Mathematical Programming (2024) 203:347–368
https://doi.org/10.1007/s10107-021-01748-6

FULL LENGTH PAPER

Series B

# Identifying optimal strategies in kidney exchange games is $\Sigma_2^p$-complete

**B. Smeulders[1] · D. A. M. P. Blom[1] · F. C. R. Spieksma[1]**

Received: 29 October 2020 / Accepted: 22 November 2021 / Published online: 21 January 2022


## Abstract

In Kidney Exchange Games, agents (e.g. hospitals or national organizations) have control over a number of incompatible recipient-donor pairs whose recipients are in need of a transplant. Each agent has the opportunity to join a collaborative effort which aims to maximize the total number of transplants that can be realized. However, the individual agent is only interested in maximizing the number of transplants within the set of recipients under its control. Then, the question becomes: which recipient-donor pairs to submit to the collaborative effort? We model this situation by introducing the Stackelberg Kidney Exchange Game, a game where an agent, having perfect information, needs to identify a strategy, i.e., to decide which recipient-donor pairs to submit. We show that even in this simplified setting, identifying an optimal strategy is $\Sigma_2^p$-complete, whenever we allow exchanges involving at most a fixed number $K \geq 3$ pairs. However, when we restrict ourselves to pairwise exchanges only, the problem becomes solvable in polynomial time.

**Keywords** Kidney exchange programmes · Computational complexity · Stackelberg games · $\Sigma_2^p$

**Mathematics Subject Classification** 90C60 · 91A65

A one-page abstract corresponding to this manuscript appeared in the proceedings of SAGT 2020 [20].

✉ B. Smeulders
b.smeulders@tue.nl

D. A. M. P. Blom
d.a.m.p.blom@tue.nl

F. C. R. Spieksma
f.c.r.spieksma@tue.nl

[1] Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands

## 1 Introduction

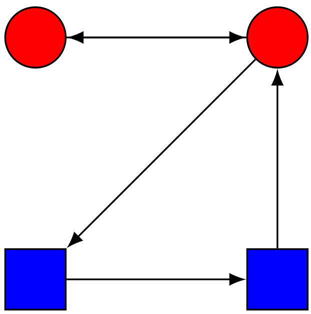
Kidney Exchange Programmes (KEPs) play a growing role in the improvement of lives of many patients suffering from end stage renal disease. Given the absence of artificial kidneys, given the lack of kidneys coming from deceased donors, and the fact that it is possible to lead a normal life with a single kidney, donation from living donors is an increasingly popular option (see [19]). For this option to succeed, compatibility of donor and recipient (in terms of blood type and immunological properties) is crucial. Recipients who have a willing, but incompatible, donor can be helped through a Kidney Exchange Program. In such a program, recipient-donor couples, referred to as *pairs*, are present; the donor of a pair is willing to donate her/his kidney to some recipient, provided that the corresponding recipient receives a kidney from some other donor. Such programmes have been established around the world; we refer to [7] for a recent overview of this practice in Europe.

The organization of collaboration between different transplant centres, organizations or countries (we use the term *agent* for an entity in control of a set of pairs) is a delicate matter. Such collaborations face many challenges, varying from legal considerations to the alignment of medical procedures. On one hand, it is clear that collaboration increases the possibilities for matching donors with recipients, and thus leads to more transplants and better overall recipient outcomes. On the other hand, these benefits of cooperation may be shared unequally, and in some cases individual agents may even lose transplants when combining pools.

Consider as an illustration the example from [4], depicted in Fig. 1. Each node in this graph corresponds to a patient-donor couple, and an arc from one node to another means that the donor of the first node is compatible with the patient from the second node. A set of node-disjoint cycles in this graph corresponds to a set of realizable transplants. In Fig. 1, two agents, red and blue, each have a private pool consisting of two pairs. Red can transplant two patients internally, while blue has no internal matches. However, if both agents combine their pools, three transplants are possible, two for blue patients and one for red. It is thus in red's best interest not to participate in the collaboration.

