780.90	1782.12	1780.45
	benefit value	1221.38	1222.4	1223.44	1222.54	1223.6
	contribution value	1221.38	1222.58	1222.56	1223.24	1222.66
	Nucleolus	1221.38	1222.28	1221.18	1221.84	1221.66
$\ell = 2$	Shapley value	1221.38	1221.6	1221.2	1223.04	1222.32
	Banzhaf value	1221.38	1222.62	1221.26	1220.9	1221.98
	Banzhaf* value	1221.38	1222.62	1220.86	1220.9	1221.82
	Tau value	1221.38	1223.96	1223.24	1224.26	1222.72

Table 5.7: Average number of kidney transplants under 6 solution concepts along with the Banzhaf* value across 5 scenarios with 10 **equal-sized** countries.

Our new results guided our simulations for IKEPs with up to ten countries, with exchange bound $\ell = \infty$. Our simulations showed a significant improvement in the total number of kidney transplants over the case where $\ell = 2$ (see Chapter 4) at the expense of long cycles. In our simulations, we first let all countries be of the same size. We then examined in the same way as done in Chapter 4, whether our conclusions would change for varying country sizes. As in Chapter 4, this turned out not to be the case.

Solution concepts/scenarios		*arbitrary*	*d1*	*d1+c*	*lemxin*	*lexmin+c*
	benefit value	1763.57	1763.39	1763.37	1763.73	1762.45
	contribution value	1763.57	1763.45	1762.90	1763.80	1761.84
	Nucleolus	1763.66	1764.02	1762.79	1763.74	1762.19
$\ell = \infty$	Shapley value	1764.04	1763.09	1762.99	1763.71	1762.60
	Banzhaf value	1763.24	1763.34	1762.85	1763.47	1762.85
	Banzhaf* value	1763.24	1763.34	1763.09	1763.47	1762.97
	Tau value	1763.27	1763.54	1762.99	1763.62	1761.84
	benefit value	1203.8	1206.14	1205.92	1205.06	1205.88
	contribution value	1204.06	1205.7	1205.46	1205.36	1206.16
	Nucleolus	1204.06	1205.34	1205.38	1205.52	1207
$\ell = 2$	Shapley value	1204.06	1205.7	1206.64	1206.34	1205.48
	Banzhaf value	1204.06	1206.14	1205.56	1205.88	1206.34
	Banzhaf* value	1204.06	1206.14	1206.34	1205.88	1205.92
	Tau value	1204.06	1206.86	1205.92	1207.32	1206.3

Table 5.8: Average number of kidney transplants under 6 solution concepts along with the Banzhaf* value across 5 scenarios with 10 **varying-sized** countries.

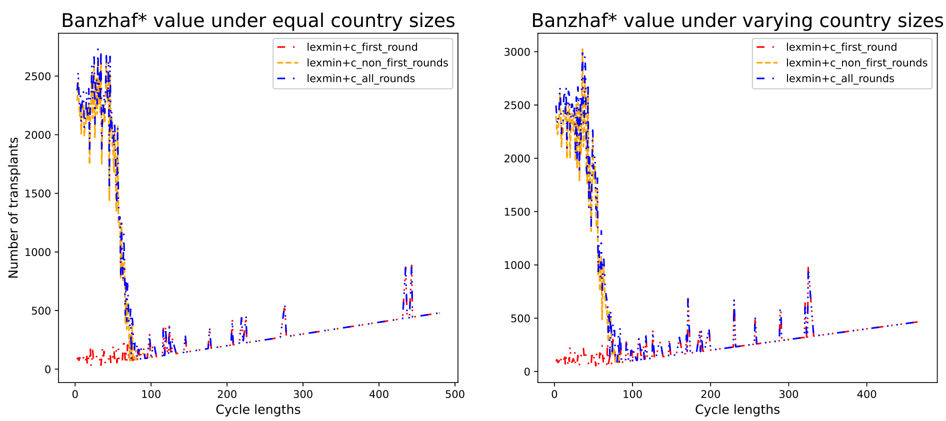

Figure 5.13: Cycle distribution for the Banzhaf* value in the *lexmin+c* scenario, where the x-axis represents the cycle length, and the y-axis shows the number of kidney transplants involved in a cycle of that length.

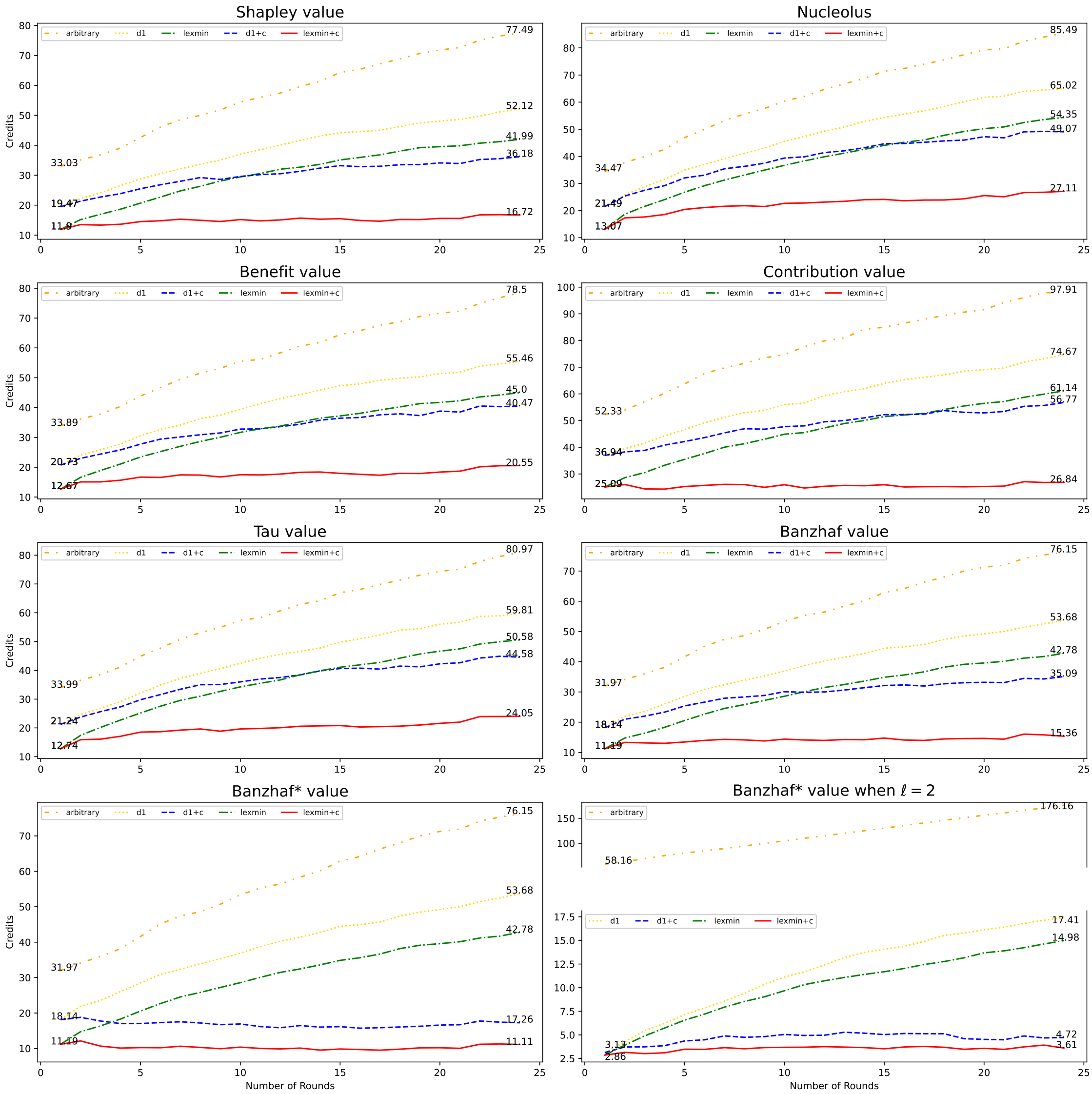

Figure 5.14: Average credits for each of the seven solution concepts under the five different scenarios for **equal** country sizes, where the number of countries is $n = 10$ and the period ranges from 1 to 24. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which also behaved best for $\ell = 2$.

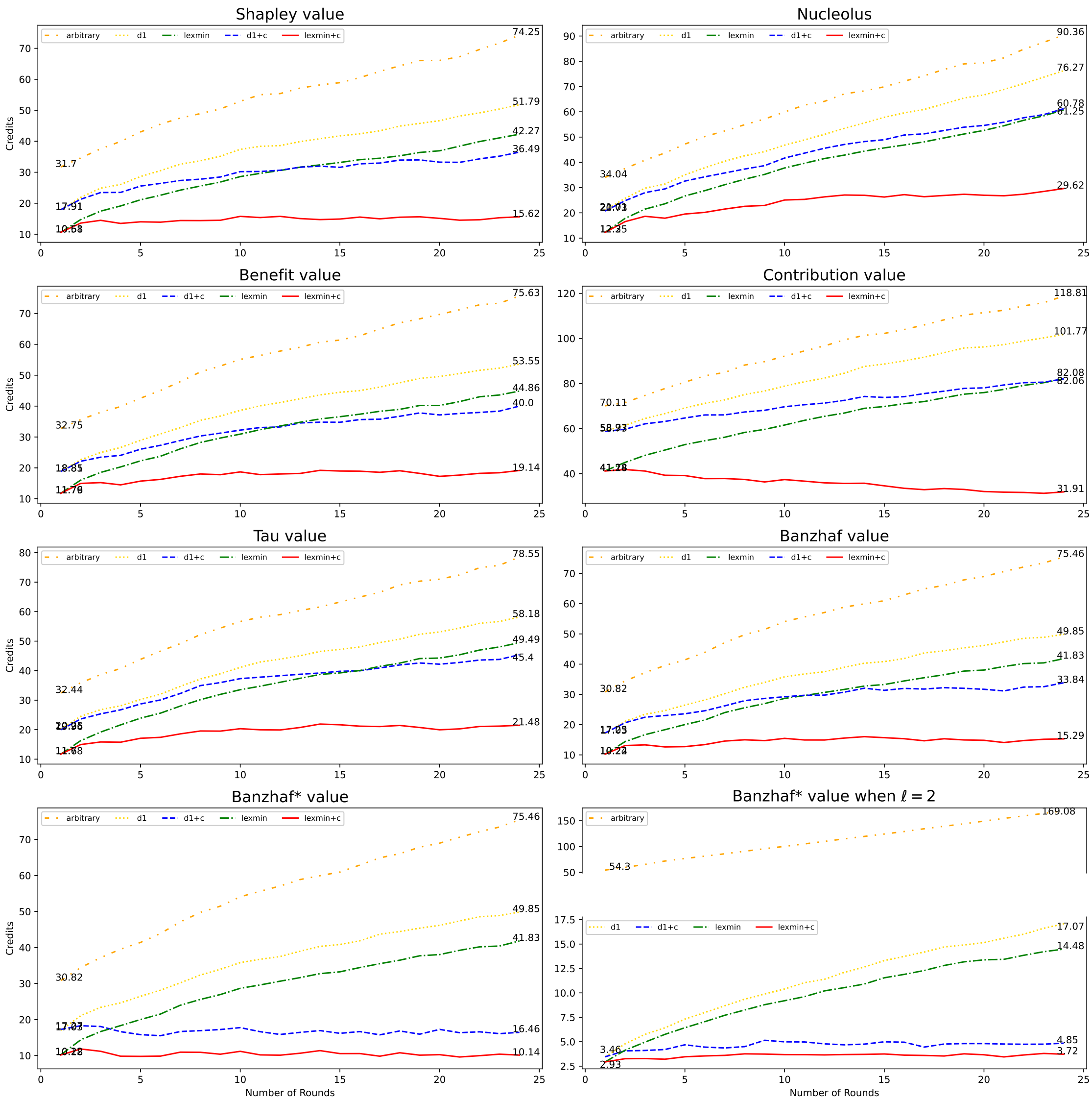

Figure 5.15: Average credits for each of the seven solution concepts under the five different scenarios for **varying** country sizes, where the number of countries is $n = 10$ and the period ranges from 1 to 24. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which also behaved best for $\ell = 2$.

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%

Table 5.9: 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	$\ell = \infty$	44.43%	40.58%	37.71%	33.29%	32.18%	31.99%	31.07%
	$\ell = 2$	42.82%	38.62%	35.31%	31.05%	27.83%	26.96%	24.81%
	difference	1.61%	1.96%	2.40%	2.24%	4.35%	5.03%	6.26%
Banzhaf value	$\ell = \infty$	43.75%	39.53%	40.34%	35.92%	32.83%	31.31%	30.06%
	$\ell = 2$	43.18%	37.15%	33.94%	31.19%	30.13%	27.61%	24.50%
	difference	0.57%	2.38%	6.40%	4.73%	2.70%	3.70%	5.56%
nucleolus	$\ell = \infty$	43.99%	37.67%	37.67%	35.10%	29.67%	30.08%	29.06%
	$\ell = 2$	46.06%	39.05%	35.94%	34.13%	31.35%	30.26%	27.10%
	difference	-2.07%	-1.38%	1.73%	0.98%	-1.68%	-0.18%	1.96%
tau value	$\ell = \infty$	45.63%	42.38%	39.61%	34.47%	31.25%	30.81%	28.82%
	$\ell = 2$	41.92%	38.50%	35.39%	36.04%	31.29%	28.05%	26.06%
	difference	3.72%	3.88%	4.22%	-1.57%	-0.04%	2.77%	2.76%
benefit value	$\ell = \infty$	45.53%	40.96%	40.33%	34.19%	32.73%	29.63%	27.95%
	$\ell = 2$	42.31%	37.14%	33.68%	31.91%	30.41%	28.76%	25.05%
	difference	3.22%	3.82%	6.65%	2.28%	2.33%	0.87%	2.91%
contribution value	$\ell = \infty$	39.67%	36.83%	36.54%	35.61%	31.01%	29.51%	27.45%
	$\ell = 2$	43.57%	36.74%	36.41%	32.65%	28.38%	27.49%	25.31%
	difference	-3.91%	0.09%	0.13%	2.96%	2.63%	2.02%	2.15%
Banzhaf* value	$\ell = \infty$	44.26%	40.10%	38.69%	36.31%	31.90%	31.17%	29.39%
	$\ell = 2$	41.60%	35.21%	31.94%	28.43%	27.12%	22.61%	21.53%
	difference	2.66%	4.89%	6.75%	7.88%	4.78%	8.56%	7.86%

Table 5.10: Relative ratios for **varying** country sizes under *lexmin+c*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	39.35%	30.99%	29.02%	25.07%	22.48%	21.05%	18.76%
	$\ell = 2$	43.38%	38.19%	33.14%	30.28%	26.48%	23.92%	22.07%
	difference	-4.03%	-7.19%	-4.12%	-5.21%	-4.00%	-2.87%	-3.31%
Banzhaf value	$\ell = \infty$	39.74%	33.02%	29.33%	24.52%	22.23%	20.96%	18.82%
	$\ell = 2$	42.73%	35.31%	30.25%	27.88%	25.00%	22.51%	21.07%
	difference	-2.99%	-2.29%	-0.92%	-3.36%	-2.77%	-1.55%	-2.25%
nucleolus	$\ell = \infty$	40.64%	32.40%	29.62%	25.55%	23.57%	21.75%	19.80%
	$\ell = 2$	42.33%	37.23%	31.97%	29.21%	25.00%	23.09%	20.56%
	difference	-1.69%	-4.83%	-2.36%	-3.65%	-1.43%	-1.33%	-0.76%
tau value	$\ell = \infty$	39.97%	32.84%	29.41%	24.10%	23.24%	21.48%	19.39%
	$\ell = 2$	56.85%	46.43%	37.42%	31.95%	29.29%	26.97%	24.74%
	difference	-16.88%	-13.59%	-8.01%	-7.85%	-6.05%	-5.49%	-5.35%
benefit value	$\ell = \infty$	39.59%	32.02%	29.45%	24.35%	23.49%	21.67%	19.32%
	$\ell = 2$	41.36%	38.72%	33.50%	28.94%	26.86%	23.02%	20.49%
	difference	-1.77%	-6.71%	-4.05%	-4.59%	-3.37%	-1.35%	-1.18%
contribution value	$\ell = \infty$	39.79%	31.56%	29.06%	25.00%	22.37%	20.50%	18.45%
	$\ell = 2$	42.38%	38.24%	33.41%	29.89%	26.22%	23.84%	21.14%
	difference	-2.59%	-6.68%	-4.36%	-4.89%	-3.85%	-3.33%	-2.69%
Banzhaf* value	$\ell = \infty$	41.21%	34.16%	28.94%	26.12%	23.40%	20.99%	19.57%
	$\ell = 2$	42.40%	35.79%	31.21%	28.45%	24.70%	22.88%	21.08%
	difference	-1.19%	-1.62%	-2.27%	-2.33%	-1.30%	-1.89%	-1.51%

Table 5.11: Relative ratios for **equal** country sizes under *d1+c*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	39.18%	31.66%	27.75%	25.12%	22.51%	21.39%	18.83%
	$\ell = 2$	42.00%	37.21%	31.11%	27.88%	25.48%	22.45%	19.78%
	difference	-2.82%	-5.56%	-3.35%	-2.76%	-2.98%	-1.05%	-0.95%
Banzhaf value	$\ell = \infty$	38.21%	33.81%	30.03%	25.03%	22.51%	20.96%	19.55%
	$\ell = 2$	41.64%	37.85%	30.53%	27.92%	25.37%	22.68%	21.09%
	difference	-3.42%	-4.04%	-0.49%	-2.89%	-2.85%	-1.72%	-1.54%
nucleolus	$\ell = \infty$	34.29%	28.92%	26.69%	25.16%	21.51%	20.13%	19.42%
	$\ell = 2$	39.79%	32.16%	26.26%	23.07%	21.66%	20.33%	18.24%
	difference	-5.50%	-3.24%	0.43%	2.10%	-0.15%	-0.20%	1.17%
tau value	$\ell = \infty$	37.66%	32.07%	29.60%	25.25%	22.80%	21.94%	19.10%
	$\ell = 2$	46.67%	35.53%	30.91%	29.27%	24.75%	21.81%	20.12%
	difference	-9.01%	-3.45%	-1.31%	-4.02%	-1.96%	0.13%	-1.02%
benefit value	$\ell = \infty$	37.56%	33.48%	29.28%	25.08%	22.56%	21.38%	19.45%
	$\ell = 2$	41.50%	34.41%	32.13%	26.41%	24.49%	21.74%	19.71%
	difference	-3.94%	-0.93%	-2.85%	-1.32%	-1.93%	-0.35%	-0.26%
contribution value	$\ell = \infty$	31.23%	29.10%	26.19%	25.19%	21.45%	19.59%	18.38%
	$\ell = 2$	38.51%	32.26%	28.71%	25.60%	22.88%	20.89%	19.32%
	difference	-7.27%	-3.16%	-2.53%	-0.42%	-1.42%	-1.30%	-0.94%
Banzhaf* value	$\ell = \infty$	38.21%	33.81%	30.03%	25.03%	22.51%	20.96%	19.55%
	$\ell = 2$	41.64%	37.85%	30.53%	27.92%	25.37%	22.68%	21.09%
	difference	-3.42%	-4.04%	-0.49%	-2.89%	-2.85%	-1.72%	-1.54%