The example above illustrates an important result: no mechanism exists guaranteeing a solution that is both socially optimal (meaning delivering a maximum number of transplants) and individually rational (meaning that each agent acts solely in its own

**Fig. 1** An example illustrating that there does not exist a mechanism guaranteeing a social welfare optimum that is individually rational



interest). Indeed, to give red an incentive to contribute any of its pairs, red must be guaranteed at least two transplants in the overall solution. However, any solution in which red receives two transplants is not socially optimal.

In case multiple agents can collaborate in a KEP, each agent is thus faced with the question whether to participate and if so, which of its pairs to submit to the common pool. This is a kidney exchange game.

Issues surrounding collaboration in KEPs have been studied in the literature. Ashlagi and Roth [4] and Toulis and Parkes [21] study cooperation in large random graphs, where blood-type compatibility is the sole limiting factor; they find that individually rational solutions exist that are close to socially optimal. Blum et al. [11] obtain similar results for arbitrary graphs. They show that with high probability, individually rational solutions are socially optimal. Additionally, they show that any socially optimal solution is with high probability close to individually rational.

Strategy-proof mechanisms for multi-agent kidney exchange are studied in [5]. They show that even for two players and a maximum cycle length of 2, no strategy-proof socially optimal mechanism can exist. They furthermore prove approximation bounds for deterministic and randomized mechanisms and propose a strategy-proof randomized mechanism that guarantees half of the optimal social welfare in the worst case. Caragiannis et al. [12] strengthen the bounds for randomized mechanisms and propose a strategy-proof randomized mechanism guaranteeing two-thirds of the maximum social welfare.

Recently, much attention has gone to mechanisms for multi-period settings. Through various credit systems, agents are incentivized to contribute pairs to a common pool. Credits are earned through contributing pairs, and agents with the most credits are advantaged in the common pool. In this way, full cooperation is stimulated, as lost transplants in a matching run will be offset later on through the credit system. Examples of such credit systems can be found in [15], who describe a strategy-proof and socially optimal mechanism based on a credit system depending upon the number of contributed pairs. However, this mechanism requires knowledge of the expected arrival rate of pairs for each agent. Agarwal et al. [2] describe a credit system where agents are rewarded for adding pairs based on the expected marginal added transplants of adding that pair to a common pool. The credit systems of [17] and [9] depend on the complete set of contributed nodes of a player. Klimentova [17] consider the maximum achievable number of transplants for an agent, as well as a system based on their marginal contribution to maximum transplants in the common pool. Biró et al. [9] investigate a credit system based on Shapley-values. Biró et al. [8] investigate the computational complexity of computing KEP solutions where agents receive transplants as close as possible to some target value while maximizing the overall number of transplants.

Carvalho et al. [13] and [14] study collaboration in KEPs as a non-cooperative game. They show that, when the cycle length equals 2, there exists a socially optimal Nash-equilibrium, and that this equilibrium can be computed in polynomial time. This problem is closely related to our setting, and we will elaborate on the similarities and differences in Sect. 2.

In this paper, we study the problem faced by an individual agent in a collaborative KEP. How easy, or how hard, is it for an agent to determine its individual rational strategy? Specifically, we consider the situation where an agent must decide which of its pairs to match internally, and which pairs to add to a common pool. In this common pool, the number of transplants is then maximized. The agent's goal is to maximize the number of its own pairs that are transplanted. Such cooperations exist in practice. For example, Italy, Portugal and Spain run a joint KEP where pairs unmatched in national programs are then matched through a common pool [10,22]. These countries could decide not to match internally if they believe sending pairs to the common pools allows for new solutions benefiting themselves. Similar situations exist in the United States, where a variety of hospital, regional and national KEPs co-exist. Clearly, the computational complexity of the agent's goal of maximizing its own transplants is a relevant issue. Indeed, if an agent is not able to efficiently compute strategies that maximize its number of transplants, the design of mechanisms that guide collaboration in KEPs can be affected. For instance, such a mechanism can safely assume that agents are not able to efficiently identify such strategies.