Table 5.12: Relative ratios for **varying** country sizes under *d1+c*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	43.66%	36.50%	35.01%	31.87%	27.73%	25.86%	23.59%
	$\ell = 2$	43.02%	37.65%	34.21%	30.58%	27.36%	25.96%	23.11%
	difference	0.64%	-1.15%	0.80%	1.29%	0.37%	-0.10%	0.48%
Banzhaf value	$\ell = \infty$	43.04%	36.34%	35.15%	31.14%	27.69%	25.74%	22.87%
	$\ell = 2$	43.71%	36.90%	33.85%	29.67%	28.06%	25.64%	23.42%
	difference	-0.66%	-0.56%	1.31%	1.48%	-0.37%	0.10%	-0.56%
nucleolus	$\ell = \infty$	43.97%	36.83%	34.53%	31.64%	28.27%	25.31%	23.27%
	$\ell = 2$	44.57%	36.52%	34.48%	29.52%	27.85%	26.20%	24.17%
	difference	-0.59%	0.31%	0.06%	2.12%	0.41%	-0.89%	-0.90%
tau value	$\ell = \infty$	43.55%	37.36%	34.61%	31.16%	27.79%	24.95%	23.68%
	$\ell = 2$	44.00%	39.67%	33.87%	30.85%	26.95%	25.18%	23.49%
	difference	-0.45%	-2.31%	0.73%	0.31%	0.84%	-0.23%	0.18%
benefit value	$\ell = \infty$	42.98%	37.59%	33.34%	32.33%	28.17%	24.73%	24.22%
	$\ell = 2$	42.98%	39.41%	32.90%	30.28%	26.75%	25.73%	23.74%
	difference	0.00%	-1.83%	0.44%	2.05%	1.42%	-1.00%	0.47%
contribution value	$\ell = \infty$	42.92%	35.82%	34.11%	30.26%	27.68%	24.79%	23.13%
	$\ell = 2$	44.52%	37.22%	33.77%	31.10%	28.12%	26.39%	22.57%
	difference	-1.60%	-1.40%	0.33%	-0.84%	-0.44%	-1.60%	0.56%
Banzhaf* value	$\ell = \infty$	43.04%	36.34%	35.15%	31.14%	27.69%	25.74%	22.87%
	$\ell = 2$	43.71%	36.90%	33.85%	29.67%	28.06%	25.64%	23.42%
	difference	-0.66%	-0.56%	1.31%	1.48%	-0.37%	0.10%	-0.56%

Table 5.13: Relative ratios for **equal** country sizes under *lexmin*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	42.41%	39.85%	35.90%	30.36%	28.87%	25.47%	22.75%
	$\ell = 2$	45.17%	37.62%	33.77%	29.46%	27.20%	25.13%	23.66%
	difference	-2.76%	2.23%	2.14%	0.90%	1.67%	0.34%	-0.90%
Banzhaf value	$\ell = \infty$	42.44%	39.50%	34.98%	30.66%	29.63%	25.36%	23.76%
	$\ell = 2$	42.08%	39.60%	34.42%	30.31%	26.87%	25.73%	24.03%
	difference	0.36%	-0.11%	0.56%	0.35%	2.75%	-0.37%	-0.26%
nucleolus	$\ell = \infty$	40.74%	35.71%	33.04%	30.84%	27.59%	26.51%	23.65%
	$\ell = 2$	42.53%	38.17%	34.08%	29.74%	28.51%	26.64%	25.27%
	difference	-1.80%	-2.46%	-1.05%	1.10%	-0.92%	-0.14%	-1.62%
tau value	$\ell = \infty$	41.81%	38.72%	34.16%	30.77%	28.55%	26.31%	23.12%
	$\ell = 2$	45.29%	37.62%	34.39%	29.38%	27.49%	25.13%	23.88%
	difference	-3.48%	1.11%	-0.22%	1.38%	1.06%	1.18%	-0.76%
benefit value	$\ell = \infty$	42.52%	39.28%	34.44%	30.99%	28.39%	25.26%	23.82%
	$\ell = 2$	43.70%	37.10%	33.92%	30.51%	27.40%	25.11%	23.98%
	difference	-1.18%	2.18%	0.51%	0.47%	0.99%	0.15%	-0.16%
contribution value	$\ell = \infty$	36.57%	35.10%	31.46%	31.39%	26.63%	25.40%	23.02%
	$\ell = 2$	37.19%	33.40%	32.36%	30.46%	28.79%	27.69%	24.80%
	difference	-0.63%	1.70%	-0.89%	0.93%	-2.16%	-2.29%	-1.78%
Banzhaf* value	$\ell = \infty$	42.44%	39.50%	34.98%	30.66%	29.63%	25.36%	23.76%
	$\ell = 2$	42.08%	39.60%	34.42%	30.31%	26.87%	25.73%	24.03%
	difference	0.36%	-0.11%	0.56%	0.35%	2.75%	-0.37%	-0.26%

Table 5.14: Relative ratios for **varying** country sizes under *lexmin*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	42.66%	37.54%	32.40%	30.34%	26.29%	26.12%	22.79%
	$\ell = 2$	42.57%	37.54%	34.20%	30.24%	27.99%	24.22%	22.87%
	difference	0.09%	0.00%	-1.81%	0.09%	-1.70%	1.90%	-0.08%
Banzhaf value	$\ell = \infty$	43.36%	37.84%	34.55%	31.11%	26.35%	24.97%	22.70%
	$\ell = 2$	43.96%	37.92%	33.42%	29.87%	27.04%	25.00%	23.11%
	difference	-0.60%	-0.08%	1.13%	1.24%	-0.69%	-0.03%	-0.40%
nucleolus	$\ell = \infty$	42.18%	36.51%	33.15%	31.99%	27.25%	24.81%	23.85%
	$\ell = 2$	45.00%	37.98%	32.49%	29.57%	27.37%	25.42%	22.73%
	difference	-2.82%	-1.47%	0.66%	2.42%	-0.13%	-0.61%	1.12%
tau value	$\ell = \infty$	42.54%	36.46%	33.55%	30.46%	27.41%	24.95%	23.71%
	$\ell = 2$	42.68%	37.32%	33.21%	29.89%	28.12%	25.07%	23.42%
	difference	-0.13%	-0.86%	0.34%	0.57%	-0.71%	-0.12%	0.28%
benefit value	$\ell = \infty$	41.25%	37.80%	33.96%	30.01%	27.99%	25.01%	23.54%
	$\ell = 2$	43.67%	37.81%	33.68%	29.98%	27.14%	24.64%	23.92%
	difference	-2.42%	-0.01%	0.28%	0.02%	0.85%	0.36%	-0.39%
contribution value	$\ell = \infty$	43.32%	36.90%	34.20%	30.21%	27.44%	24.70%	22.31%
	$\ell = 2$	42.91%	37.76%	32.70%	30.33%	27.63%	25.55%	23.26%
	difference	0.40%	-0.86%	1.49%	-0.11%	-0.19%	-0.85%	-0.96%
Banzhaf* value	$\ell = \infty$	43.36%	37.84%	34.55%	31.11%	26.35%	24.97%	22.70%
	$\ell = 2$	43.96%	37.92%	33.42%	29.87%	27.04%	25.00%	23.11%
	difference	-0.60%	-0.08%	1.13%	1.24%	-0.69%	-0.03%	-0.40%

Table 5.15: Relative ratios for **equal** country sizes under *d1*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	42.74%	38.66%	33.65%	31.96%	28.69%	25.47%	23.17%
	$\ell = 2$	44.61%	38.56%	33.07%	29.70%	27.57%	24.85%	22.91%
	difference	-1.87%	0.10%	0.58%	2.26%	1.13%	0.62%	0.26%
Banzhaf value	$\ell = \infty$	44.07%	39.81%	34.52%	31.04%	28.49%	25.85%	23.58%
	$\ell = 2$	42.96%	38.70%	34.52%	29.45%	27.90%	25.48%	23.01%
	difference	1.11%	1.11%	0.00%	1.60%	0.59%	0.38%	0.57%
nucleolus	$\ell = \infty$	38.99%	35.17%	31.74%	30.03%	26.66%	25.90%	23.07%
	$\ell = 2$	41.91%	36.93%	33.97%	29.82%	27.56%	25.70%	23.59%
	difference	-2.93%	-1.75%	-2.23%	0.21%	-0.89%	0.20%	-0.52%
tau value	$\ell = \infty$	41.58%	37.88%	33.09%	31.44%	27.77%	26.33%	23.96%
	$\ell = 2$	48.20%	38.31%	32.66%	30.14%	26.17%	25.75%	23.44%
	difference	-6.62%	-0.43%	0.43%	1.30%	1.60%	0.59%	0.52%
benefit value	$\ell = \infty$	42.62%	38.50%	34.37%	31.24%	28.34%	25.48%	24.08%
	$\ell = 2$	41.88%	36.39%	34.85%	30.41%	28.05%	25.58%	22.31%
	difference	0.75%	2.11%	-0.48%	0.84%	0.29%	-0.10%	1.76%
contribution value	$\ell = \infty$	36.67%	33.28%	30.37%	30.77%	25.79%	23.44%	21.56%
	$\ell = 2$	38.40%	33.79%	31.53%	30.64%	27.91%	25.48%	24.00%
	difference	-1.73%	-0.51%	-1.16%	0.13%	-2.12%	-2.04%	-2.44%
Banzhaf* value	$\ell = \infty$	44.07%	39.81%	34.52%	31.04%	28.49%	25.85%	23.58%
	$\ell = 2$	42.96%	38.70%	34.52%	29.45%	27.90%	25.48%	23.01%
	difference	1.11%	1.11%	0.00%	1.60%	0.59%	0.38%	0.57%

Table 5.16: Relative ratios for **varying** country sizes under *d1*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	43.98%	38.25%	33.81%	30.90%	26.32%	25.66%	23.25%
	$\ell = 2$	39.36%	34.50%	29.35%	26.18%	23.18%	21.57%	19.49%
	difference	4.62%	3.75%	4.46%	4.72%	3.14%	4.09%	3.75%
Banzhaf value	$\ell = \infty$	44.06%	38.36%	33.94%	30.56%	27.16%	25.49%	23.25%
	$\ell = 2$	39.37%	34.46%	29.35%	26.26%	23.19%	21.57%	19.55%
	difference	4.68%	3.90%	4.59%	4.31%	3.97%	3.92%	3.70%
nucleolus	$\ell = \infty$	43.55%	37.44%	34.60%	30.29%	27.71%	25.34%	24.00%
	$\ell = 2$	39.83%	35.14%	29.60%	26.69%	23.49%	22.09%	20.07%
	difference	3.71%	2.29%	5.00%	3.59%	4.22%	3.26%	3.93%
tau value	$\ell = \infty$	43.20%	37.75%	34.42%	29.98%	27.80%	25.45%	23.76%
	$\ell = 2$	39.50%	34.80%	29.60%	26.25%	23.27%	21.74%	19.72%
	difference	3.70%	2.95%	4.82%	3.73%	4.53%	3.71%	4.04%
benefit value	$\ell = \infty$	43.12%	37.85%	34.67%	29.64%	27.81%	24.61%	23.38%
	$\ell = 2$	39.39%	34.84%	29.48%	26.04%	23.25%	21.76%	19.59%
	difference	3.73%	3.01%	5.19%	3.60%	4.57%	2.85%	3.78%
contribution value	$\ell = \infty$	44.02%	37.40%	34.34%	30.24%	26.69%	24.59%	23.38%
	$\ell = 2$	39.56%	34.43%	29.99%	26.29%	23.49%	22.17%	20.12%
	difference	4.46%	2.97%	4.35%	3.95%	3.21%	2.43%	3.27%
Banzhaf* value	$\ell = \infty$	44.06%	38.36%	33.94%	30.56%	27.16%	25.49%	23.25%
	$\ell = 2$	39.37%	34.46%	29.35%	26.26%	23.19%	21.57%	19.55%
	difference	4.68%	3.90%	4.59%	4.31%	3.97%	3.92%	3.70%

Table 5.17: Relative ratios for **equal** country sizes under *arbitrary*, for $\ell = \infty$, $\ell = 2$ and their difference.

n		4	5	6	7	8	9	10
Shapley value	$\ell = \infty$	43.61%	39.47%	35.60%	31.60%	29.39%	26.50%	23.95%
	$\ell = 2$	48.45%	44.14%	34.83%	33.25%	28.02%	27.37%	23.15%
	difference	-4.84%	-4.67%	0.77%	-1.65%	1.38%	-0.87%	0.79%
Banzhaf value	$\ell = \infty$	43.90%	39.04%	35.01%	31.64%	29.10%	26.15%	23.91%
	$\ell = 2$	47.85%	42.82%	34.13%	32.82%	27.54%	26.71%	22.78%
	difference	-3.96%	-3.78%	0.88%	-1.18%	1.56%	-0.56%	1.13%
nucleolus	$\ell = \infty$	41.05%	36.85%	33.31%	31.74%	28.16%	26.13%	23.71%
	$\ell = 2$	49.57%	47.31%	37.30%	35.07%	29.78%	29.64%	24.81%
	difference	-8.52%	-10.46%	-4.00%	-3.33%	-1.62%	-3.50%	-1.10%
tau value	$\ell = \infty$	43.03%	38.65%	34.12%	31.94%	29.69%	25.28%	24.72%
	$\ell = 2$	54.97%	45.25%	35.75%	33.56%	28.53%	28.18%	23.67%
	difference	-11.94%	-6.59%	-1.63%	-1.62%	1.17%	-2.90%	1.04%
benefit value	$\ell = \infty$	43.53%	39.76%	34.39%	32.04%	30.22%	25.84%	24.33%
	$\ell = 2$	48.63%	45.02%	35.56%	33.39%	28.49%	27.87%	23.42%
	difference	-5.10%	-5.25%	-1.16%	-1.35%	1.73%	-2.03%	0.91%
contribution value	$\ell = \infty$	37.03%	34.58%	32.17%	31.31%	26.83%	25.12%	23.61%
	$\ell = 2$	48.74%	47.81%	37.86%	36.47%	30.58%	31.08%	25.45%
	difference	-11.71%	-13.24%	-5.69%	-5.16%	-3.74%	-5.96%	-1.84%
Banzhaf* value	$\ell = \infty$	43.90%	39.04%	35.01%	31.64%	29.10%	26.15%	23.91%
	$\ell = 2$	47.85%	42.82%	34.13%	32.82%	27.54%	26.71%	22.78%
	difference	-3.96%	-3.78%	0.88%	-1.18%	1.56%	-0.56%	1.13%

Table 5.18: Relative ratios for **varying** country sizes under *arbitrary*, for $\ell = \infty$, $\ell = 2$ and their difference.

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	14.55	28.50	53.07	131.07	205.14	411.67	862.56
Nucleolus	11.98	25.55	49.71	96.95	181.50	390.55	994.93
Benefit value	9.94	11.55	12.91	16.20	19.55	19.88	21.52
Contribution value	8.98	9.94	11.12	13.63	19.44	23.28	25.62
Banzhaf value	12.60	24.99	49.60	97.31	187.44	420.78	840.45
Banzhaf* value	12.94	25.82	51.26	102.05	187.54	399.10	852.50
Tau value	21.87	40.05	79.66	126.31	252.16	405.81	943.97
total: arbitrary	11.63	20.35	35.35	65.49	123.81	257.33	530.78
total: d1	18.72	41.75	44.53	92.90	143.91	282.67	546.95
total: lexmin	33.93	80.73	95.63	178.48	315.86	471.83	1023.51
total: d1+c	20.21	40.31	50.13	101.81	162.01	352.32	596.70
total: lexmin+c	229.98	239.99	283.66	420.16	552.67	638.24	1103.81

Table 5.19: 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, which includes computing the initial allocations, and only excludes the time for data preparation and graph building.

CHAPTER 6

Simulations and Results for $\ell = 3$

In this chapter, we first investigate the known results for $\ell = 3$ in Section 6.1 and present our contributions in Section 6.2. Recall that it is NP-hard to compute optimal solutions under any constant $\ell \geq 3$ (see Theorem 1). For conducting simulations, we formulate ILPs for computing optimal solutions for $\ell = 3$ and strongly close optimal solutions in Section 6.3, which can also be used to compute weakly close optimal solutions.. And we conduct simulations for $\ell = 3$ with the same simulation setup as of $\ell = 2$ in Chapter 4 and $\ell = \infty$ in Chapter 5, and present our simulation results for $\ell = 3$ and comparison with $\ell = 2$ in Section 6.4. Finally, we present the conclusions of our simulation results in Section 6.5.

6.1 Known Results

In this section, we present the known simulation results for $\ell = 3$. As mentioned in Chapter 1, Klimentova et al. [10] and Biró et al. [11] both conducted simulations on $\ell = 3$. We summarize their simulation setup in Table 6.1.

Klimentova et al. [10] and Biró et al. [11] both identified optimal solutions under rationality constraints, that is, ensuring each country receives at least as many transplants in the international optimal solution as it could achieve in the national KEP. To find balanced solutions, rather than selecting weakly or strongly close optimal solutions, they chose optimal solution $\mathcal{C}$ from the set of individually rational optimal solutions that minimize the total deviations across all participating countries from target allocations, that is $\sum_{p \in N} d_p(\mathcal{C})$, which we call a *globally close optimal solution*. However, globally close optimal solutions are incomparable with weakly or strongly close optimal solutions, even when all optimal solutions are individual rational. For example, consider all optimal solutions $\mathcal{C}_1$ and $\mathcal{C}_2$ for an IKEP involving three countries, both of which satisfy individual rationality. Suppose that their deviations are $d(\mathcal{C}_1)$ and $d(\mathcal{C}_2)$, where $d(\mathcal{C}_1) = (0.6, 1.5, 3.3, 1.6, 2.6)$ and $d(\mathcal{C}_2) = (0.4, 2.5, 2.3, 2.6, 2.6)$. While $\mathcal{C}_2$ would be selected for strongly or weakly close optimal solutions since $d_1(\mathcal{C}_2) < d_1(\mathcal{C}_1)$, $\mathcal{C}_1$ would be selected as a globally close optimal solution because it yields the lower total deviations across all countries compared to $\mathcal{C}_2$. As for Klimentova et al. [10], they investigated the fairness of an IKEP by the introduction of the credit system. They used two solution concepts as initial allocations: the benefit value and *potential value*, calculated as $y_p = v(\{p\}) + (v(N) - \sum_{q \in N} v(\{q\}))\frac{UB_p - v(\{p\})}{\sum_{q \in N}(UB_q - v(\{q\}))}$ for each $p \in N$, where UB_p is an upper bound on the number of transplants that could be achieved by country p in international optimal solutions of an IKEP. Their results showed that the benefit value performs better than the potential value for balancing joint benefits for countries, and both of them lead to a similar number of transplants. As mentioned, the potential value is outperformed by the benefit value in [10] and we therefore replaced it by the contribution value in all our simulations. Biró et al. [11] considered the Shapley value and benefit value and showed that the Shapley value outperforms benefit value in finding balanced solutions and both of them result in a similar number of transplants.

6.2 Our Contributions

In our simulations, we perform simulations when $\ell = 3$ for 4 to 8 countries, both for equal and varying country sizes. We use 6 solution concepts and 1 variant of the

	Klimentova et al. [10]	Biró et al. [11]	Our simulations
number of countries	4	3	4-8
number of rounds	24	28	24
4 warm-up rounds	Y	Y	Y
number of instances	100	100	100
instance size	1650	1650	2000
number of solution concepts	2	2	6
maximum length of cycles	3	3	3
maximum length of chains	3	2	0
equal country sizes	N	Y	Y
varying country sizes	Y	Y	Y
weakly close optimal solutions	N	N	Y
strongly close optimal solutions	N	N	Y
globally close optimal solutions	Y	Y	N

Table 6.1: Comparison of simulation setup when $\ell = 3$ among Klimentova et al. [10], Biró et al. [11] and our simulations. Here, Y and N stand for Yes and No respectively.

Banzhaf value, derived from our alternative credit system (see Chapter 2).

Our contributions are fourfold. First of all, we conduct simulations for a larger number of countries, up to 8 countries, compared to 3 of Biró et al. [11] and 4 of Klimentova et al. [10]. Secondly, we extend the number of solution concepts from 2 (as of Klimentova et al. [10] and Biró et al. [11]) to 7, which provides a broader range of options for policy makers depending on their preferences for different axioms of fairness. Due to Theorem 1, computational challenges must be overcome for solution concepts when $\ell = 3$. As the number of countries n grows, it makes the problem even harder. Particularly for hard-to-compute solution concepts, like the Shapley value, we need to compute an exponential number of v-values (see Chapter 2) with respect to n. Furthermore, we use other different choices of optimal solutions, namely weakly close optimal solutions and strongly close optimal solutions. In the previous work [10, 11] for $\ell = 3$, Klimentova et al. [10] and Biró et al. [11] chose globally close optimal solutions which ensure individual rationality. In contrast, we follow a more refined approach to get balanced solutions by selecting strongly close optimal solutions, which we compare to arbitrarily chosen optimal solutions and arbitrarily chosen weakly close optimal solutions. Fourthly, we not only have a larger pool size of 2000 patient-donor pairs across 24 rounds compared to 1650 of Klimentova et al. [10] and Biró et al. [11], we also present the transplant distribution under length-2 and length-3 cycles across 24 rounds. However, unlike us, Klimentova et al. [10] and Biró et al. [11] included non-directed donors and considered chains in their simulations.

6.3 ILP Formulation

In this section, we show how to find an optimal solution of partitioned 3-permutation games that is strongly close to a given target allocation x by solving a sequence of Integer Linear Programs. Let (N, v) be a partitioned 3-permutation game defined on a directed graph $G = (V, A)$. Recall that for a maximum 3-cycle packing $\mathcal{C}$ of G, we let $s_p(\mathcal{C})$ denote the number of arcs (u, v) with $v \in V_p$ that belong to some directed cycle of $\mathcal{C}$. Recall also that $|x_p - s_p(\mathcal{C})|$ is the deviation of country $p \in N$ from its target x_p if $\mathcal{C}$ is chosen as optimal solution. Moreover, in the vector $d(\mathcal{C}) = (|x_{p_1} - s_{p_1}(\mathcal{C})|, \ldots, |x_{p_n} - s_{p_n}(\mathcal{C})|)$, the deviations $|x_p - s_p(\mathcal{C})|$ are ordered non-increasingly. Finally, we recall that $\mathcal{C}$ is strongly close to x if $d(\mathcal{C})$ is lexicographically minimal over all optimal solutions for (N, v).

In the kidney exchange literature the following ILP is called the *cycle-formulation*. Let $\mathcal{C}_3$ denote the set of all feasible cycles of length at most 3. For a cycle c, V_c is the set of vertices that belong to c and, for a country p, V_p is the set of vertices that belong to p. For each $c \in \mathcal{C}_3$, let $b_c \in \{0, 1\}$ be a binary cycle-variable.

$$m^* := \max_{c \in \mathcal{C}_3} \sum |V_c| b_c \quad \text{s.t.} \qquad \text{(cycle-formulation)}$$

$$\sum_{c \in \mathcal{C}_3 : i \in V_c} b_c \leq \quad 1 \qquad \forall i \in V \qquad (6.1)$$

$$b_c \in \quad \{0, 1\} \quad \forall c \in \mathcal{C}_3 \qquad (6.2)$$

This ILP contains $|\mathcal{C}_3|$ binary variables and $|V|$ constraints, and computes an optimal solution for $\ell = 3$. We take value 1 of b_c if cycle c is selected in the optimal solution, and 0 otherwise ($c \in \mathcal{C}_3$). Constraint 6.1 ensures that each vertex (patient-donor pair) appears in at most one of the selected cycles. Due to Theorem 1, unlike using the package of [87] to compute an optimal solution for partitioned permutation games, we use the above ILP to compute an optimal solution for partitioned 3-permutation games.

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 one 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, while $|N| \leq |V|$ holds by definition and typically $|N|$ is much smaller than $|\mathcal{C}_3|$. 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 number of countries with fixed deviation), the number of iterations are at most $|N|$ (as $t \leq |N|$). Hence, in addition to using one ILP for computing the value m^* of an optimal solution $\mathcal{C}$ (see ILP cycle-formulation) for partitioned 3-permutation games, we will solve no more than $2|N|$ ILPs to find a strongly close optimal solution, among which the largest has $\mathcal{O}(|\mathcal{C}_3| + |N|^2)$ binary variables and $\mathcal{O}(|V| + |N|)$ constraints.

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

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

$$\sum_{c \in \mathcal{C}_3 : i \in V_c} b_c \leq 1 \qquad \forall i \in V \qquad (6.1)$$

$$\sum_{c \in \mathcal{C}_3} |V_c| b_c = m^* \qquad (6.3)$$

$$\sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq d_1 \quad \forall p \in N \qquad (6.4)$$

$$x_p - \sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c \leq d_1 \quad \forall p \in N \qquad (6.5)$$

Constraints 6.1 and 6.3 guarantee that all solutions are in fact maximum 3-cycle-packings. These constraints will be part of the formulation throughout the entire

ILP-series. Constraints 6.4 and 6.5, together with the objective function guarantees that we minimize the largest country deviation. Note that for each country p, exactly one of $\sum_{c\in\mathcal{C}_3}|V_c\cap V_p|b_c - x_p$ and $x_p - \sum_{c\in\mathcal{C}_3}|V_c\cap V_p|b_c$ is positive and exactly one of them is negative, unless both of them are zero. However, as soon as we reach $d_t^* \leq 1/2$ we have found a strongly close optimal solution for $\ell = 3$. 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 $2|N|+|V|+1$ additional constraints. For every country $p \in N$ we have that $\left|\sum_{c\in\mathcal{C}_3}|V_c\cap V_p|b_c - x_p\right| \leq d_1^*$. However, there exists a smallest subset $N_1 \subseteq N$ (which may not necessarily be unique) such that

$$\left|\sum_{c\in\mathcal{C}_3}|V_c\cap V_p|b_c - x_p\right| = d_1^* \qquad \forall p \in N_1$$

$$\left|\sum_{c\in\mathcal{C}_3}|V_c\cap V_p|b_c - x_p\right| \leq d_2^* < d_1^* \quad \forall p \in N\setminus N_1$$

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 d_1^* deviation. 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 3-cycle packing is an integer. Hence, the number of fractional parts of possible country deviations is at most $2|N|$ and depends only on x. The fractional part of a country deviation is either $\mathrm{frac}(x_p)$ or $1-\mathrm{frac}(x_p)$. Therefore, to find the minimal positive difference in between the deviations of any two countries p and q, we have to compare the values $\mathrm{frac}(x_p)$ and $1-\mathrm{frac}(x_p)$, with $\mathrm{frac}(x_q)$ or $1-\mathrm{frac}(x_q)$ and take the minimum of those four possible differences. Let ε be a small positive constant that is weakly smaller than the minimum possible positive difference between any two countries. Recall that there is a case where the minimum possible positive difference does not exist when $\text{frac}(p) = 0.5$ for every $p \in N$. In this case, we let ε be a small constant satisfying $\varepsilon \in (0, 1]$.

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 $z_p^t \in \{0, 1\}$ binary variables, where $z_p^t = 1$ indicates that $p \in N_t$.

$$\min_{z^1, b} \sum_{p \in N} z_p^1 \quad \text{s.t.} \tag{ILP$_{N_1}$}$$

$$\sum_{c \in \mathcal{C}_3 : i \in V_c} b_c \leq \quad 1 \qquad \forall i \in V \tag{6.1}$$

$$\sum_{c \in \mathcal{C}_3} |V_c| b_c = \quad m^* \tag{6.3}$$

$$\sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad d_1^* - \varepsilon(1 - z_p^1) \quad \forall p \in N \tag{6.6}$$

$$x_p - \sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad d_1^* - \varepsilon(1 - z_p^1) \quad \forall p \in N \tag{6.7}$$

As discussed, for each country p, the left hand side of either constraint 6.6 or constraint 6.7 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 constraint 6.6 or constraint 6.7 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 optimal solution for $\ell = 3$ starts with exactly n_1^* many d_1^* values, followed by some $d_2^* < d_1^*$. (ILP$_{N_1}$) has $|\mathcal{C}_3| + |N|$ binary variables and $|V| + 2|N| + 1$ constraints. Now, to find d_2^*, we solve the following ILP:

$$\min_{d_2, b, z^1} d_2 \quad \text{s.t.} \tag{ILP$_{d_2}$}$$

$$\sum_{c \in \mathcal{C}_3 : i \in V_c} b_c \leq \quad 1 \qquad \forall i \in V \tag{6.1}$$

$$\sum_{c \in \mathcal{C}_3} |V_c| b_c = \quad m^* \tag{6.3}$$

$$\sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad d_1^* \qquad \forall p \in N \tag{6.8}$$

$$x_p - \sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad d_1^* \qquad \forall p \in N \tag{6.9}$$

$$\sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad d_2 + z_p^1 d_1^* \qquad \forall p \in N \tag{6.10}$$

$$x_p - \sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad d_2 + z_p^1 d_1^* \qquad \forall p \in N \tag{6.11}$$

$$\sum_{p \in N} z_p^1 = \quad n_1^* \tag{6.12}$$

(ILP$_{d_2}$) has $|\mathcal{C}_3| + |N|$ binary variables and one continuous variable (d_2) with $|V| + 4|N| + 2$ 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 $L \geq d_1^*$:

$$\min_{z^1, z^2, b} \sum_{p \in N} z_p^2 \quad \text{s.t.} \tag{ILP$_{N_2}$}$$

$$\sum_{c\in\mathcal{C}_3:i\in V_c} b_c \leq \quad 1 \qquad \forall i \in V \tag{6.1}$$

$$\sum_{c\in\mathcal{C}_3} |V_c| b_c = \quad m^* \tag{6.3}$$

$$\sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad d_1^* \qquad \forall p \in N \tag{6.8}$$

$$x_p - \sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad d_1^* \qquad \forall p \in N \tag{6.9}$$

$$\sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad d_2^* - \varepsilon(1 - z_p^2) + z_p^1 L \quad \forall p \in N \tag{6.13}$$

$$x_p - \sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad d_2^* - \varepsilon(1 - z_p^2) + z_p^1 L \quad \forall p \in N \tag{6.14}$$

$$z_p^1 + z_p^2 \leq \quad 1 \qquad \forall p \in N \tag{6.15}$$

$$\sum_{p\in N} z_p^1 = \quad n_1^* \tag{6.12}$$

Subsequently we follow a similar approach for all $t \geq 3$, until either $|N| = n_1^* + n_2^* + \cdots + n_t^*$ 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. $L = d_{t-1}^*$.

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

$$\sum_{c \in \mathcal{C}_3 : i \in V_c} b_c \leq \quad 1 \qquad \forall i \in V \tag{6.1}$$

$$\sum_{c \in \mathcal{C}_3} |V_c| b_c = \quad m^* \tag{6.3}$$

$$\sum_{i=1}^{t-1} z_p^i \leq \quad 1 \qquad \forall p \in N \tag{6.16}$$

$$\sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad d_t + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \tag{6.17}$$

$$x_p - \sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad d_t + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \tag{6.18}$$

$$\sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad \sum_{i=1}^{t-1} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t-1} z_p^i\right) L \qquad \forall p \in N \tag{6.19}$$

$$x_p - \sum_{c \in \mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad \sum_{i=1}^{t-1} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t-1} z_p^i\right) L \qquad \forall p \in N \tag{6.20}$$

$$\sum_{p \in N} z_p^i = \quad n_i^* \qquad \forall i \in \{1, \ldots, t-1\} \tag{6.21}$$

In the following formulation, L' is a large constant satisfying $L' > d_t^*$, e.g. $L' = d_{t-1}^*$.

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

$$\sum_{c\in\mathcal{C}_3:i\in V_c} b_c \leq \quad 1 \qquad \forall i \in V \qquad (6.1)$$

$$\sum_{c\in\mathcal{C}_3} |V_c| b_c = \quad m^* \qquad (6.3)$$

$$\sum_{i=1}^{t} z_p^i \leq \quad 1 \qquad \forall p \in N \qquad (6.22)$$

$$\sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad d_t^* - \varepsilon(1 - z_p^t) + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \qquad (6.23)$$

$$x_p - \sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad d_t^* - \varepsilon(1 - z_p^t) + \sum_{i=1}^{t-1} z_p^i d_i^* \qquad \forall p \in N \qquad (6.24)$$

$$\sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c - x_p \leq \quad \sum_{i=1}^{t} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t} z_p^i\right) L' \qquad \forall p \in N \qquad (6.25)$$

$$x_p - \sum_{c\in\mathcal{C}_3} |V_c \cap V_p| b_c \leq \quad \sum_{i=1}^{t} z_p^i d_i^* + \left(1 - \sum_{i=1}^{t} z_p^i\right) L' \qquad \forall p \in N \qquad (6.26)$$

$$\sum_{p\in N} z_p^i = \quad n_i^* \qquad \forall i \in \{1, \ldots, t-1\} \qquad (6.21)$$

From the above, we conclude that the following theorem holds.

Theorem 20. *For a partitioned* 3*-permutation game* (N, v) *defined on a graph* $G = (V, A)$ *and given the value* m^* *of an optimal solution, 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* $O(|\mathcal{C}_3| + |N|^2)$ *binary variables and* $O(|V| + |N|)$ *constraints.*

Note that if we just want to find a weakly close optimal solution we can stop after solving ILP_{d_1}.

6.4 Simulation Results

We now present our simulation results for $\ell = 3$, using the ILP formulations described in Section 6.3, and compare them with the results for $\ell = 2$ in Chapter 4. These two cases of $\ell = 2$ and $\ell = 3$ are implemented in countries such as France and UK respectively, where maximum cycle lengths of 2 and 3 are both realistic and relatively close, making them suitable for comparison. In contrast, we do not do an in-depth comparison between the cases $\ell = 3$ and $\ell = \infty$. The reason is that the main purpose for the simulations with $\ell = \infty$ was to quantify how limiting the maximum cycle length affects both the total number of kidney transplants and the balancedness of the credit system and to make a comparison with the other extreme case, where $\ell = 2$.

6.4.1 Total Relative Deviations

In Figure 6.1 we display our main results for *equal country sizes*, that is, when all countries are of the same size (see also Chapter 3). The ordering of five choices of optimal solutions (five scenarios; see Chapter 3) remains consistent across 6 solution concepts and the Banzhaf* value we use. In Figure 6.1, we compare different solution concepts under different scenarios for $\ell = 3$. Figure 6.1 also shows the effects of weakly and strongly close solutions and the credit system. Recall that solution concepts have different computational complexities. Hence, we believe comparing different solution concepts might be helpful for policy makers. As expected, using an arbitrary optimal solution for $\ell = 3$ in each round makes the kidney exchange scheme significantly more unbalanced, with average total relative deviations over 3.15% for all initial allocations y. 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.

The above observations are in line with the results under the setting where $\ell = 2$ in Chapter 4 and $\ell = \infty$ in Chapter 5. However, for $\ell = 2$, the effect of using arbitrary optimal solutions is much worse, while deviations are smaller than for $\ell = 3$ when weakly close or strongly close optimal solutions are chosen. From Figure 6.1 we see that the Banzhaf* value in the *lexmin+c* scenario provides the smallest deviations from the target allocations (as when $\ell = 2$ in Chapter 4). The Shapley value provides the lower total relative deviations across all scenarios compared to the benefit value, which is in line with Biró et al. [11]. However, all solution concepts are within 0.72% (for *lexmin+c*) when $n = 8$, and, as mentioned,

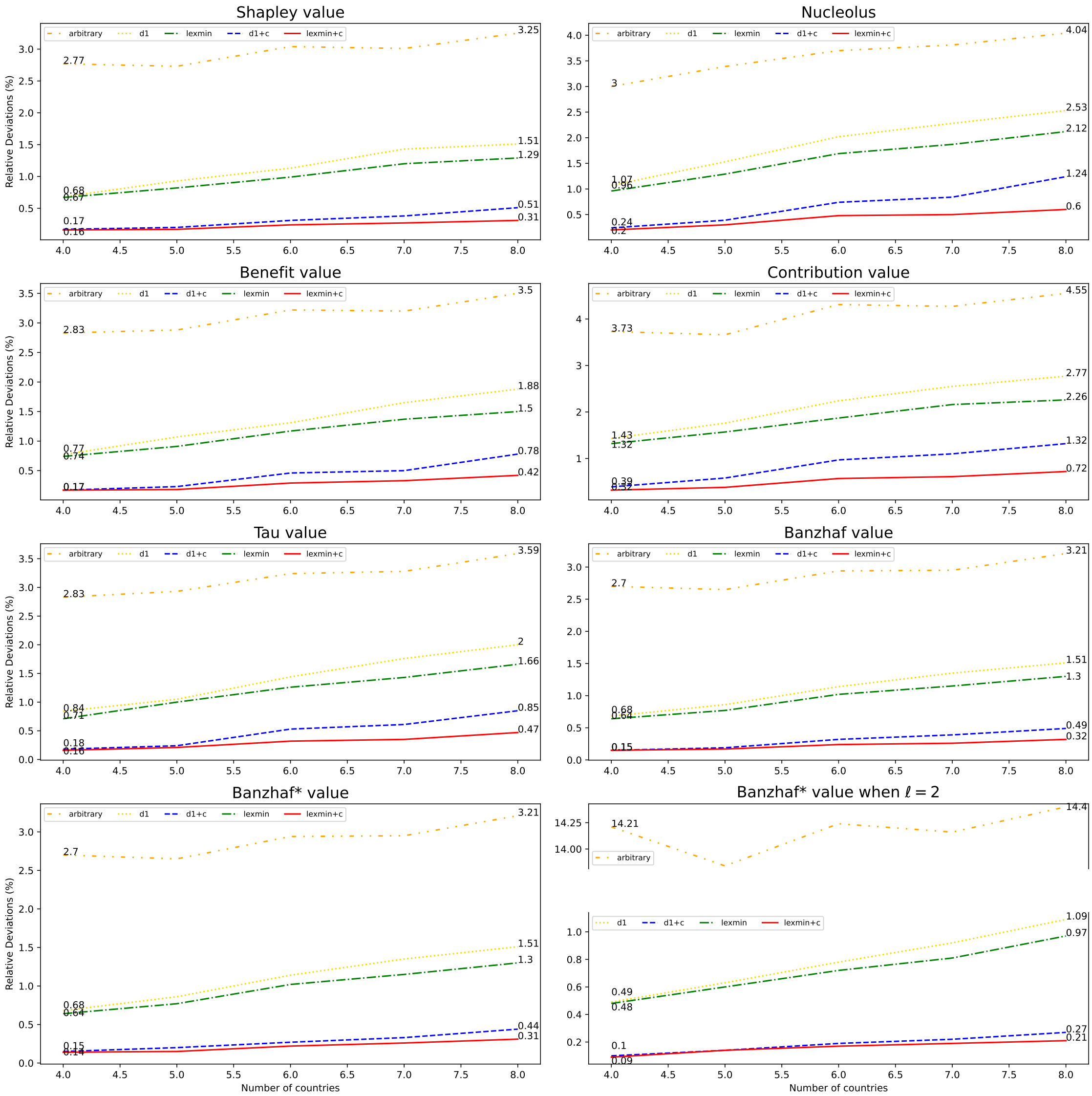

Figure 6.1: Average total relative deviations for each of the seven solution concepts under the five different scenarios for **equal** country sizes, where the number of countries n is ranging from 4 to 8. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which yields the lowest average relative deviations compared to the other six solution concepts 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.

which solution concept to select should be decided by the policy makers of the IKEP.