The paper is organized as follows. In the next section, we formally define the problem. In Sect. 3 we describe our main result. We will show that, even if an agent knows exactly which pairs of other agents are present in the common pool, together with their respective compatibilities, the problem of deciding which pairs to contribute and which to match internally, to guarantee a given number of its pairs are transplanted, is $\Sigma_2^p$-complete. The class $\Sigma_2^p$ is a complexity class of decision problems that generalizes the traditional classes P and NP to a setting with two decision makers (or players/agents). It contains problems that can be expressed by a logical formula using two consecutive quantifiers, where the first quantifier is of the type "does there exist", while the second quantifier is of the type "for all". We refer to [3] for an introduction into computational complexity including the polynomial hierarchy. One practical implication of a problem being $\Sigma_2^p$-complete is that the existence of a compact Integer Program modelling the problem is unlikely, see [18] and [23]. Our result implies that it is computationally very hard for an agent who is solely interested in maximizing the number of transplants among its own recipients, to determine which recipient-donor pairs to submit to the common pool, and which not. Nevertheless, whenever we restrict ourselves to a maximum cycle length of two, we prove in Sect. 4 that the problem becomes polynomially solvable. We conclude in Sect. 5.

## 2 The problem

We consider simple directed graphs $G = (V, A)$. A *cycle* in $G$ is a set of distinct nodes $\{v_1, v_2, \ldots, v_q\}$ such that $v_i \in V$ for $i = 1, \ldots, q$, $(v_i, v_{i+1}) \in A$ for $i = 1, \ldots, q-1$, and $(v_q, v_1) \in A$; the *length* of the cycle is its number of nodes $q$. A $K$*-cycle packing* in $G$ is a set of node-disjoint cycles each of which has length at most $K$; the *size* of a $K$-cycle packing is the total number of nodes contained in its cycles. We use the phrase *cycle packing* when the length of the cycles in the cycle packing is not explicitly specified. We say a node $v \in V$ is *covered* by a cycle packing if that cycle packing contains a cycle which includes node $v$. We denote by $G[W] = (W, E \cap (W \times W))$ the subgraph of $G$ induced by the node subset $W \subseteq V$. When given subsets of nodes $U, W \subseteq V$, we use $w_K^U(G[W])$ to denote the minimum number of nodes in $U$ that are covered in a maximum size $K$-cycle packing in $G[W]$. In case $U = W$, $w_K^U(G[W])$ is just the maximum size of a $K$-cycle packing in $G[W]$. For ease of notation, we denote this by $w_K(G[W])$.

In our formulation of the problem, we distinguish between the *leader* on the one hand, and the *follower* on the other hand. Here, the leader stands for the individual agent (i.e., country or hospital), whereas the follower stands for the organisation responsible for running the larger Kidney Exchange Programme.

Initially, the leader controls a set of nodes (recall that each node refers to a recipient-donor pair), denoted by $L$, and the follower controls a set of nodes, denoted by $F$, such that $L \cap F = \emptyset$. The leader has two options for each node in $L$. One option is to withhold the node from the follower, and perhaps use the node, if possible, in a cycle packing that is *internal* to the leader, i.e., a cycle packing containing only nodes in $L$. The other option is to submit the node, thereby adding the node to the set of nodes under control of the follower, called the global pool. In the latter option it is no longer in the control of the leader which cycle packing will be selected for nodes in the global pool. However, the follower will select a maximum size cycle packing in the global pool of nodes. The question is whether the leader can guarantee that a given number of its nodes will be covered by a cycle packing. We will refer to the node set $S$ chosen by the leader to withhold from the follower, as the leader's *strategy* $S$.