Figure 6.2 shows the same kind of results as Figure 6.1 but now for *varying country sizes*, that is, when we have small, medium and large countries, divided exactly as in Chapter 3 for $\ell = 2$ and $\ell = \infty$. Note that subject to minor fluctuations we can draw the same conclusions from Figure 6.2 for varying country sizes as we did from Figure 6.1 for equal country sizes.

Figures 6.1 and 6.2 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 Figures 6.1 and 6.2 in Figure 6.3 and all the *d1+c* plots from Figures 6.1 and 6.2 in Figure 6.4. Figures 6.3 and 6.4 both show a consistent ordering across 6 solution concepts and the Banzhaf* value under the strongly close optimal solution and weakly close optimal solution with the credit-compensation system. In nearly all cases for the two most important scenarios, *lexmin+c* and *d1+c*, the Banzhaf* value performs best, while the contribution value performs worst according to total relative deviations.

We note that Figures 6.3 and 6.4 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 $n \in \{4, 5, 6\}$.

6.4.2 Maximum Relative Deviations

We can draw exactly the same conclusions as above if we use the average maximum relative deviation instead of the average total relative deviation. We refer to Figures 6.5–6.8 for the analogs of Figures 6.1–6.4 if we use the average maximum relative deviation as our second evaluation measure.

6.4.3 Steady Relative Deviations

Recall that in the first round, one quarter of the total patient-donor pairs, namely 500 pairs, arrive. The remaining three quarters of the pairs arrive randomly over the next 23 rounds, following a uniform distribution. This substantial difference in the

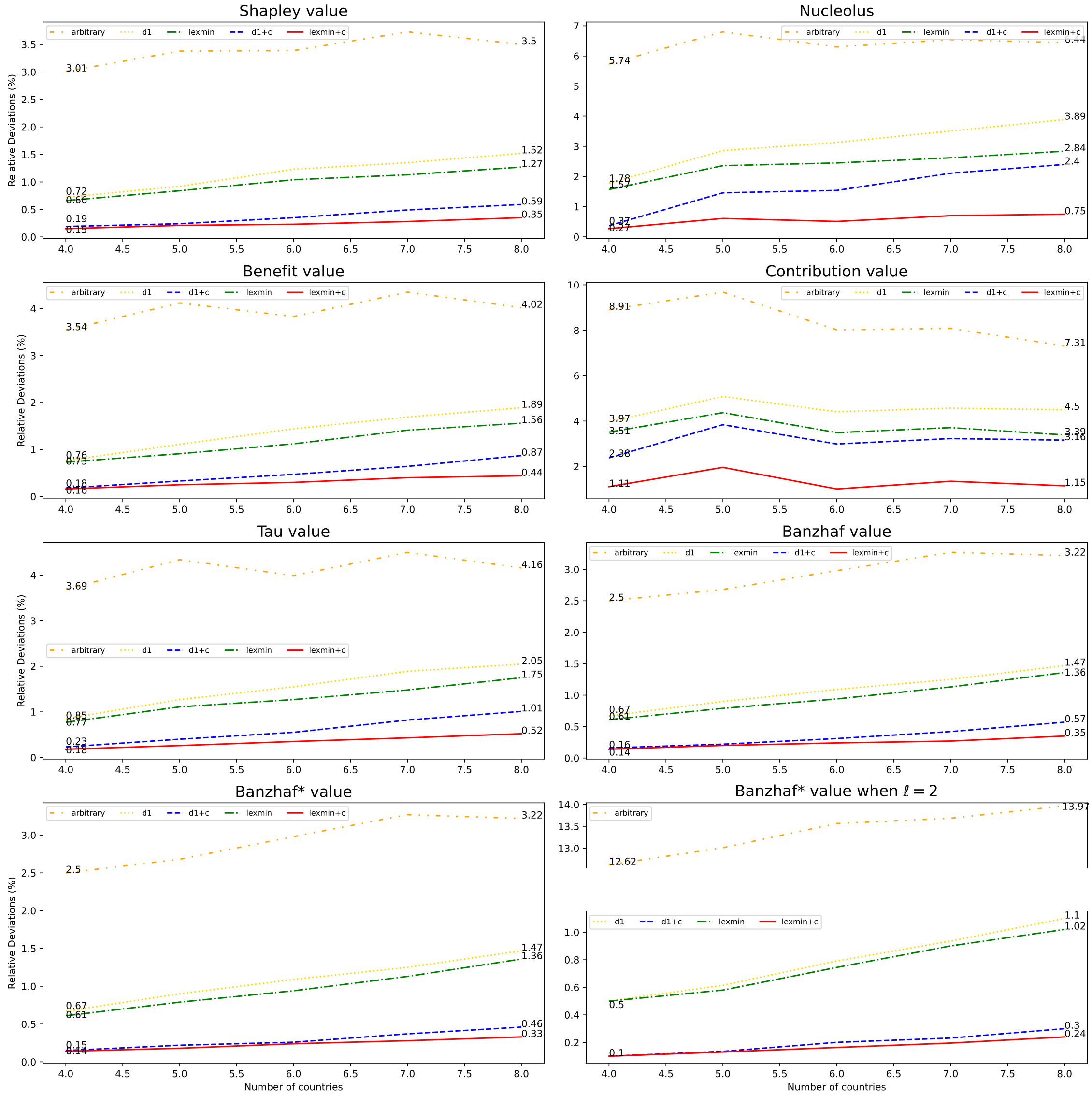

Figure 6.2: Average total relative deviations for each of the seven solution concepts under the five different scenarios for **varying** country sizes, where the number of countries n is ranging from 4 to 8. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which yields the lowest average relative deviations compared to the other six solution concepts for $\ell = 2$.

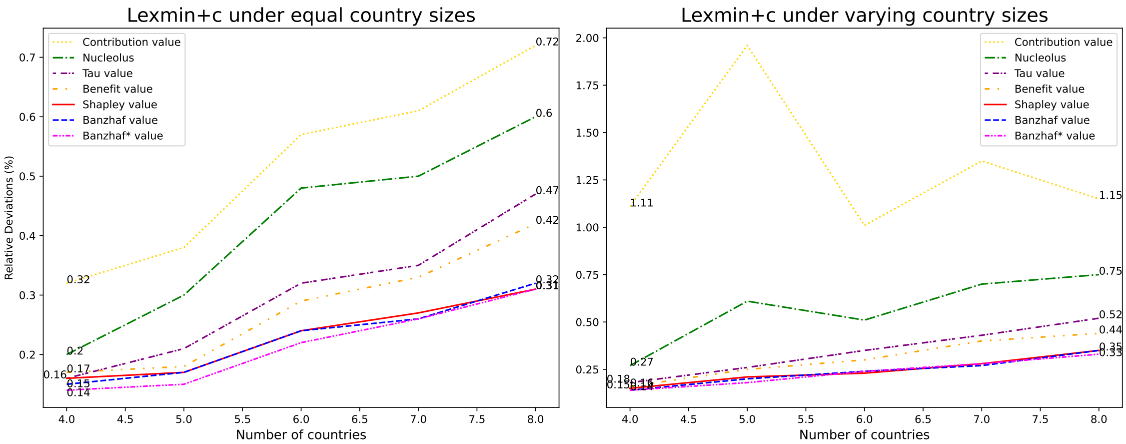

Figure 6.3: Average total relative deviations for all solution concepts in the *lexmin+c* scenario, where the number of countries n ranges from 4 to 8.

number of arrivals between the first round and the later rounds raises an important question: is our main measure (see Chapter 3), total relative deviation measured from the first to the 24th round, robust enough under such imbalance in the number of arrivals?

To address this, we propose the *steady relative deviation* as a measure of stability over the 20 rounds from the 5th to 24th, excluding the initial four. We treat the first four rounds as a warm-up phase, during which an arbitrary optimal solution is selected in each round. Following this warm-up, stability is assessed from round 5 through round 24. Let y^s be the *total initial allocation* of a single simulation instance for the last 20 rounds, that is, y^s is obtained by taking the sum of the 20 initial allocations from 5th to 24th of that instance. Let $\mathcal{C}^s$ be the union of the chosen maximum cycle packings in each of the corresponding 20 rounds. We defined the steady relative deviation as below:

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

Figure 6.9 presents the steady relative deviations, which is a analog of Figures 6.1 and 6.5. Unsurprisingly, when using the third measure, our results exhibit the same behavior as of Figures 6.1 and 6.5, leading us to draw and further reinforce the same conclusions.

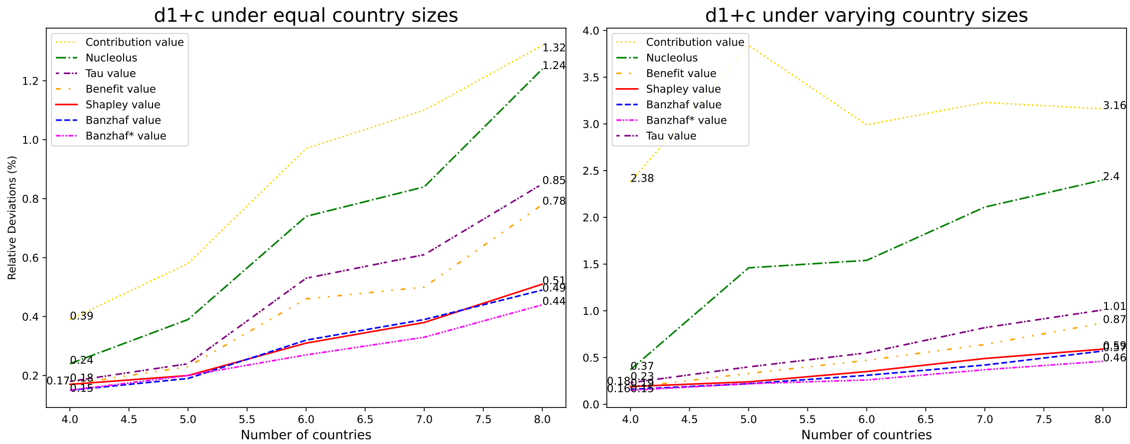

Figure 6.4: Average total relative deviations for all solution concepts in the *d1+c* scenario, where the number of countries n ranges from 4 to 8.

Incomplete Instances. In the simulations for $\ell = 2$ in Chapter 4, ILPs were not used, and all simulation instances were solved to optimality. However, our new simulations for $\ell = 3$ heavily relied on ILPs, including computing values v and finding optimal solutions, weakly close and strongly close optimal solutions. As previously mentioned 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 a few ILPs that could not be solved within the 1-hour 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-hour 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 and varying country sizes, 0 incomplete instances were observed for scenario *arbitrary*. Furthermore, $lexmin + c$ exhibited the highest total incomplete rate while $d1$ achieved the lowest incomplete rate in addition to *arbitrary*. Tables 6.2 and 6.3 summarize the distribution of incomplete simulation instances aggregated over different choices of solution concepts or scenarios. We refer to Tables 6.4 and 6.5 for a complete breakdown of the distribution in both these tables.

Given that the solutions found for incomplete simulation instances are all optimal solutions for $\ell = 3$ (which might not be weakly or strongly close), policy makers

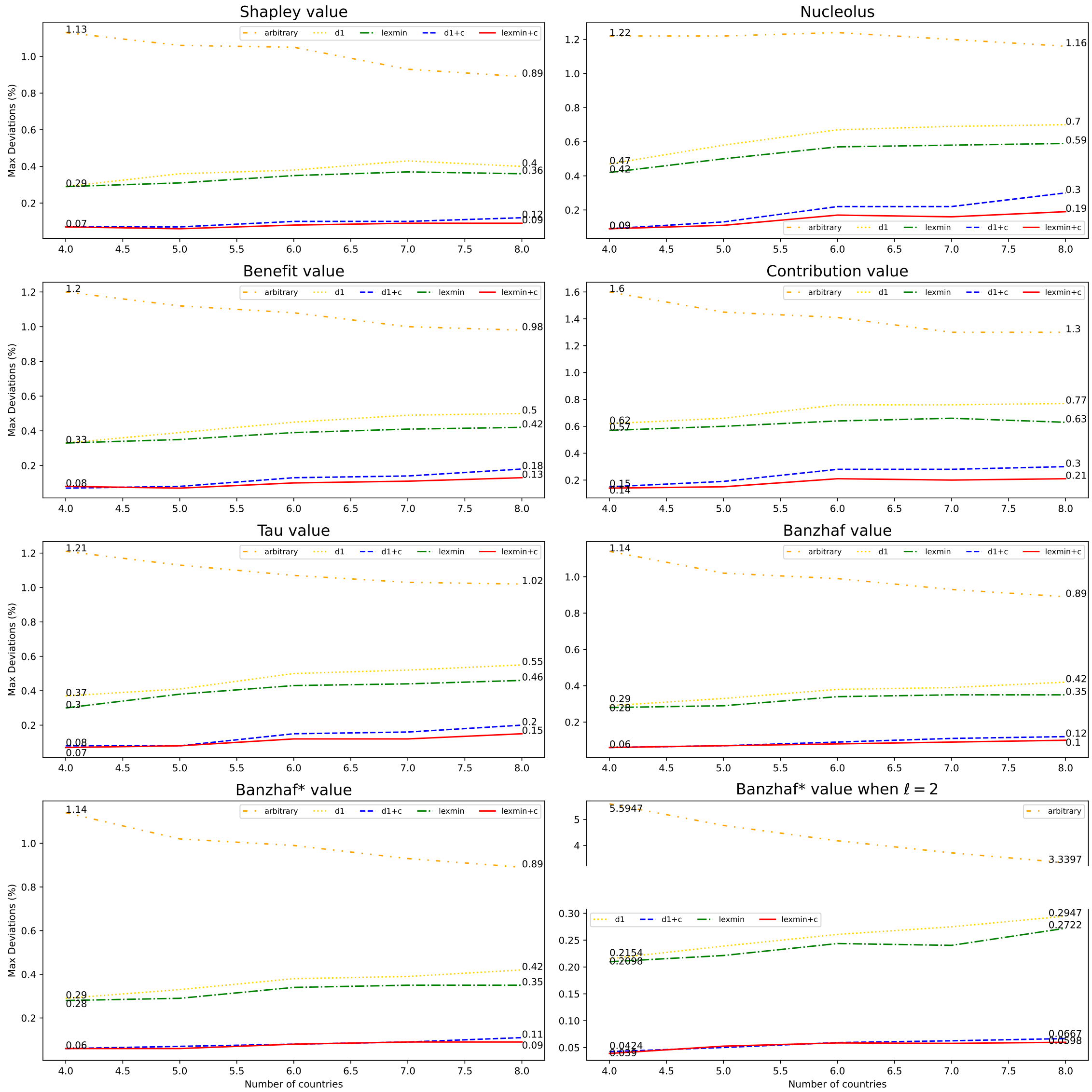

Figure 6.5: Average maximum relative deviations for each of the seven solution concepts under the five different scenarios for **equal** country sizes, where the number of countries n is ranging from 4 to 8. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which yields the lowest average maximum deviations compared to the other six solution concepts for $\ell = 2$.

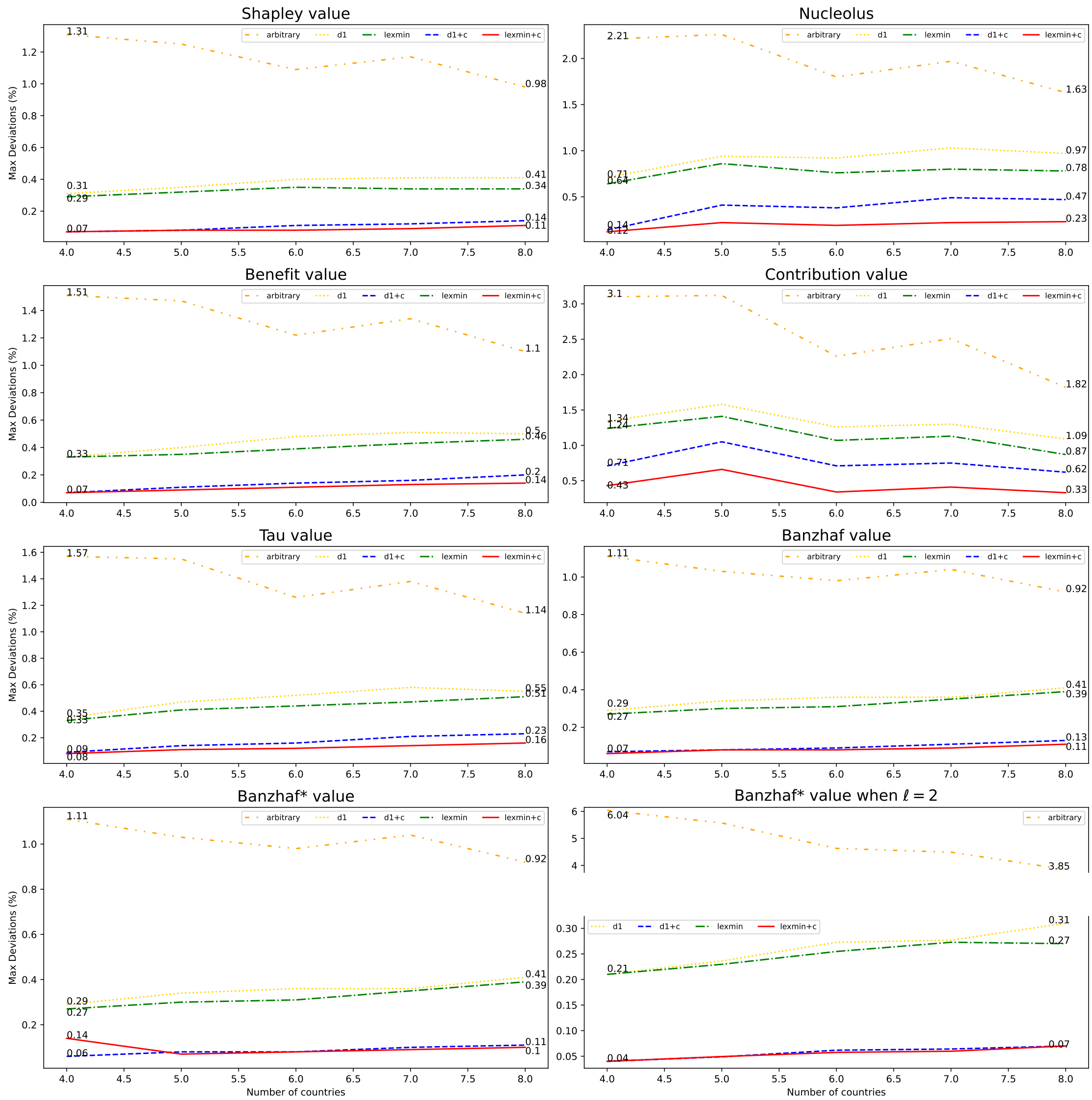

Figure 6.6: Average maximum relative deviations for each of the seven solution concepts under the five different scenarios for **varying** country sizes, where the number of countries n is ranging from 4 to 8. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which yields the lowest average maximum deviations compared to the other six solution concepts for $\ell = 2$.

could therefore still decide to use them. For this reason, we decided to include the incomplete simulation instances in Figures 6.1–6.12. We believe this was justified after doing some additional research. That is, we also constructed the same figures as Figures 6.1 and 6.2 but *without* the incomplete simulation instances; see Figures 6.13 and 6.14. It turned out that the largest percentage points difference in Figure 6.1 of the average relative deviations is only 0.022% (with an average of 0.0006%). 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.

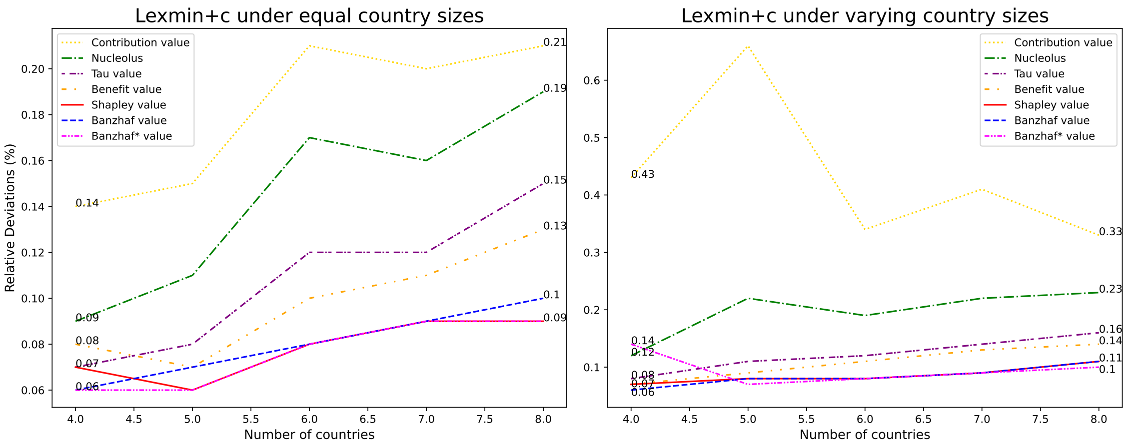

Figure 6.7: Average maximum relative deviations for all solution concepts in the *lexmin+c* scenario, where the number of countries n ranges from 4 to 8.

6.4.4 Number of Transplants

When $\ell = 3$ instead of $\ell = 2$, we expect more kidney transplants as exchange cycles can now include the size of both 2 and 3. In Tables 6.6–6.9 and Figure 6.10 we quantify this. From Tables 6.8–6.9 we see that when $\ell = 3$ instead of $\ell = 2$, over 100 instances, on average it achieves 2.70% more kidney transplants with $n = 8$ across all scenarios and initial allocations. This improvement holds for both equal and varying country sizes, with the number of countries ranging from 4 to 8. Notably, in the arbitrary scenario, the increase is slightly higher, namely 3.15%, whereas for the other four scenarios, the average increase is 2.59%. These results hold not only

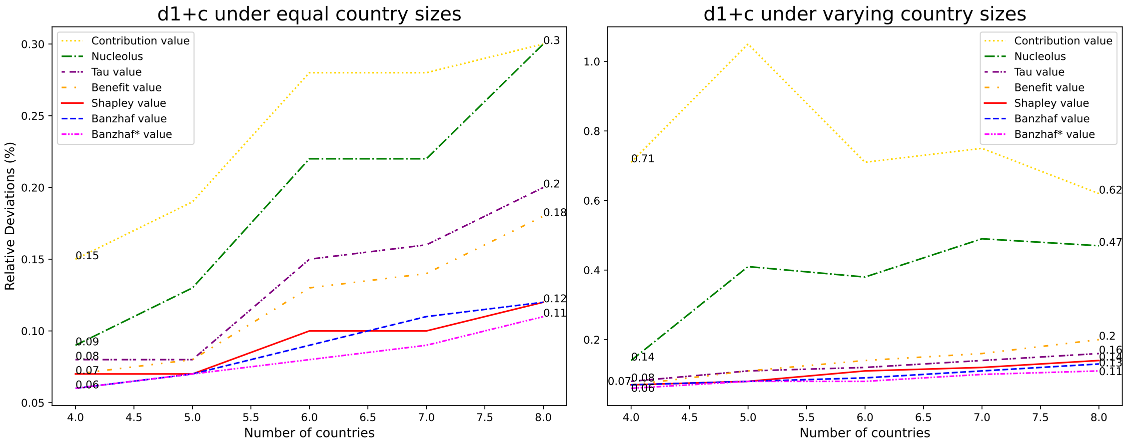

Figure 6.8: Average maximum relative deviations for all solution concepts in the *d1+c* scenario, where the number of countries n ranges from 4 to 8.

#incomplete / n	4	5	6	7	8	Total
Shapley value	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Nucleolus	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Benefit value	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Contribution value	0.40%	0.40%	0.00%	0.00%	0.20%	0.20%
Banzhaf value	0.00%	0.40%	0.80%	0.80%	0.20%	0.44%
Tau value	0.00%	0.00%	0.00%	0.80%	0.00%	0.16%
Banzhaf* value	0.00%	0.00%	1.00%	1.00%	0.00%	0.40%
total: arbitrary	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
total: d1	0.00%	0.00%	0.00%	0.17%	0.00%	0.03%
total: lexmin	0.17%	0.33%	0.33%	0.50%	0.00%	0.27%
total: d1+c	0.00%	0.14%	0.00%	0.14%	0.00%	0.06%
total: lexmin+c	0.14%	0.14%	0.57%	0.71%	0.29%	0.37%

Table 6.2: Average number of incomplete simulation instances for **equal** country sizes.

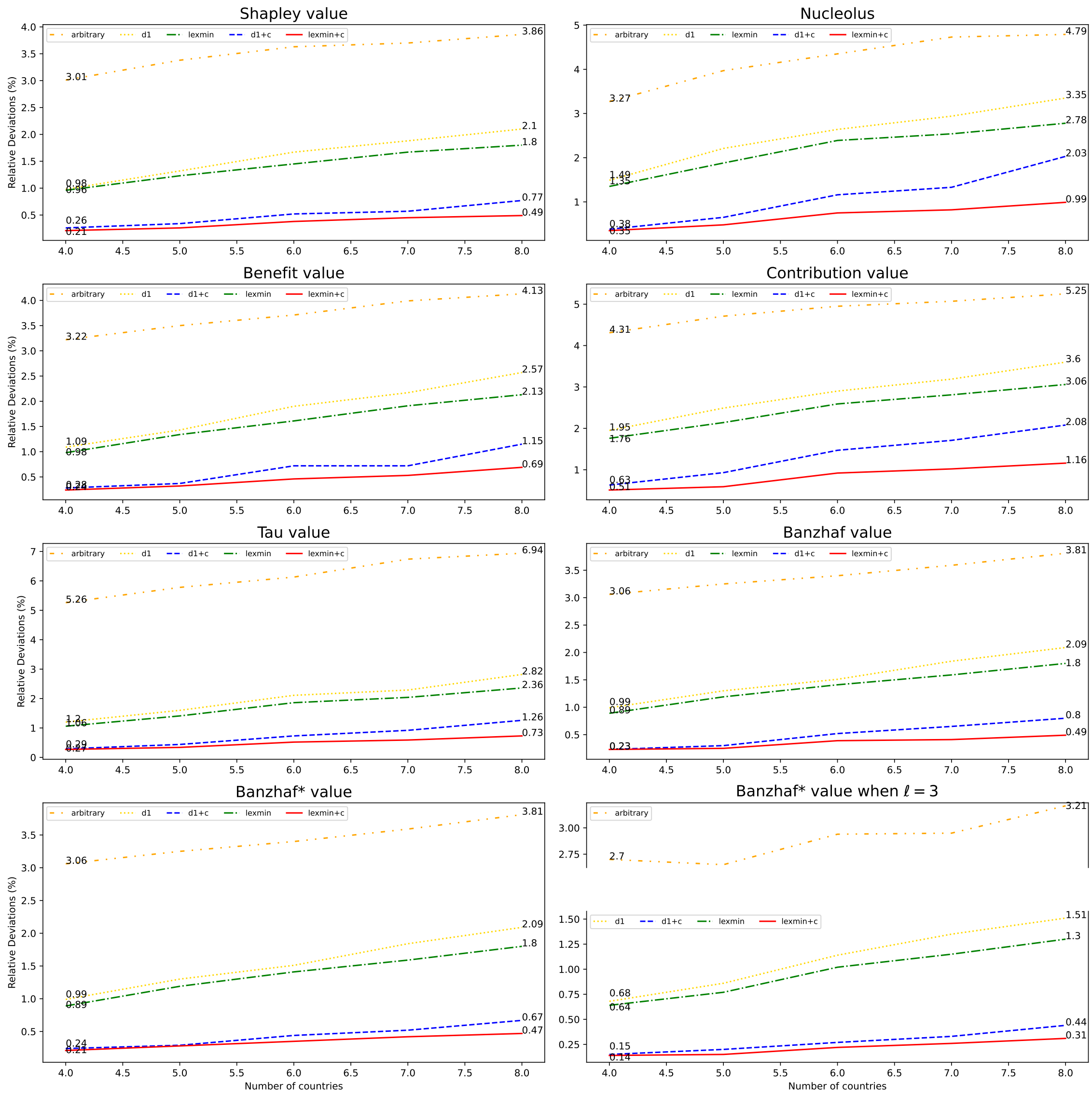

Figure 6.9: Average steady relative deviations for each of the seven solution concepts under the five different scenarios for **equal** country sizes, where the number of countries n is ranging from 4 to 8. For comparison, the bottom right figure displays the total relative deviations for $\ell = 3$ from Figure 6.1, specifically for the Banzhaf* value, which yields the lowest average relative deviations compared to the other six solution concepts for $\ell = 3$.

#incomplete / n	4	5	6	7	8	Total
Shapley value	0.00%	0.00%	0.00%	0.40%	0.00%	0.08%
Nucleolus	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Benefit value	0.00%	0.00%	0.00%	0.40%	0.40%	0.16%
Contribution value	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Banzhaf value	0.00%	0.80%	0.40%	0.40%	0.80%	0.48%
Tau value	0.00%	0.40%	0.40%	0.00%	0.00%	0.16%
Banzhaf* value	0.00%	1.00%	0.50%	0.50%	1.50%	0.70%
total: arbitrary	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
total: d1	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
total: lexmin	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
total: d1+c	0.00%	0.50%	0.33%	0.50%	0.50%	0.37%
total: lexmin+c	0.00%	0.71%	0.43%	0.57%	0.86%	0.51%

Table 6.3: Average number of incomplete simulation instances for **varying** country sizes.

for $n = 8$, but also consistently across for all integer values of n ranging from 4 to 8. Among the four strongly and weakly close scenarios, the total number of kidney transplants is nearly identical for $\ell = 3$ where n ranging from 4 to 8, indicating that the choice among these four has little difference on the number of transplants and the result is in line with Klimentova et al. [10] and Biró et al. [11]. However, when considering all five scenarios over 100 instances, the *arbitrary* scenario, on average, gains 4 extra kidney transplants compared to the other four scenarios both for equal and varying country sizes. In contrast, the total number of transplants remains nearly the same across five scenarios when $\ell = 2$ both for equal and varying country sizes. Recall that for $\ell = 3$, we select an arbitrary strongly or weakly close optimal solution without applying any additional criteria even when multiple strongly or weakly close optimal solutions exist. The 4 additional transplants may result from some extremely hard-to-match patient-donor pairs that happen to be selected in the length-3 cycle in the arbitrary scenario but not in the other four scenarios.

Figure 6.10 illustrates the distribution of the number of kidney transplants for strongly optimal solutions when $n = 8$ and Banzhaf* value is selected. It shows the breakdown of transplants into length-2 and length-3 cycles under $\ell = 3$. Specifically, over 100 instances, length-3 cycles contribute an average of 64.14 kidneys out of a total of 345.08 in the first round. In subsequent rounds, they account for an average

#incomplete / n	4	5	6	7	8	Total
			Shapley value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	0	0	0	0	0
lexmin+c	0	0	0	0	0	0
			Nucleolus			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	0	0	0	0	0
lexmin+c	0	0	0	0	0	0
			Benefit value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	0	0	0	0	0
lexmin+c	0	0	0	0	0	0
			Contribution value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	1	1	0	0	0	2
lexmin+c	1	1	0	0	1	3
			Tau value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	1	0	1
d1+c	0	0	0	1	0	1
lexmin	0	0	0	1	0	1
lexmin+c	0	0	0	1	0	1
			Banzhaf value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	1	0	0	0	1
lexmin	0	1	2	2	0	5
lexmin+c	0	0	2	2	1	5
			Banzhaf* value			
d1+c	0	0	0	0	0	0
lexmin+c	0	0	2	2	0	4

Table 6.4: Complete breakdown of the number of incomplete simulation instances (out of 100) for the seven different solution concepts and the five different scenarios for **equal** country sizes.

#**incomplete**/n	4	5	6	7	8	Total
			Shapley value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	0	0	1	0	1
lexmin+c	0	0	0	1	0	1
			Nucleolus			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	0	0	0	0	0
lexmin+c	0	0	0	0	0	0
			Benefit value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	0	0	1	1	2
lexmin+c	0	0	0	1	1	2
			Contribution value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	0	0	0	0	0
lexmin+c	0	0	0	0	0	0
			Tau value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	1	1	0	0	2
lexmin+c	0	1	1	0	0	2
			Banzhaf value			
arbitrary	0	0	0	0	0	0
d1	0	0	0	0	0	0
d1+c	0	0	0	0	0	0
lexmin	0	2	1	1	2	6
lexmin+c	0	2	1	1	2	6
			Banzhaf* value			
d1+c	0	0	0	0	0	0
lexmin+c	0	2	1	1	3	7

Table 6.5: Complete breakdown of the number of incomplete simulation instances (out of 100) for the seven different solution concepts and the five different scenarios for **varying** country sizes.

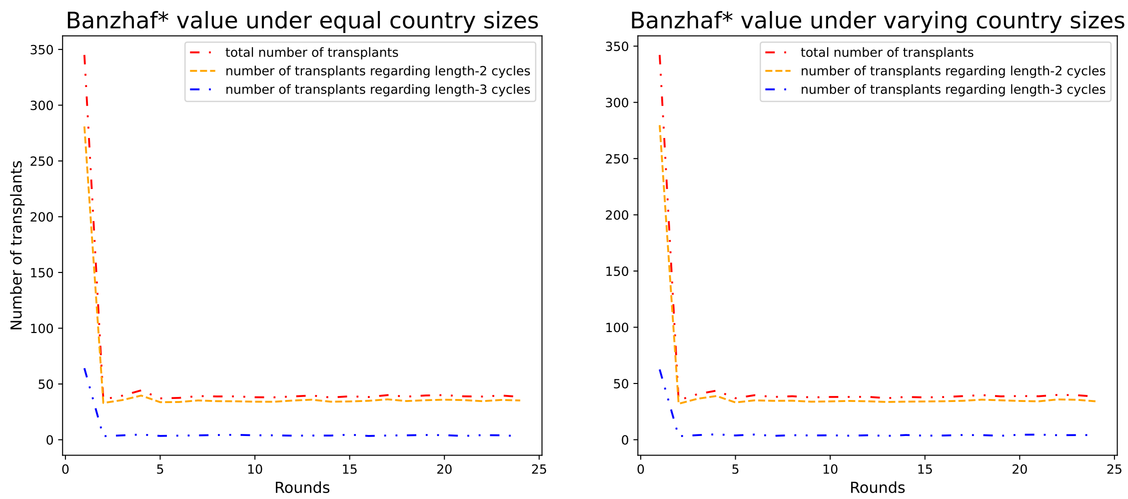

Figure 6.10: Transplant distribution for the Banzhaf* value in the *lexmin+c* scenario when $n = 8$, where the x-axis represents the exchange round, and the y-axis shows the number of kidney transplants involved in that round. The red dotted line represents the total number of transplants occurring in that round when $\ell = 3$, while the orange dotted line represents the number of transplants occurring when only length-2 cycles are allowed and the blue dotted line represents the number of transplants occurring when only length-3 cycles are allowed.

of 3.89 kidneys out of 38.93.

6.4.5 Credit Accumulation

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 Chapter 1 for a theoretical example of this behaviour when $\ell = 2$).

In Figures 6.11 and 6.12 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, as in this case the target allocations are equivalent to the initial allocations. Therefore, credits $\sum_{p \in N^r} |c_p^r|$ for round r are computed by summing the absolute difference between the initial allocations and actual allocations for each country $p \in N^r$ when the target allocations and initial allocations are equivalent, that is, $\sum_{p \in N^r} |c_p^r| = \sum_{p \in N^r} |y_p^r - s_p^r(\mathcal{C})|$. We note that over a period of 24 rounds, credits accumulate more and more (but at different rates) under all three scenarios *arbitary*, *d1* and *lexmin*, even though

Solution concepts/n		4	5	6	7	8
benefit value	*arbitrary*	1259.76	1257.11	1253.47	1246.72	1245.74
	d1	1256.13	1253.81	1249.02	1243.74	1239.75
	d1+c	1255.55	1252.94	1248.02	1243.87	1240.19
	lexmin	1255.76	1253.43	1248.24	1243.60	1240.22
	lexmin+c	1256.43	1253.29	1248.80	1244.08	1240.23
contribution value	*arbitrary*	1259.76	1257.11	1253.47	1246.72	1245.74
	d1	1256.62	1252.74	1249.01	1243.35	1240.18
	d1+c	1256.16	1252.91	1248.98	1244.35	1241.00
	lexmin	1255.53	1253.17	1249.20	1243.33	1241.13
	lexmin+c	1255.89	1253.56	1248.55	1244.14	1240.57
Nucleolus	*arbitrary*	1259.76	1257.11	1253.47	1246.72	1245.74
	d1	1256.05	1253.33	1248.88	1243.88	1240.75
	d1+c	1256.93	1252.55	1247.58	1243.77	1240.25
	lexmin	1256.99	1253.53	1248.21	1242.97	1241.15
	lexmin+c	1256.00	1253.14	1248.62	1243.91	1239.73
Shapley value	*arbitrary*	1259.76	1257.11	1253.47	1246.72	1245.74
	d1	1257.10	1253.13	1248.90	1243.45	1239.74
	d1+c	1256.72	1253.17	1248.83	1243.03	1240.47
	lexmin	1256.54	1253.10	1248.11	1243.87	1240.42
	lexmin+c	1256.58	1253.24	1248.42	1243.33	1240.07
Banzhaf value	*arbitrary*	1259.76	1257.11	1253.47	1246.72	1245.74
	d1	1255.84	1253.83	1247.96	1244.26	1240.12
	d1+c	1256.46	1253.59	1248.54	1243.71	1240.46
	lexmin	1256.32	1253.90	1248.11	1243.18	1240.23
	lexmin+c	1256.17	1252.90	1247.93	1244.05	1240.11
Banzhaf* value	*arbitrary*	1259.76	1257.11	1253.47	1246.72	1245.74
	d1	1255.84	1253.83	1247.96	1244.26	1240.12
	d1+c	1256.36	1253.43	1247.99	1242.77	1240.63
	lexmin	1256.32	1253.90	1248.11	1243.18	1240.23
	lexmin+c	1256.60	1253.58	1247.95	1243.82	1240.58
Tau value	*arbitrary*	1259.76	1257.11	1253.47	1246.72	1245.74
	d1	1255.94	1252.49	1248.53	1243.68	1240.46
	d1+c	1256.94	1252.79	1248.52	1243.09	1240.38
	lexmin	1257.02	1252.64	1247.86	1243.92	1240.63
	lexmin+c	1255.99	1252.85	1248.50	1244.12	1240.40

Table 6.6: Average number of kidney transplants for 6 solution concepts along with the Banzhaf* value across 5 scenarios where $\ell = 3$ with the number of **equal-sized** countries ranging from 4 to 8. See Chapter 4 for full results in the case where $\ell = 2$.