**Definition 1** **(Stackelberg KEP game)** We define a **Stackelberg KEP game** as follows.

**Given:** A directed graph $G = (V = L \cup F, A)$ with $L \cap F = \emptyset$ and an integer $K$.

**Rules:** In the first phase, the leader selects a strategy $S \subseteq L$ of nodes to withhold, and calculates a maximum size $K$-cycle packing on $G[S]$. In the second phase, the follower calculates a maximum size $K$-cycle packing on $G[V \setminus S]$.

We now define the problem faced by the leader, who has to choose a strategy in the Stackelberg KEP game to maximize the number of transplants for its pool. We refer to the corresponding decision problem as **DEC-S-KEP**.

**Definition 2** **(DEC-S-KEP)** **Given:** A Stackelberg KEP game with directed graph $G = (V = L \cup F, A)$ with $L \cap F = \emptyset$ and two integers, $K$ and $t$.

**Question:** Is there a strategy $S \subseteq L$, such that $w_K(G[S]) + w_K^L(G[V \setminus S]) \geq t$?

Given the definition of $w_K^L(G)$, it follows that by considering the quantity $w_K(G[S]) + w_K^L(G[V \setminus S])$, we are considering the worst-case scenario for the leader. Indeed, this assumes that among all possible maximum size packings the follower can choose, the follower chooses the one covering the minimum number of nodes of the leader. This objective is reasonable for a risk-averse agent when the follower's tie-breaking rules are unknown.

Stackelberg KEP is closely related to, yet different from, a game considered by [14] and [13]. They consider a game with maximum cycle length $K = 2$ and $N \geq 2$ players in which each player $i$ simultaneously chooses a (restricted) set of nodes $S^i$, for $1 \leq i \leq N$. Each player $i$ then computes a maximum size cycle packing on $G[S^i]$ $(1 \leq i \leq N)$, and an independent agent (comparable to the follower in our setting) computes a maximum size cycle packing on $G[V \setminus \bigcup_{i=1}^N S^i]$. The goal for every player $i$ is to maximize $w_K(G[S^i]) + w_K^{V^i}(G[V \setminus \bigcup_{i=1}^N S^i])$, $(1 \leq i \leq N)$; this game

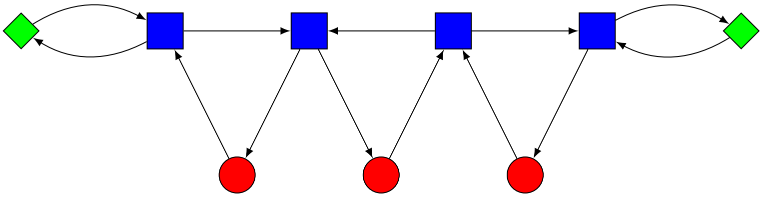


**Fig. 2** A Stackelberg KEP game with leader (red circles) and follower nodes (green diamonds, blue squares). Notice that the leader cannot internally match recipient-donor pairs and that withholding the middle red node leads to more transplants involving red recipients

is referred to as $N$-KEG. Carvalho et al. [14] give a polynomial-time algorithm for finding a socially optimal Nash equilibrium in this game if the cycle length is limited to at most two.

Apart from the fact that [13] consider an arbitrary number of players, there are two main differences between a Stackelberg KEP game and $N$-KEG. First, in a Stackelberg KEP game there are no restrictions on the strategy $S$, whereas in $N$-KEG each node in $S_i$ must be covered by a cycle packing internal to player $i$, $1 \leq i \leq N$. This is relevant as, for $K \geq 3$, it can be a dominant strategy for the leader to not submit nodes, even when such a node is not contained in a cycle packing internal to the leader. We demonstrate this phenomenon in Fig. 2. Second, in a Stackelberg KEP game, the follower is allowed to use any cycle in its solution, while in $N$-KEG the independent agent is not allowed to use cycles containing only nodes of a single player.

## 3 The Stackelberg KEP game with $K \geq 3$

In this section, we prove that DEC-S-KEP is $\Sigma_2^p$-complete. First, we prove its membership in the class $\Sigma_2^p$. Next, through a reduction from Adversarial (2,2)-SAT, we will prove that DEC-S-KEP is $\Sigma_2^p$-complete.

**Lemma 1** *DEC-S-KEP is in* $\Sigma_2^p$.

***Proof*** Recall that $\Sigma_2^p$ is defined as the set of decision problems solvable in non-deterministic polynomial-time when given access to an oracle for an NP-complete problem [3,16]; such an oracle accepts as input an instance of a decision problem in NP, and outputs the correct answer.

Thus, let us be given as a certificate a leader strategy $S \subseteq L$ for which we must check whether $w_K(G[S]) + w_K^L(G[V \setminus S]) \geq t$. Furthermore, we are given an oracle for solving the NP-complete weighted $K$-cycle packing problem [1]. We can use this oracle to check the certificate in the following way.

First, we determine $w_K(G[S])$. This is done by determining the maximum value for which the oracle returns Yes, given the graph $G[S]$. Using binary search, this requires $O(\log |V|)$ calls to the oracle. Likewise, we can determine the value $w_K(G[V \setminus S])$. To determine the minimum number of leader nodes covered in this maximum size $K$-cycle packing, we call the oracle for a weighted graph. Specifically, every follower node receives a weight of $|V|$, while every leader node receives a weight of $|V|-1$. By binary search over the range of values $[(|V|-1)w_K(G[V\setminus S]), |V|w_K(G[V\setminus S])]$, we can determine the optimal value of this weighted problem, and thus the minimum number of leader nodes in the packing, $w_K^L(G[V\setminus S])$. We can now check whether $w_K(G[S]) + w_K^L(G[V\setminus S]) \geq t$. In total, this procedure requires $O(\log|V|)$ calls to the oracle, and we can check in polynomial time whether $S$ is indeed a valid certificate for this decision problem. Hence, DEC-S-KEP is in $\Sigma_2^p$. □

Having established membership of DEC-S-KEP in $\Sigma_2^p$, we now turn to its completeness for this class. We do so through a reduction from a decision problem known as Adversarial (2,2)-SAT. Adversarial problems are problems that can be formulated using variables that are partitioned into two disjoint sets. Control of each set of variables is given to a different player. The question is whether there exist values for the first player's variables such that, for any values of the second player's variables, the resulting instance is a no-instance [16]. In this particular case, we proceed by defining (2,2)-SAT, and its corresponding adversarial problem Adversarial (2,2)-SAT.

**Definition 3** **((2,2)-SAT) Given:** A set $Y$ of variables, and a Boolean expression $E$ in conjunctive normal form, consisting of a set of clauses $C$ over $Y$. Each variable occurs exactly four times in $E$: two times in negated form and two times in unnegated form.
**Question:** Does there exist a truth assignment for $Y$ satisfying $E$?

**Definition 4** **(Adversarial (2,2)-SAT) Given:** Disjoint sets $X$ and $Y$ of variables, and a Boolean expression $E$ in conjunctive normal form, consisting of a set of clauses $C$, over $X$ and $Y$. Each variable occurs exactly four times in $E$: two times in negated form and two times in unnegated form.
**Question:** Does there exist a truth assignment for $X$ such that there does not exist a truth assignment for $Y$ satisfying $E$?

Johannes [16] shows that the existence of a reduction from 3-SAT to some non-adversarial decision problem NON-ADV that satisfies certain conditions, implies that the corresponding adversarial problem (ADV) is $\Sigma_2^p$-complete. We define 3-SAT and state the result of Johannes below.

**Definition 5** **(3-SAT) Given:** A set $Y$ of variables, and a Boolean expression $E$ in conjunctive normal form, consisting of a set of clauses $C$ over $Y$. Each clause consists of exactly 3 variables.
**Question:** Does there exist a truth assignment for $Y$ satisfying $E$?