Solution concepts/n		4	5	6	7	8
benefit value	*arbitrary*	1252.17	1259.33	1240.05	1227.32	1234.55
	d1	1248.66	1254.83	1236.74	1223.64	1229.38
	d1+c	1248.58	1254.43	1236.56	1223.29	1229.53
	lexmin	1247.56	1255.12	1236.14	1224.35	1229.27
	lexmin+c	1249.05	1254.89	1236.56	1224.03	1230.43
contribution value	*arbitrary*	1252.17	1259.33	1240.05	1227.32	1234.55
	d1	1247.65	1254.66	1236.67	1223.55	1229.90
	d1+c	1248.20	1253.67	1234.90	1224.23	1229.94
	lexmin	1248.45	1254.40	1235.97	1223.68	1230.10
	lexmin+c	1247.63	1253.79	1234.99	1223.97	1231.13
Nucleolus	*arbitrary*	1252.17	1259.33	1240.05	1227.32	1234.55
	d1	1248.03	1254.10	1237.04	1223.30	1230.47
	d1+c	1248.09	1254.61	1235.92	1223.86	1229.96
	lexmin	1248.02	1253.27	1235.19	1223.13	1230.10
	lexmin+c	1248.27	1253.87	1235.03	1223.22	1230.07
Shapley value	*arbitrary*	1252.17	1259.33	1240.05	1227.32	1234.55
	d1	1247.44	1254.40	1237.07	1222.86	1230.39
	d1+c	1247.97	1254.12	1236.44	1222.78	1230.12
	lexmin	1247.19	1254.26	1236.19	1223.69	1230.58
	lexmin+c	1248.46	1254.43	1235.88	1223.44	1230.30
Banzhaf value	*arbitrary*	1252.17	1259.33	1240.05	1227.32	1234.55
	d1	1247.82	1255.11	1235.47	1223.65	1230.50
	d1+c	1248.56	1254.47	1235.96	1223.85	1230.33
	lexmin	1246.34	1254.08	1236.05	1221.81	1230.69
	lexmin+c	1247.54	1255.24	1235.70	1223.19	1230.29
Banzhaf* value	*arbitrary*	1252.17	1259.33	1240.05	1227.32	1234.55
	d1	1247.82	1255.11	1235.47	1223.65	1230.50
	d1+c	1248.45	1254.77	1236.10	1223.73	1230.36
	lexmin	1246.34	1254.08	1236.05	1221.81	1230.69
	lexmin+c	1248.64	1253.55	1235.70	1223.50	1229.25
Tau value	*arbitrary*	1252.17	1259.33	1240.05	1227.32	1234.55
	d1	1248.20	1255.01	1237.08	1223.74	1230.05
	d1+c	1248.42	1255.19	1236.38	1223.30	1230.33
	lexmin	1248.49	1255.13	1236.61	1223.16	1229.44
	lexmin+c	1248.34	1254.68	1235.67	1223.48	1230.34

Table 6.7: Average number of kidney transplants for 6 solution concepts along with the Banzhaf* value across 5 scenarios where $\ell = 3$ with the number of **varying-sized** countries ranging from 4 to 8.

Solution concepts/scnearios		*arbitrary*	*d1*	*d1+c*	*lexmin*	*lexmin+c*
	benefit value	1245.74	1239.75	1240.19	1240.22	1240.23
	contribution value	1245.74	1240.18	1241.00	1241.13	1240.57
	Nucleolus	1245.74	1240.75	1240.25	1241.15	1239.73
$\ell = 3$	Shapley value	1245.74	1239.74	1240.47	1240.42	1240.07
	Banzhaf value	1245.74	1240.12	1240.46	1240.23	1240.11
	Banzhaf* value	1245.74	1240.12	1240.63	1240.23	1240.58
	Tau value	1245.74	1240.46	1240.38	1240.63	1240.40
	benefit value	1207.7	1209.24	1208.8	1209.64	1209.92
	contribution value	1207.7	1210.96	1208.36	1208.96	1209.84
	Nucleolus	1207.7	1208.92	1209.24	1208.8	1208.08
$\ell = 2$	Shapley value	1207.7	1207.98	1209.74	1208.64	1208.42
	Banzhaf value	1207.7	1208.7	1208.72	1208.9	1208.5
	Banzhaf* value	1207.7	1208.7	1208.94	1208.9	1208.8
	Tau value	1207.7	1209.74	1208.76	1208.5	1209.46

Table 6.8: Average number of kidney transplants under 6 solution concepts along with the Banzhaf* value across 5 scenarios with 8 **equal-sized** countries.

Solution concepts/scnearios		*arbitrary*	*d1*	*d1+c*	*lexmin*	*lexmin+c*
	benefit value	1234.55	1229.38	1229.53	1229.27	1230.43
	contribution value	1234.55	1229.90	1229.94	1230.10	1231.13
	Nucleolus	1234.55	1230.47	1229.96	1230.10	1230.07
$\ell = 3$	Shapley value	1234.55	1230.39	1230.12	1230.58	1230.30
	Banzhaf value	1234.55	1230.50	1230.33	1230.69	1230.29
	Banzhaf* value	1234.55	1230.50	1230.36	1230.69	1229.25
	Tau value	1234.55	1230.05	1230.33	1229.44	1230.34
	benefit value	1193.58	1197.92	1197.78	1197.98	1197.72
	contribution value	1194.12	1197.14	1197.78	1196.48	1197.52
	Nucleolus	1194.12	1196.96	1197.32	1197.18	1197.26
$\ell = 2$	Shapley value	1194.12	1197.12	1196.86	1197.26	1197.8
	Banzhaf value	1194.12	1197.68	1197.3	1197.78	1196.68
	Banzhaf* value	1194.12	1197.68	1197.02	1197.78	1197.28
	Tau value	1194.12	1196.38	1196.26	1196.36	1196.36

Table 6.9: Average number of kidney transplants under 6 solution concepts along with the Banzhaf* value across 5 scenarios with 8 **varying-sized** countries.

especially under *d1* and *lexmin* we do find solutions that are relatively close to the target allocations (as we saw from Figures 6.1 and 6.2). 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 $\ell = 2$ in Chapter 4. Again, the main difference between $\ell = 3$ and $\ell = 2$ is that the *arbitrary* scenario accumulates significantly higher credits under $\ell = 2$ than under $\ell = 3$, and this gap widens as the number of countries increases.

6.4.6 Computation Time

Table 6.10 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 (the benefit value and contribution value) is less expensive, especially compared to the other, more sophisticated but hard-to-compute solution concepts.

In Table 6.10 we also see that the computation time for the *arbitrary* scenario, which does not require strongly or weakly close optimal solutions, is slightly less time-consuming than for the other four scenarios. This is in line with Theorem 1, which states that for $\ell = 3$, it is NP-hard to compute optimal solutions. As a result, we need to rely on using ILPs for computing v-values for initial allocations regardless of the chosen optimal solution. Additionally, for the strongly and weakly close optimal solutions, we solve a series of ILPs to to find a suitable optimal solution, whereas, for the *arbitrary* scenario, we stop once an optimal solution is found. From Table 6.10, it can be clearly seen that the computation of the initial allocations requires the majority of the work. The computation workload increases with the number of countries, particularly in the *lexmin* and *lexmin+c* scenarios, which require solving longer sequences of ILPs. In contrast, the *d1* and *d1+c* scenarios are less demanding, as they only require solving a single ILP, namely, ILP_{d_1}. Strongly guided by Theorem 1 and given CPU time increases significantly, nearly doubling as n grows, we are only able to perform simulations for up to 8 countries for $\ell = 3$, particularly due to the computational challenges of the hard-to-compute solution

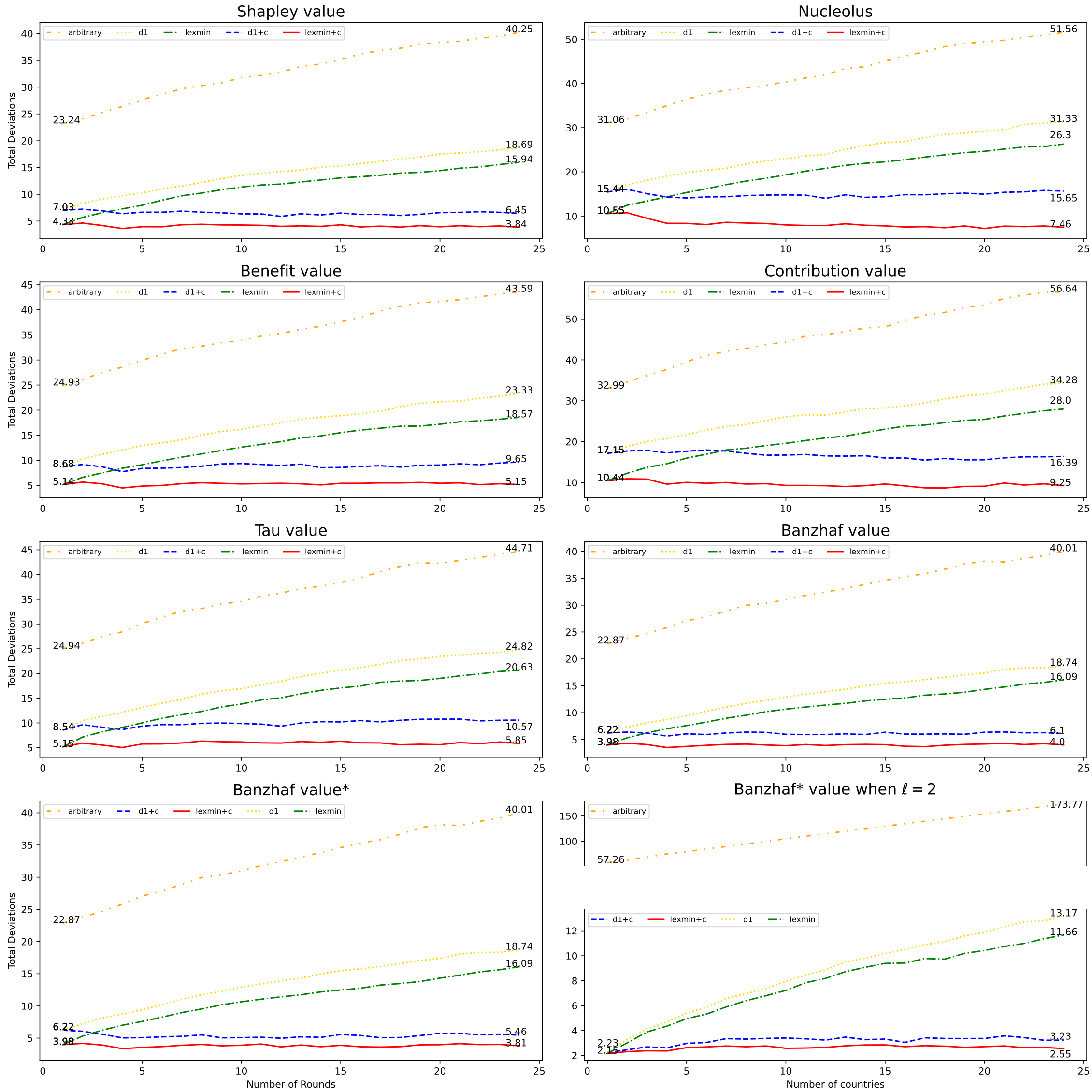

Figure 6.11: Average credits for each of the seven solution concepts under the five different scenarios for **equal** country sizes, where the number of countries is $n = 8$ and the period ranges from 1 to 24. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which yields the lowest average credits compared to the other six solution concepts for $\ell = 2$.

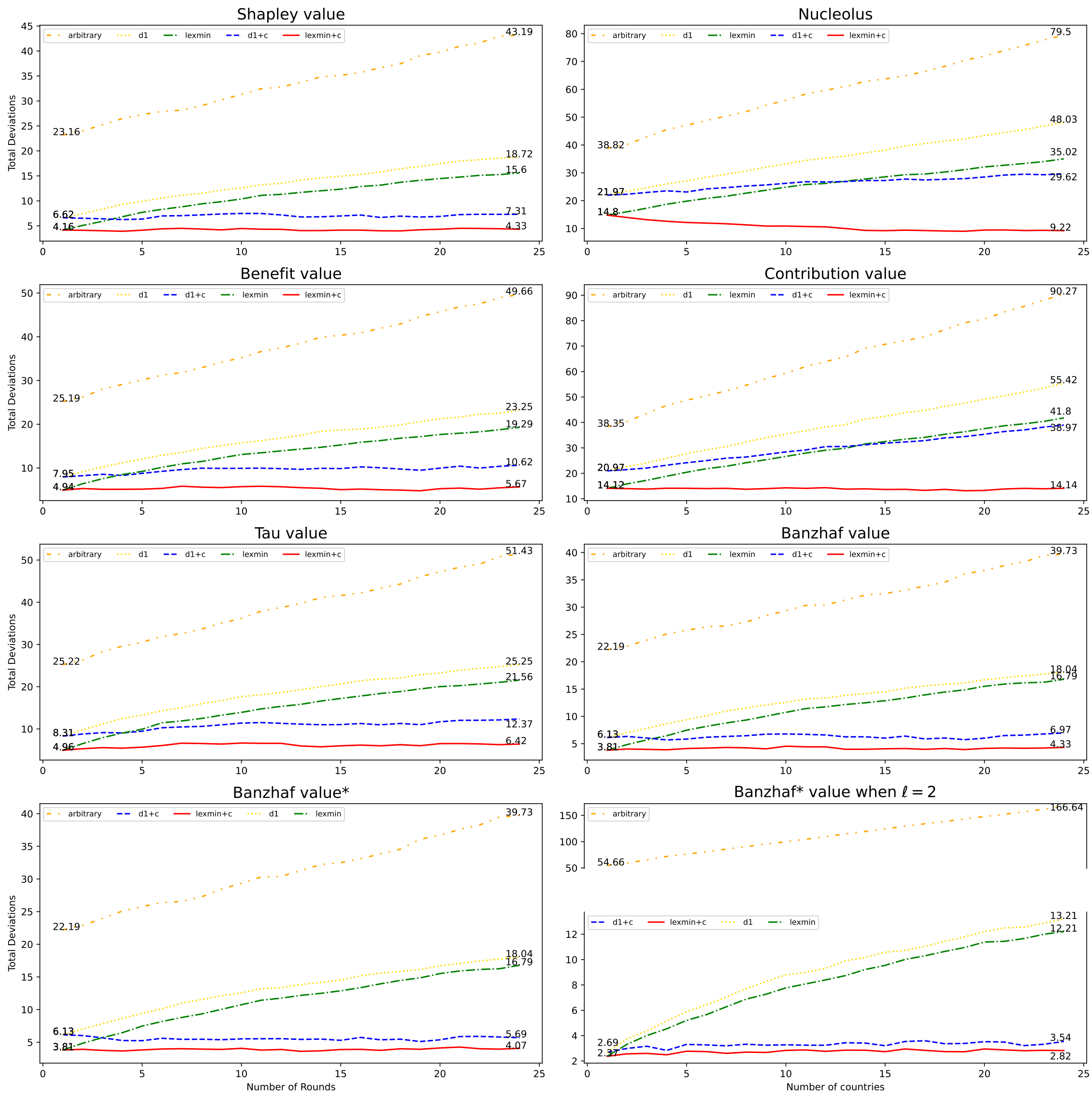

Figure 6.12: Average credits for each of the seven solution concepts under the five different scenarios for **varying** country sizes, where the number of countries is $n = 8$ and the period ranges from 1 to 24. For comparison, the lower right figure displays a result from Chapter 4 for $\ell = 2$, namely for the Banzhaf* value, which yields the lowest average credits compared to the other six solution concepts for $\ell = 2$.

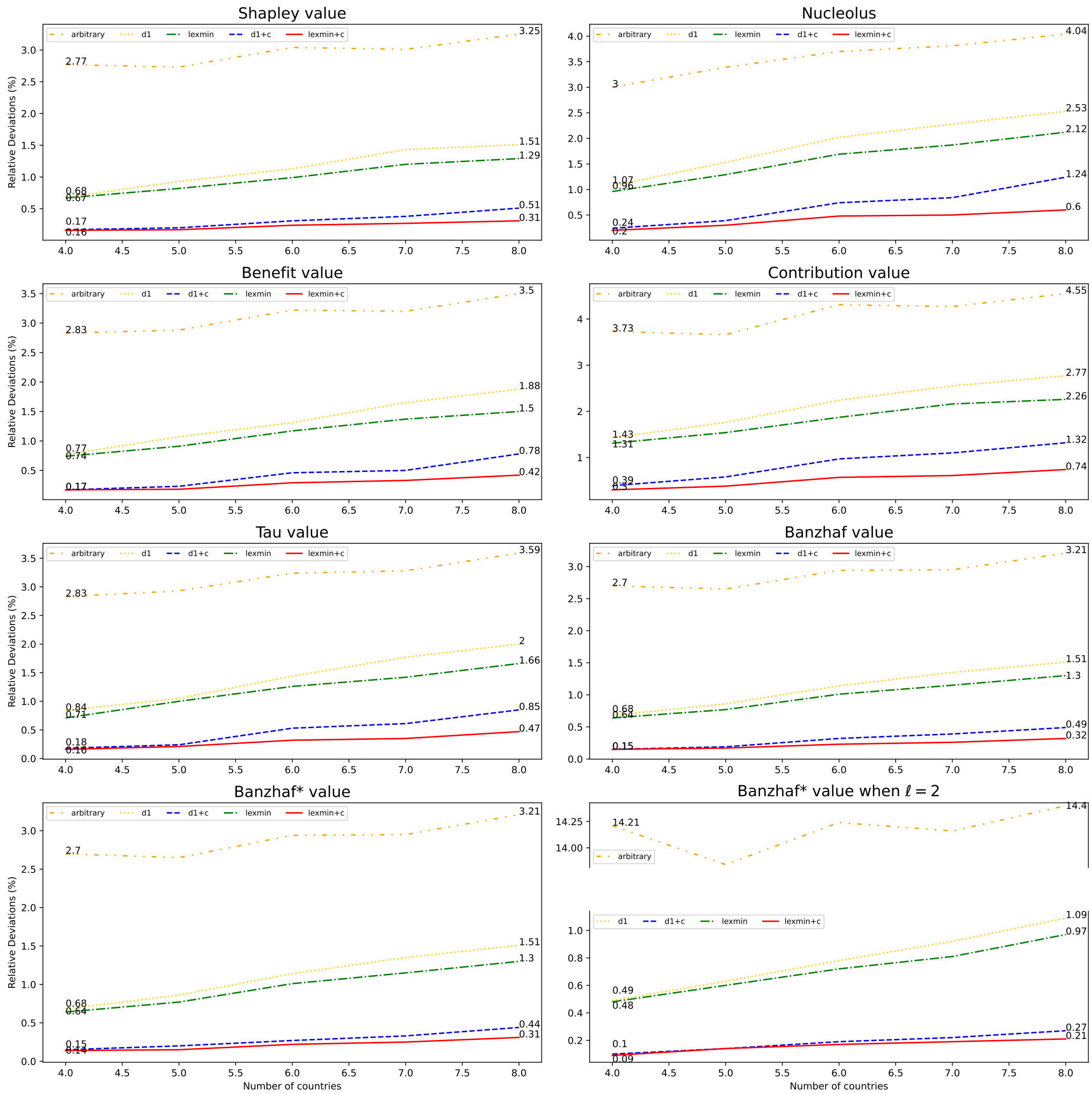

Figure 6.13: Average total relative deviations, leaving out the incomplete simulation instances under the five different scenarios for **equal** country sizes, where the number of countries ranges from 4 to 8. We note that this figure is identical to Figure 6.1 that includes all simulation instances.