**Theorem 1** [16] *Let (ADV) be an adversarial problem. Let (NON-ADV) denote the corresponding non-adversarial problem. We assume that (NON-ADV) is in NP. Let $f$ be a polynomial transformation from 3-SAT to (NON-ADV) that satisfies the following property. If $U$ is the set of binary variables of an instance $I_{(3SAT)}$ of 3-SAT and $Z$ is the set of binary variables of the instance $f(I_{(3SAT)})$ of (NON-ADV), then there is a subset $Z'$ of $Z$ and a bijective function $g : U \to Z'$ such that:*

1. *If $S^U$ is a satisfying solution of $I_{(3SAT)}$, then the 0-1 assignment $S_{Z'}$ to the variables in $Z'$ with $S_{Z'}(z) = S^U(g^{-1}(z))$ for all $z \in Z'$ can be extended to a 0-1 assignment $S$ of all variables in $Z$ such that $S_Z$ is a satisfying solution.*

2. *If $S_Z$ is a satisfying solution of $f(I_{(3SAT)})$, then the 0-1 assignment $S^U$ with $S^U(x) = S_Z(g(x))$ for all $x \in U$ represents a satisfying solution to $I_{(3SAT)}$.*

*Then (ADV) is $\Sigma_2^p$-complete.*

We now show that Adversarial (2,2)-SAT is $\Sigma_2^p$-complete through a reduction from 3-SAT to (2,2)-SAT that satisfies the requirements of Theorem 1.

**Theorem 2** *Adversarial (2,2)-SAT is $\Sigma_2^p$-complete.*

***Proof*** Given an instance of 3-SAT (specified by a set of variables $U = \{u_1, \ldots, u_n\}$ and a set of clauses $C$), let us now construct a (2,2)-SAT instance. Let $u_i \in U$ be a variable in the 3-SAT instance occurring $k_i$ times ($1 \leq i \leq n$). We then construct variables $z_i^1, \ldots, z_i^{k_i} \in Z$ ($1 \leq i \leq n$). We furthermore construct the clauses $\{(z_i^1, \neg z_i^2), \ldots, (z_i^{k_i-1}, \neg z_i^{k_i}), (z_i^{k_i}, \neg z_i^1)\}$ ($1 \leq i \leq n$). We refer to these as the *consistency clauses*. Note immediately that for every satisfying truth assignment in the (2,2)-SAT instance, all variables $z_i^1, \ldots, z_i^{k_i}$ must have the same truth value ($1 \leq i \leq n$). Furthermore, for each clause $c \in C$ in the 3-SAT instance, we construct a clause $c'$ in the (2,2)-SAT instance. Let the occurrence of $u_i$ in $c$ be the $j$-th occurrence (negated and unnegated combined) of $u_i$ ($1 \leq j \leq k_i$), then $z_i^j \in c'$ if $u_i$ appears unnegated or $\neg z_i^j \in c'$ if $u_i$ appears negated. Note that each variable $z_i^j \in Z$ now occurs exactly three times; once negated and once unnegated in the clauses $(z_i^j, \neg z_i^{j+1})$ and once negated OR unnegated in the other clauses ($1 \leq i \leq n$, $1 \leq j \leq k_i$). Given that we are constructing an instance of (2,2)-SAT, each variable must occur exactly twice unnegated and twice negated. We construct one more clause, the *rest clause*, which contains all these other occurrences. As we can assume that there is at least one variable in the 3-SAT instance that occurs at least once unnegated and at least once negated, this additional clause is automatically satisfied if the consistency clauses are satisfied.

Now, let $Z' = \{z_1^1, z_2^1, \ldots z_n^1\}$ and let $g$ be such that $z_i^1$ has the same truth value as $u_i$. It now easily follows that this specification of $Z'$ and $g$ satisfies the conditions from Theorem 1 [16]. We extend the truth assignment from $Z'$ to $Z$ by setting the truth assignment of $z_i^j$ equal to that of $z_i^1$ for all $i = 1, \ldots, n$ and $j = 2, \ldots, k_i$. The consistency clauses and the rest clause are then satisfied. Given the identical truth values for all $z_i^j$ for a fixed $i$, ($1 \leq i \leq n$), and because these truth values are equal to that of $u_i$, the construction of the remaining clauses also ensures these clauses are satisfied. We thus have a satisfying solution for the (2,2)-SAT instance. The other direction can be argued in a similar fashion. As this reduction satisfies the conditions from Theorem 1 [16], the proof of Theorem 2 is complete. □

We are now in a position to prove our main result.