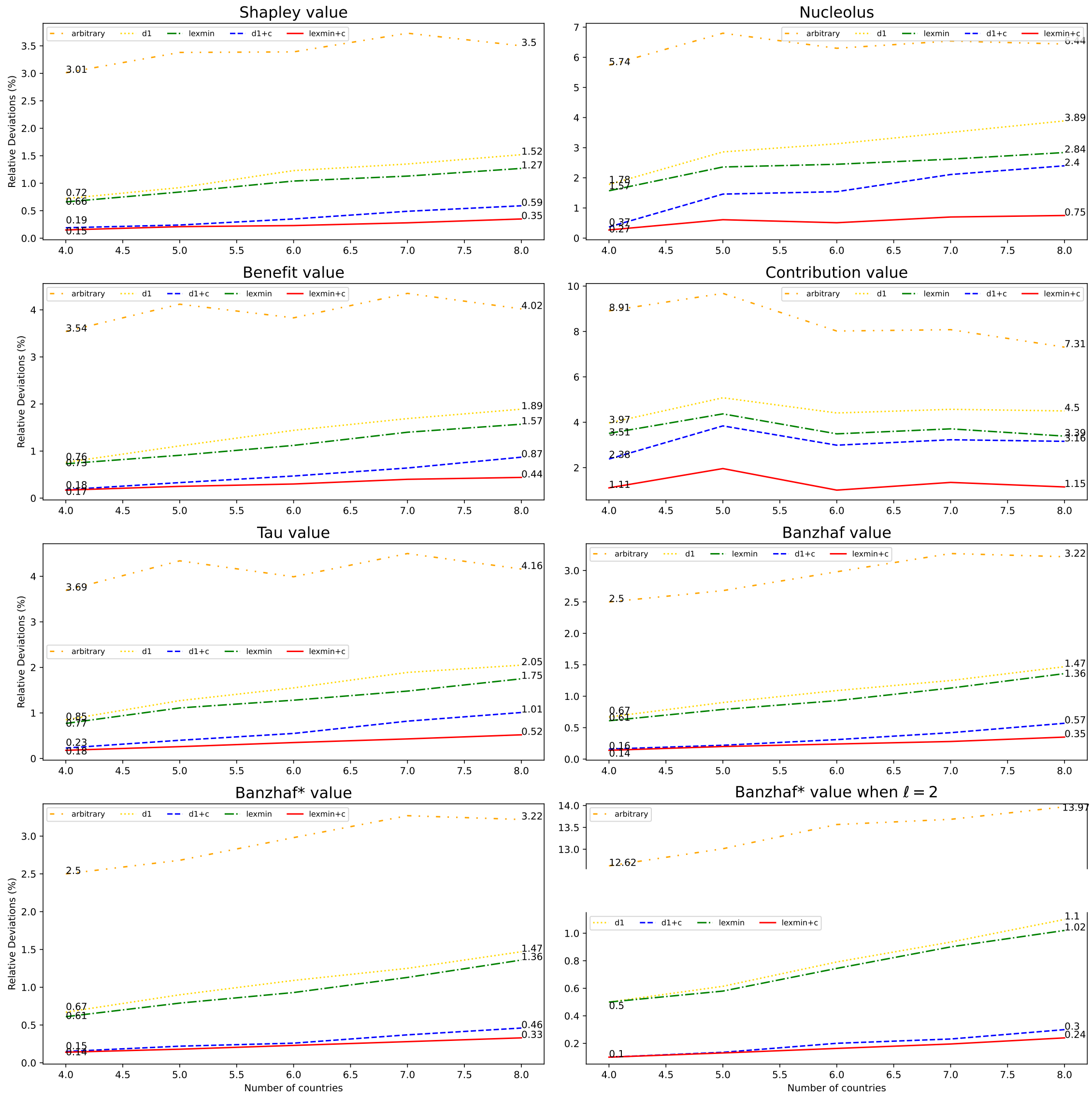

Figure 6.14: Average total relative deviations, leaving out the incomplete simulation instances under the five different scenarios for **varying** country sizes, where the number of countries ranges from 4 to 8. We note that this figure is identical to Figure 6.2 that includes all simulation instances.

concepts and selecting weakly or strongly optimal solutions when n is large.

On a side note, Table 6.10 shows that computation time generally increases at an expected rate as n grows, with a few notable exceptions. In particular, Gurobi's ILP solver unexpectedly performs significantly better for $n = 7$ than for $n = 8$, especially in the *lexmin* and *lexmin+c* scenarios. We do not have a clear explanation for this behavior; it may be due to internal heuristics or optimization strategies within Gurobi, which remain opaque to us.

CPU time / n	4	5	6	7	8
data preparation	0.01	0.01	0.01	0.01	0.01
cycle searches	0.05	0.05	0.05	0.05	0.04
Shapley value	125.53	248.73	439.40	813.90	1866.37
Nucleolus	117.04	225.62	451.42	826.36	1636.94
Benefit value	79.98	95.92	105.66	107.84	107.59
Contribution value	86.49	95.15	106.44	111.92	126.55
Banzhaf value	119.46	230.53	425.37	799.56	1761.88
Banzhaf* value	106.67	205.56	421.12	825.80	1666.58
Tau value	122.75	227.02	469.76	861.32	1751.84
total: arbitrary	97.51	183.66	298.02	557.04	1174.41
total: d1	120.75	200.26	418.80	676.52	1209.09
total: lexmin	137.87	218.61	465.39	570.31	1266.81
total: d1+c	117.32	209.42	422.11	678.10	1339.39
total: lexmin+c	129.09	226.79	512.94	755.40	1344.49

Table 6.10: Average CPU time for a single 24-round simulation instance, broken down into the different computational tasks for equal country sizes. Here, the times for data preparation and cycle searches 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, which includes computing the initial allocations, and only excludes the time for data preparation and cycle searches.

6.5 Concluding Remarks

Our simulations showed that selecting arbitrary optimal solutions for $\ell = 3$ yields the least balanced solution out of five choices of optimal solutions for IKEPs according to our three stability measures, namely, total relative deviations, maximum relative deviations and steady relative deviations. However, using strongly optimal solutions leads to a significant improvement in stability for IKEPs compared to weakly close optimal solutions. Moreover, we showed that this improvement is even more significant when the number of countries is large and credits are involved. Among the six solution concepts and one variant, the Banzhaf* value, when credits are integrated into games, achieves the lowest deviations from our target allocations across all three stability measures. These conclusions always hold across three stability measures both for equal and varying country sizes.

For $\ell = 3$, the average number of transplants under arbitrary optimal solutions is slightly higher than the other four choices of optimal solutions by 4.16 kidney transplants for equal country sizes and 4.24 kidney transplants for varying country sizes. However, no significant differences were observed in the average number of transplants among the other four choices of optimal solutions.

CHAPTER 7

Conclusions

In this thesis, we introduced a class of partitioned matching games for $\ell = 2$, partitioned permutation games for $\ell = \infty$ and partitioned ℓ-permutation games for $\ell \geq 3$. We not only showed the theoretical results for $\ell = \infty$, but also presented simulation results for a large number of countries with the identical simulation setup for $\ell \in \{2, 3, \infty\}$.

7.1 Theoretical Results

Recall that in Chapter 2, we surveyed complexity dichotomies of the core on three problems (P1–P3; see Chapter 2), namely the core membership, core non-emptiness and core allocation computation, for b-assignment games, b-matching games and partitioned matching games. We extended these theoretical results to partitioned ℓ-permutation games where $\ell \in \{2, 3, \ldots, \infty\}$ and summarize all known and new theoretical results for IKEPs, subject to maximum cycle length ℓ in Table 7.1.

	$\ell = 2$	$\ell \in \{3, 4, \ldots\}$	$\ell = \infty$
finding an optimal solution	poly [1]	NP-h [1]	poly [1]
testing core nonemptyiness	poly if $c \leq 2$ coNP-h if $c \geq 3$ [95]	NP-h	poly [89]
finding a core allocation	poly if $c \leq 2$ coNP-h if $c \geq 3$ [95]	NP-h	poly [90]
deciding core membership	poly if $c \leq 2$ coNP-c if $c \geq 3$ [95]	NP-h	poly if $c = 1$ coNP-c if $c \geq 2$
finding a weakly close solution to target allocation	poly [2]	NP-h	NP-h
finding a strongly close solution to target allocation	poly [2]	NP-h	NP-h

Table 7.1: Survey of complexity results relevant for IKEPs, for fixed cycle length $\ell = 2$, $\ell \in \{3, 4, \ldots\}$ 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 purple are new results shown in this thesis, whereas non-referenced results follow directly from the referenced results.

7.2 Comparison of Simulation Results

We now turn to summarizing the IKEPs simulation results for $\ell \in \{2, 3, \infty\}$. In this section, all conclusions and comparisons are based on three different values of ℓ, whereas in Chapters 5 and 6, conclusions and comparisons were made only between two specific values of ℓ. Using an identical simulation setup for all simulations, we unsurprisingly found that, at the high level, the following conclusions hold for $\ell \in \{2, 3, \infty\}$:

- An arbitrary optimal solution is the least balanced solution out of five choices of optimal solutions both according to our relative total deviations and maximum relative deviations across all initial allocations and numbers of countries.
- Strongly close optimal solutions consistently perform better than weakly close optimal solutions in terms of both total relative deviations and maximum relative deviations, across all initial allocations, choices of optimal solutions and numbers of countries.
- Introducing a credit-compensation system makes our solutions more balanced. As the number of countries grows, the improvement is even greater. Specifi-

cally, strongly close optimal solutions with the credit system (*lexmin+c*) yield the lowest deviations out of five choices of optimal solutions both in terms of relative total deviations and maximum relative deviations. They are lower than those of strongly close optimal solutions (*lexmin*). Likewise, weakly close optimal solutions with the credit system (*d1+c*) exhibit lower deviations compared to weakly close optimal solutions without the credit system (*d1*). Furthermore, weakly close optimal solutions with the credit-compensation system (*d1+c*) provide lower deviations both in terms of relative total deviations and maximum relative deviations, compared to strongly close optimal solutions without credits (*lexmin*).

- The Banzhaf* value (see Chapter 2), that incorporates credits into games, consistently performs the best in terms of total relative deviations and maximum relative deviations across all choices of optimal solutions and numbers of countries compared to other six solution concepts. In contrast, the contribution value consistently performs the worst across all initial allocations and numbers of countries according to our stability measures.

- The average number of kidney transplants stays the same across all initial allocations regardless of whether weakly or strongly close optimal solutions are used, or whether a credit-compensation system is applied for $\ell \in \{2, 3, \infty\}$.

  We now compare the main results for $\ell = 2$, $\ell = 3$ and $\ell = \infty$ in terms of total relative deviations and the number of kidney transplants.

7.2.1 Total Relative Deviations

We begin by comparing simulation results for three different values of ℓ in the most two distinctive scenarios (the *arbitrary* and *lexmin+c*), using our main measure, namely, the average total relative deviations. As mentioned, the *arbitrary* and *lexmin+c* scenarios generally yield the highest and lowest average total relative deviations respectively. Therefore, we compare simulation results across $\ell = 2$, $\ell = \infty$ and $\ell = 3$ on these two scenarios for equal country sizes. We obtain similar simulation results for the case of varying country sizes.

- In *arbitrary* scenarios, $\ell = 2$ exhibits the highest total relative deviations, whereas $\ell = 3$ and $\ell = \infty$ show significantly lower deviations; see Figure 7.1.

- In *lexmin+c* scenarios, $\ell = 2$ exhibits the lowest total relative deviations, followed by $\ell = 3$ with the second highest deviations and $\ell = \infty$ with the highest; see Figure 7.1. The other three scenarios display consistent patterns across all initial allocations and numbers of countries.

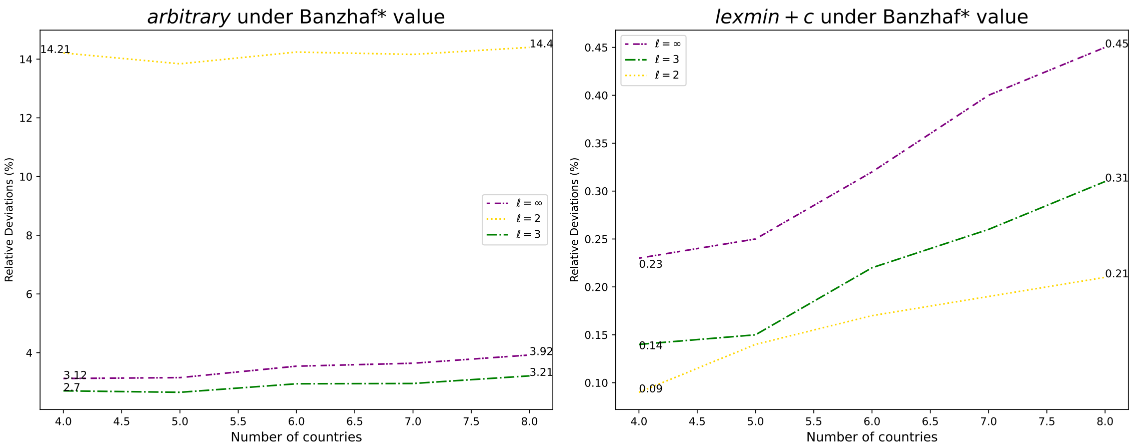

Figure 7.1: Average total relative deviations for *arbitrary* and *lexmin+c* when Banzhaf* value is selected for $\ell \in \{2, 3, \infty\}$ with equal country sizes.

7.2.2 Number of Kidney Transplants

Recall that, in kidney exchange, our primary goal is to maximize the number of transplants. We now compare the number of kidney transplants achieved under $\ell = 2$, $\ell = \infty$ and $\ell = 3$ for equal country sizes. The results presented below also hold for the case of varying country sizes.

- For $\ell \in \{2, 3, \infty\}$, $\ell = \infty$ presents the highest average number of kidney transplants across 100 instances, reaching 1775 out of a total of 2000 patient-donor pairs. In contract, $\ell = 2$ and $\ell = 3$ result in 1216 and 1249 kidney transplants, respectively. Moving from $\ell = 2$ to $\ell = 3$ leads to an average increase of 2.73%, while changing from $\ell = 2$ to $\ell = \infty$ results in a substantial increase of 45.99% on average, across all initial allocations, choices of optimal solutions and numbers of countries.

- Even though $\ell = \infty$ yields significant gains in the number of kidney transplants, the resulting cycle length could exceed 400, which is highly unrealistic in practice. Nonetheless, as mentioned, our aim was to obtain a general simulation results for IKEPs and investigate how stability and the number of kidney transplants vary as ℓ shifts from realistic values to the extreme case when $\ell = \infty$.

- For $\ell = 3$, on average, over 100 instances, length-3 cycles result in 64 kidney transplants in the first round when $n = 8$ and Banzhaf* value is selected under *lexmin+c*, while for the remaining 23 rounds, they contribute a total of 90 kidney transplants. Overall, across 24 rounds, length-3 cycles account for 12.39% of all kidney transplants.

7.3 Future Research

We now turn to a discussion about future research on IKEPs from both theoretical and simulation perspectives.

7.3.1 Complexity Aspects of IKEPs

We first state some theoretical open problems that resulted from our research.

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

Open Problem 2. *Determine the complexity of recognizing quasi-balanced games for partitioned matching games and partitioned ℓ-permutation games when $\ell \geq 3$.*

With respect to Open Problem 2, the cases of permutation games and partitioned permutation games have already been solved. Recall that both permutation games and partitioned permutation games always have a non-empty core. Therefore, each such game is balanced according to Bondareva–Shapley Theorem [52, 53], and naturally, it is quasi-balanced as well. However, for partitioned ℓ-permutation games, the core can be empty for $\ell \geq 2$; see examples in Chapter 2.

7.3.2 Simulation Aspects of IKEPs

We now shift our focus to future research directions from the simulation perspective.

In practice, longer maximum cycle lengths are permitted, such as 4 in Netherlands and 7 in Czech Republic. Therefore, all the above simulation results for $\ell \in \{2, 3, \infty\}$ are also interesting to research for a setting with exchange cycles for $\ell \in \{4, 5\}$.

Open Problem 3. *What are the effects on the number of transplants, fairness, and stability observed in simulations when longer cycles are permitted in IKEPs?*

To do meaningful experiments for a large number of countries, computational challenges must be overcome when longer cycles are allowed. Recall the aforementioned NP-hard result of [1] for the case where $\ell \geq 3$ and thus we used ILPs based on cycle formulations to compute and select optimal solutions for $\ell = 3$. So, when longer cycles are allowed, our ILPs are required to search not only for length-2 and length-3 cycles, as in the case of $\ell = 3$, but also for all longer cycles up to the maximum allowed length ℓ. As a result, finding optimal solutions becomes even more challenging for $\ell \in \{4, 5\}$ compared to the case of $\ell = 3$ for the same compatibility graph.

There might be multiple optimal solutions, regardless whether they are weakly close optimal solutions or strongly close optimal solutions, which were observed in our simulations for $\ell = \infty$. When multiple strongly or weakly close optimal solutions exist, our simulations simply used the one returned by our solver or our algorithm without applying any further selection criteria. In the future, further criteria should be considered when there are multiple solutions under a given scenario.

Open Problem 4. *How do hierarchical criteria impact the number of transplants, fairness, and stability in simulation studies of IKEPs?*

In Europe, maximizing the number of transplants is the first objective (as in our setting). We also plan to consider compatibility graphs with *weights* $w(i, j)$ on the arcs (i, j) representing the utility of transplant (i, j). Computing a maximum-weight solution that minimizes the largest deviation d_1 now becomes NP-hard [95].

However, instead of finding the set of maximum-weight ℓ-cycle packings directly, we could still consider the set of maximum ℓ-cycle packings to maximize the number of transplants. Within this set, we can then find a maximum-weight ℓ-cycle packing $w(\mathcal{C})$ that is either weakly or strongly close to our original target allocations. The main challenge is to set weights $w(i, j)$ appropriately, since optimization policies may vary widely in national KEPs. However, further scores are based on different objectives, such as improving the quality of transplants, easing the complexity of logistics, or giving priority to highly sensitized patients; see [14] for further details.

When $\ell = 3$, due to Theorem 1, computing $2^n - 1$ v-values for hard-to-compute initial allocations, such as the Shapley value, nucleolus, Banzhaf value and tau value, accounted for the majority of our computational time; see Chapter 6.

Open Problem 5. *How do approximation algorithms for computing solution concepts affect the number of transplants, fairness, and stability in simulation studies of IKEPs? Additionally, how do these approximation algorithms affect simulation outcomes in credit-adjusted games under different solution concepts?*

For future research, a natural direction is to investigate the application of approximation algorithms [96–98] to compute hard-to-compute solution concepts under IKEPs. Moreover, we are interested in studying the behaviors of games with an alternative credit system by directly incorporating credits into games when considering the approximation algorithms for solution concepts.

Additionally, we observed that for $\ell = \infty$, the resulting cycle lengths could be even longer than 400, as mentioned, which is highly vulnerable and unrealistic in practice. Therefore, we aim to minimize the exchange cycle bound ℓ without decreasing the number of transplants.

Open Problem 6. *To what extent can we avoid the use of longer cycles in IKEPs while still achieving the same number of transplants? Furthermore, how does this impact stability?*

In other words, our goal is to preserve the same number of transplants while minimize the maximum cycle length, thereby avoiding excessively long and impractical exchange cycles. This line of research similarly extends to any constant $\ell \geq 3$. Identifying such trade-offs could inform the design of more practical and implementable exchange mechanisms.