**Theorem 3** *DEC-S-KEP is a $\Sigma_2^p$-complete problem, for $K = 3$.*

***Proof*** Lemma 1 establishes membership of DEC-S-KEP in $\Sigma_2^p$. It remains to be shown that DEC-S-KEP is $\Sigma_2^p$-hard. Given an instance $(X, Y, C, E)$ of Adversarial (2,2)-SAT, we construct an instance $G = (L \cup F, A)$ of DEC-S-KEP. To clearly distinguish

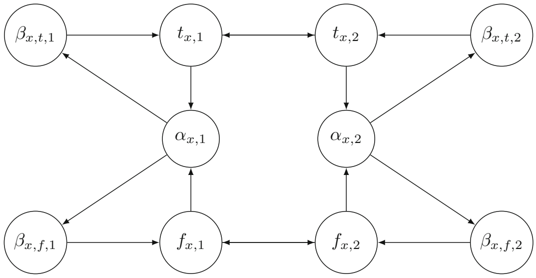



**Fig. 3** The gadget corresponding to a variable $x \in X$

between nodes belonging to the leader (i.e., the set $L$) and nodes belonging to the follower (i.e., the set $F$), we use Latin letters to denote the former, and Greek letters to denote the latter.

For each variable $x \in X$, we construct a gadget as depicted in Fig. 3. The gadget contains four leader nodes, $\{t_{x,1}, t_{x,2}, f_{x,1}, f_{x,2}\} \in L$ and six follower nodes, $\{\alpha_{x,1}, \alpha_{x,2}, \beta_{x,t,1}, \beta_{x,t,2}, \beta_{x,f,1}, \beta_{x,t,2}\} \in F$. We denote the arcs between nodes in the gadget corresponding to $x \in X$, by the set $A_x$:

$$\begin{aligned} A_x &= \{(\alpha_{x,i}, \beta_{x,t,i}), (\beta_{x,t,i}, t_{x,i}), (t_{x,i}, \alpha_{x,i}) |\ i = 1, 2\} \\ &\cup \{(\alpha_{x,i}, \beta_{x,f,i}), (\beta_{x,f,i}, f_{x,i}), (f_{x,i}, \alpha_{x,i}) |\ i = 1, 2\} \\ &\cup \{(t_{x,1}, t_{x,2}), (t_{x,2}, t_{x,1}), (f_{x,1}, f_{x,2}), (f_{x,2}, f_{x,1})\}. \end{aligned}$$

For each variable $y \in Y$, we have an identical gadget, except that all of its ten nodes are in $F$; to follow our naming convention, instead of nodes $t_{x,i}$ and $f_{x,i}$, the corresponding nodes in this gadget are called $\tau_{y,i}$ and $\phi_{y,i}$, $i = 1, 2$. The set of arcs between nodes of a gadget corresponding to $y \in Y$ is denoted by $A_y$.

For reasons of convenience, we define a set $B$ containing all so-called $\beta$ nodes as:

$$B = \{\{\beta_{x,t,i}, \beta_{x,f,i}\} |\ x \in X, i = 1, 2\} \cup \{\{\beta_{y,t,i}, \beta_{y,f,i}\} |\ y \in Y, i = 1, 2\}.$$

The construction per clause is relatively simple. For each clause $c \in C$, there exists one node, called the *clause node*, $\delta_c \in F$. Additionally, there is one node $d \in L$ in total. We have now specified the node sets $L$ and $F$; notice that $|L| = 4|X| + 1$, and $|F| = 6|X| + 10|Y| + |C|$.