Last but not least, in our simulations for $\ell \in \{2, 3, \infty\}$, we only allowed transplants to occur within exchange cycles. However, we are interested in incorporating chains into our exchange model in the future.

Open Problem 7. *What impact does the inclusion of chains have on simulation outcomes such as efficiency, fairness, and stability in IKEPs?*

Bibliography

[1] D. J. Abraham, A. Blum, and T. Sandholm, “Clearing algorithms for barter exchange markets: enabling nationwide kidney exchanges,” *Proceedings EC 2007*, pp. 295–304, 2007.

[2] M. Benedek, P. Biró, W. Kern, D. Pálvölgyi, and D. Paulusma, “Partitioned matching games for international kidney exchange,” *Mathematical Programming*, pp. 1–36, 2025.

[3] M. Benedek, P. Biró, D. Paulusma, and X. Ye, “Computing balanced solutions for large international kidney exchange schemes,” *Autonomous Agents and Multi-Agent Systems*, vol. 38, p. 18, 2024.

[4] P. Biró, W. Kern, D. Paulusma, and P. Wojuteczky, “The stable fixtures problem with payments,” *Games and Economic Behavior*, vol. 108, pp. 245–268, 2018.

[5] L. Farczadi, K. Georgiou, and J. Könemann, “Network bargaining with general capacities,” *Proceedings ESA 2013, Lecture Notes in Computer Science*, vol. 8125, pp. 433–444, 2013.

[6] T. C. Koopmans and M. Beckmann, “Assignment problems and the location of economic activities,” *Econometrica*, vol. 25, pp. 53–76, 1957.

[7] L. S. Shapley and M. Shubik, “The assignment game I: The core,” *International Journal of Game Theory*, vol. 1, pp. 111–130, 1971.

[8] M. Sotomayor, “The multiple partners game,” in *Equilibrium and dynamics: essays in honor of David Gale*, Macmillan Press Ltd, New York, 1992.

[9] X. Deng, T. Ibaraki, and H. Nagamochi, “Algorithmic aspects of the core of combinatorial optimization games,” *Mathematics of Operations Research*, vol. 24, pp. 751–766, 1999.

[10] X. Klimentova, A. Viana, J. a. P. Pedroso, and N. Santos, “Fairness models for multi-agent kidney exchange programmes,” *Omega*, vol. 102, p. 102333, 2021.

[11] P. Biró, M. Gyetvai, X. Klimentova, J. P. Pedroso, W. Pettersson, and A. Viana, "Compensation scheme with Shapley value for multi-country kidney exchange programmes," *Proceedings ECMS 2020*, pp. 129–136, 2020.

[12] P. Biró, B. Haase-Kromwijk, T. Andersson, and et al., "Building kidney exchange programmes in Europe – an overview of exchange practice and activities," *Transplantation*, vol. 103, pp. 1514–1522, 2019.

[13] M. Barkel, R. Colley, M. Delorme, D. Manlove, and W. Pettersson, "Operational research approaches and mathematical models for kidney exchange: A literature survey and empirical evaluation," 2025.

[14] P. Biró, J. van de Klundert, D. Manlove, W. Pettersson, T. Andersson, L. Burnapp, P. Chromy, P. Delgado, P. Dworczak, B. Haase, A. Hemke, R. Johnson, X. Klimentova, D. Kuypers, A. Nanni Costa, B. Smeulders, F. Spieksma, M. O. Valentín, and A. Viana, "Modelling and optimisation in european kidney exchange programmes," *European Journal of Operational Research*, vol. 291, no. 2, pp. 447–456, 2021.

[15] D. F. Manlove and G. O'malley, "Paired and altruistic kidney donation in the uk: Algorithms and experimentation," *Journal of Experimental Algorithmics (JEA)*, vol. 19, pp. 1–21, 2015.

[16] M. Kim, H. Ro, Y. Kim, H. Park, J. Jeong, H. Jeon, H. Kim, M. Park, K.-H. Oh, Y. Kim, *et al.*, "Management of patients on the waiting list for deceased donor kidney transplantation," in *Transplantation proceedings*, vol. 44, pp. 66–71, Elsevier, 2012.

[17] G. A. Böhmig, J. Fronek, A. Slavcev, G. F. Fischer, G. Berlakovich, and O. Viklicky, "Czech-Austrian kidney paired donation: first european cross-border living donor kidney exchange," *Transplant International*, vol. 30, pp. 638–639, 2017.

[18] M. O. Valentín, M. Garcia, A. N. Costa, C. Bolotinha, L. Guirado, F. Vistoli, A. Breda, P. Fiaschetti, and B. Dominguez-Gil, "International cooperation for kidney exchange success," *Transplantation*, vol. 103, pp. 180–181, 2019.

[19] M. Carvalho and A. Lodi, "A theoretical and computational equilibria analysis of a multi-player kidney exchange program," *European Journal of Operational Research*, vol. 305, pp. 373–385, 2023.

[20] M. Carvalho, A. Lodi, J. a. P. Pedroso, and A. Viana, "Nash equilibria in the two-player kidney exchange game," *Mathematical Programming*, vol. 161, pp. 389–417, 2017.

[21] B. Smeulders, D. A. Blom, and F. C. Spieksma, "Identifying optimal strategies in kidney exchange games is σ 2 p-complete," *Mathematical Programming*, vol. 203, no. 1, pp. 347–368, 2024.

[22] Z. Sun, T. Todo, and T. Walsh, "Fair pairwise exchange among groups," *Proceedings IJCAI 2021*, pp. 419–425, 2021.

[23] D. Blom, B. Smeulders, and F. Spieksma, “Rejection-proof mechanisms for multi-agent kidney exchange,” *Games and Economic Behavior*, vol. 143, pp. 25–50, 2024.

[24] R.-S. Mincu, P. Biró, M. Gyetvai, A. Popa, and U. Verma, “Ip solutions for international kidney exchange programmes,” *Central European Journal of Operations Research*, vol. 29, p. 403–423, Aug. 2020.

[25] R. Colley, D. Manlove, D. Paulusma, and M. Zhang, “Complexity and manipulation of international kidney exchange programmes with country-specific parameterss,” *arXiv preprint arXiv:2506.04092*, 2025.

[26] I. Ashlagi and A. E. Roth, “New challenges in multihospital kidney exchange,” *American Economic Review*, vol. 102, pp. 354–359, 2012.

[27] I. Ashlagi and A. E. Roth, “Free riding and participation in large scale, multi-hospital kidney exchange,” *Theoretical Economics*, vol. 9, pp. 817–863, 2014.

[28] I. Ashlagi, F. Fischer, I. A. Kash, and A. D. Procaccia, “Mix and match: A strategyproof mechanism for multi-hospital kidney exchange,” *Games and Economic Behavior*, vol. 91, pp. 284 – 296, 2015.

[29] A. Blum, I. Caragiannis, N. Haghtalab, A. D. Procaccia, E. B. Procaccia, and R. Vaish, “Opting into optimal matchings,” *Proceedings SODA 2017*, pp. 2351–2363, 2017.

[30] C. Hajaj, J. P. Dickerson, A. Hassidim, T. Sandholm, and D. Sarne, “Strategy-proof and efficient kidney exchange using a credit mechanism,” *Proceedings AAAI 2015*, pp. 921–928, 2015.

[31] P. Toulis and D. C. Parkes, “Design and analysis of multi-hospital kidney exchange mechanisms using random graphs,” *Games and Economic Behavior*, vol. 91, pp. 360–382, 2015.

[32] N. Agarwal, I. Ashlagi, E. Azevedo, C. R. Featherstone, and Ö. Karaduman, “Market failure in kidney exchange,” *American Economic Review*, vol. 109, pp. 4026–4070, 2019.

[33] H. Aziz, Á. Cseh, J. P. Dickerson, and D. C. McElfresh, “Optimal kidney exchange with immunosuppressants,” *Proceedings AAAI 2021*, pp. 21–29, 2021.

[34] P. Biró, F. Klijn, X. Klimentova, and A. Viana, “Shapley-scarf housing markets: Respecting improvement, integer programming, and kidney exchange,” *Mathematics of Operations Research*, vol. to appear, 2023.

[35] P. Biró, D. F. Manlove, and R. Rizzi, “Maximum weight cycle packing in directed graphs, with application to kidney exchange programs,” *Discrete Mathematics, Algorithms and Applications*, vol. 1, pp. 499–518, 2009.

[36] A. E. Roth, T. Sönmez, and M. U. Ünver, “Kidney exchange,” *Quarterly Journal of Economics*, vol. 119, pp. 457–488, 2004.

[37] B. Peleg and P. Sudhölter, *Introduction to the theory of cooperative games*, vol. 34. Springer Science & Business Media, 2007.

[38] W. Pettersson and J. Trimble, "Kidney matching tools data set generator," 2021. `https://wpettersson.github.io/kidney-webapp/#/generator`.

[39] M. Benedek, P. Biró, G. Csáji, M. Johnson, D. Paulusma, and X. Ye, "Computing balanced solutions for large international kidney exchange schemes when cycle length is unbounded," in *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pp. 2153–2155, 2024.

[40] M. Benedek, P. Biró, M. Johnson, D. Paulusma, and X. Ye, "The complexity of matching games: A survey," *Journal of Artificial Intelligence Research*, vol. 77, pp. 459–485, 2023.

[41] M. Benedek, P. Biró, D. Paulusma, and X. Ye, "Computing balanced solutions for large international kidney exchange schemes," *Autonomous Agents and Multi-Agent Systems*, vol. 38, no. 1, p. 18, 2024.

[42] H. Peters, *Game theory: A Multi-leveled approach.* Springer, 2008.

[43] D. B. Gillies, "Solutions to general non-zero-sum games," *Annals of Mathematics Studies*, vol. 40, pp. 47–85, 1959.

[44] M. Maschler, B. Peleg, and L. S. Shapley, "Geometric properties of the kernel, nucleolus, and related solution concepts," *Mathematics of Operations Research*, vol. 4, pp. 303–338, 1979.

[45] L. S. Shapley, "Cores of convex games," *International Journal of Game Theory*, vol. 1, pp. 11–26, 1971.

[46] L. S. Shapley, "A value for n-person games," *Annals of Mathematical Studies*, vol. 28, pp. 307–317, 1953.

[47] D. Schmeidler, "The nucleolus of a characteristic function game," *SIAM Journal on Applied Mathematics*, vol. 17, pp. 1163–1170, 1969.

[48] M. Maschler, B. Peleg, and L. S. Shapley, "The kernel and bargaining set for convex games," *International Journal of Game Theory*, vol. 1, pp. 73–93, 1971.

[49] S. H. Tijs, "Bounds for the core of a game and the t-value," *Game Theory and Mathematical Economics*, pp. 123–132, 1981.

[50] E. B. Yanovskaya, "The nucleolus and the τ-value of interval games," *Contributions to Game Theory and Management*, vol. 3, no. 0, pp. 421–430, 2010.

[51] J. F. B. III, "Weighted voting doesn't work: A mathematical analysis," *Rutgers Law Review*, vol. 19, pp. 317–343, 1964.

[52] O. N. Bondareva, "Some applications of linear programming methods to the theory of cooperative games," *Problemy Kibernet*, vol. 10, p. 119, 1963.

[53] L. S. Shapley, "On balanced sets and cores," *Naval research logistics quarterly*, vol. 14, no. 4, pp. 453–460, 1967.

[54] T. Király and Z. Mészáros-Karkus, "Complexity of the NTU international matching game," *EGRES Technical Report*, vol. 2019-12, 2019.

[55] L. Gourvès, J. Monnot, and F. Pascual, "Cooperation in multiorganization matching," *Algorithmic Operations Research*, vol. 7, pp. 111–124, 2013.

[56] J. Staudacher and J. Anwander, "Conditions for the uniqueness of the gately point for cooperative games," *arXiv preprint arXiv:1901.01485*, 2019.

[57] D. Gately, "Sharing the gains from regional cooperation: A game theoretic application to planning investment in electric power," *International Economic Review*, pp. 195–208, 1974.

[58] K. Eriksson and J. Karlander, "Stable outcomes of the roommate game with transferable utility," *International Journal of Game Theory*, vol. 29, pp. 555–569, 2001.

[59] M. Grötschel, L. Lovász, and A. Schrijver, "The ellipsoid method and its consequences in combinatorial optimization," *Combinatorica*, vol. 1, pp. 169–197, 1981.

[60] L. G. Khachiyan, "A polynomial algorithm in linear programming," *Soviet Mathematics Doklady*, vol. 20, pp. 191–194, 1979.

[61] P. Biró, W. Kern, and D. Paulusma, "Computing solutions for matching games," *International Journal of Game Theory*, vol. 41, pp. 75–90, 2012.

[62] M. L. Balinski, "Integer programming: Methods, uses, computation," *Management Science*, vol. 12, pp. 253–313, 1965.

[63] G. L. Nemhauser and L. E. Trotter Jr., "Vertex packings: Structural properties and algorithms," *Mathematical Programming*, vol. 8, pp. 232–248, 1975.

[64] J. Edmonds, "Maximum matching and a polyhedron with $0,1$-vertices," *Journal of Research of the National Bureau of Standards Section B*, vol. 69B, pp. 125–130, 1965.

[65] L. Sanità and L. Verberk, "A note on the core of 2-matching games," *Operations Research Letters*, p. 107299, 2025.

[66] W. T. Tutte, "A short proof of the factor theorem for finite graphs," *Canadian Journal of Mathematics*, vol. 6, pp. 347–352, 1954.

[67] T. Solymosi and T. E. S. Raghavan, "An algorithm for finding the nucleolus of assignment games," *International Journal of Game Theory*, vol. 23, pp. 119–143, 1994.

[68] U. Faigle, W. Kern, S. P. Fekete, and W. Hochstättler, "The nucleon of cooperative games and an algorithm for matching games," *Mathematical Programming*, vol. 83, pp. 195–211, 1998.

[69] N. Chen, P. Lu, and H. Zhang, "Computing the nucleolus of matching, cover and clique games," *Proceedings AAAI 2012*, pp. 1319–1325, 2012.

[70] W. Kern and D. Paulusma, "Matching games: The least core and the nucleolus," *Mathematics of Operations Research*, vol. 28, pp. 294–308, 2003.

[71] D. Paulusma, *Complexity Aspects of Cooperative Games.* PhD thesis, University of Twente, Enschede, The Netherlands, 2001.

[72] L. Farczadi, *Matchings and Games on Networks.* PhD thesis, University of Waterloo, Ontario, Canada, 2015.

[73] J. Hardwick, *Graphical Algorithms for Finding the Nucleolus of Binary-Valued Matching Games.* PhD thesis, University of Illinois, Chicago, Illinois, USA, 2017.

[74] J. Könemann, K. Pashkovich, and J. Toth, "Computing the nucleolus of weighted cooperative matching games in polynomial time," *Mathematical Programing*, vol. 183, pp. 555–581, 2020.

[75] J. Könemann, J. Toth, and F. Zhou, "On the complexity of nucleolus computation for bipartite b-matching games," *Proceedings SAGT 2021, Lecture Notes in Computer Science*, vol. 12885, pp. 171–185, 2021.

[76] M. Bateni, M. Hajiaghayi, N. Immorlica, and H. Mahini, "The cooperative game theory foundations of network bargaining games," *Proceecings ICALP 2010, Lecture Notes in Computer Science*, vol. 6198, pp. 67–78, 2010.

[77] J. Könemann and J. Toth, "A general framework for computing the nucleolus via dynamic programming," *Proceedings SAGT 2020, Lecture Notes in Computer Science*, vol. 12283, pp. 307–321, 2020.

[78] V. V. Vazirani, "The general graph matching game: Approximate core," *Games and Economic Behavior*, vol. 132, pp. 478–486, 2022.

[79] H. Xiao, T. Lu, and Q. Fang, "Approximate core allocations for multiple partners matching games," *CoRR*, vol. abs/2107.01442, 2021.

[80] H. Aziz and B. de Keijzer, "Shapley meets Shapley," *Proceedings STACS 2014, LIPIcs*, vol. 25, pp. 99–111, 2014.

[81] N. Bousquet, "The Shapley value of matching games on trees," *Manuscript*, 2015.

[82] G. Greco, F. Lupia, and F. Scarcello, "Coalitional games induced by matching problems: Complexity and islands of tractability for the Shapley value," *Artificial Intelligence*, vol. 278, 2020.

[83] M. Benedek, "International kidney exchange scheme," 2021. https://github.com/blrzsvrzs/int_kidney_exchange.

[84] M. Delorme, S. García, J. Gondzio, J. Kalcsics, D. F. Manlove, W. Pettersson, and J. Trimble, "Improved instance generation for kidney exchange programmes," *Computers & Operations Research*, vol. 141, p. 105707, 2022.

[85] X. Ye, "International kidney exchange data," 2023. https://github.com/Arya1531/international_kidney_exchange_program/tree/main.

[86] X. Ye, "International kidney exchange data," *Durham University*, 2025. https://collections.durham.ac.uk/files/r28336h194z.

[87] B. Dezso, A. Jüttner, and P. Kovács, "LEMON - an open source C++ graph template library," *Proceedings WGT@ETAPS, Electronic Notes in Theoretical Computer Science*, vol. 264, pp. 23–45, 2010.

[88] M. Benedek, J. Fliege, and T. Nguyen, "Finding and verifying the nucleolus of cooperative games," *Mathematical Programming*, vol. 190, pp. 135–170, 2021.

[89] S. Tijs, T. Parthasarathy, J. A. Potters, and V. R. Prassad, "Permutation games: another class of totally balanced games," *OR Spektrum*, vol. 6, pp. 119–123, 1984.

[90] I. J. Curiel and S. H. Tijs, "Assignment games and permutation games," *Methods of Operations Research*, vol. 54, pp. 323–334, 1986.

[91] C. H. Papadimitriou and M. Yannakakis, "The complexity of restricted spanning tree problems," *Journal of the ACM*, vol. 29, pp. 285–309, 1982.

[92] J. Hein, T. Jiang, L. Wang, and K. Zhang, "On the complexity of comparing evolutionary trees," *Discrete Applied Mathematics*, vol. 71, pp. 153–169, 1996.

[93] G. Hickey, F. Dehne, A. Rau-Chaplin, and C. Blouin, "Spr distance computation for unrooted trees," *Evolutionary Bioinformatics*, vol. 4, pp. EBO–S419, 2008.

[94] R. Gurjar, A. Korwar, J. Messner, S. Straub, and T. Thierauf, "Planarizing gadgets for perfect matching do not exist," *Proceedings MFCS 2012, Lecture Notes in Computer Science*, vol. 7464, pp. 478–490, 2012.

[95] P. Biró, W. Kern, D. Pálvölgyi, and D. Paulusma, "Generalized matching games for international kidney exchange," *Proceedings AAMAS 2019*, pp. 413–421, 2019.

[96] J. Bilbao, J. Fernandez, A. J. Losada, and J. Lopez, "Generating functions for computing power indices efficiently," *Top*, vol. 8, no. 2, pp. 191–213, 2000.

[97] J. M. Bilbao, J. R. Fernandez, N. Jiménez, and J. J. Lopez, "Voting power in the european union enlargement," *European Journal of Operational Research*, vol. 143, no. 1, pp. 181–196, 2002.

[98] I. Gemp, M. Lanctot, L. Marris, Y. Mao, E. Duéñez-Guzmán, S. Perrin, A. Gyorgy, R. Elie, G. Piliouras, M. Kaisers, *et al.*, "Approximating the core via iterative coalition sampling," *arXiv preprint arXiv:2402.03928*, 2024.