For each clause $c \in C$, there exist arcs $(\delta_c, d)$, $(d, \delta_c)$ in $A$. The clause nodes are connected to the nodes in the variable gadgets as follows. For each variable $x \in X$ ($y \in Y$), there are arcs $(\beta_{x,t,1}, \delta_c)$ ($(\beta_{y,t,1}, \delta_c)$) and $(\delta_c, \beta_{x,t,1})$ ($(\delta_c, \beta_{y,t,1})$) whenever $c$ is the first clause in which $x$ ($y$) occurs unnegated, with respect to a lexicographical ordering of the clauses in the set $C$. Analogously, the node $\beta_{x,t,2}$ ($\beta_{y,t,2}$) is connected to the clause node that corresponds to the second clause in which $x$ ($y$) occurs unnegated. Similarly, the nodes $\beta_{x,f,i}$ ($\beta_{y,f,i}$), are connected to the clause node of the $i$-th clause where $x$ ($y$) occurs negated, $i = 1, 2$, through the arcs $(\beta_{x,f,i}, \delta_c)$ and $(\delta_c, \beta_{x,f,i})$ ($(\beta_{y,f,i}, \delta_c)$ and $(\delta_c, \beta_{y,f,i})$).

Summarizing the construction, we have specified the graph $G = (L \cup F, A)$ by choosing:

$$
\begin{aligned}
L &= \Big\{\{t_{x,i}, f_{x,i}\} \mid x \in X, i = 1, 2\Big\} \cup \{d\},\\
F &= \Big\{\{\alpha_{x,i}, \alpha_{y,i}\} \mid x \in X, y \in Y, i = 1, 2\Big\} \cup B \cup \Big\{\{\tau_{y,i}, \phi_{y,i}\} \mid y \in Y, i = 1, 2\Big\}\\
&\quad \cup \Big\{\delta_c \mid c \in C\Big\},\\
A &= \bigcup_{x \in X} A_x \cup \bigcup_{y \in Y} A_y \cup \{(\delta_c, d), (d, \delta_c) \mid c \in C\}\\
&\quad \cup \{(\beta_{x,t,i}, \delta_c), (\delta_c, \beta_{x,t,i}) \mid c \in C \text{ is the } i\text{-th clause containing } x,\ x \in X, i = 1, 2\}\\
&\quad \cup \{(\beta_{x,f,i}, \delta_c), (\delta_c, \beta_{x,f,i}) \mid c \in C \text{ is the } i\text{-th clause containing } \neg x,\ x \in X, i = 1, 2\}\\
&\quad \cup \{(\beta_{y,t,i}, \delta_c), (\delta_c, \beta_{y,t,i}) \mid c \in C \text{ is the } i\text{-th clause containing } y,\ y \in Y, i = 1, 2\}\\
&\quad \cup \{(\beta_{y,f,i}, \delta_c), (\delta_c, \beta_{y,f,i}) \mid c \in C \text{ is the } i\text{-th clause containing } \neg y,\ y \in Y, i = 1, 2\}.
\end{aligned}
$$

We set $K = 3$, meaning that the length of a cycle present in a solution cannot exceed 3. Finally, we set $t = 4|X| + 1$. This completes the description of an instance of DEC-S-KEP.

$\Rightarrow$ Given a truth assignment for $X$ such that no truth assignment exists for $Y$ that satisfies $E$, we now show the existence of a strategy $S \subseteq L$ such that $w_K(G[S]) + w_K^L(G[V \setminus S]) \geq 4|X| + 1 = t$.

Given such a truth assignment for $X$, we propose the following strategy $S$:

$$S = \Big\{\{t_{x,1}, t_{x,2}\} \mid x \in X \text{ is true}\Big\} \cup \Big\{\{f_{x,1}, f_{x,2}\} \mid x \in X \text{ is false}\Big\}. \tag{1}$$

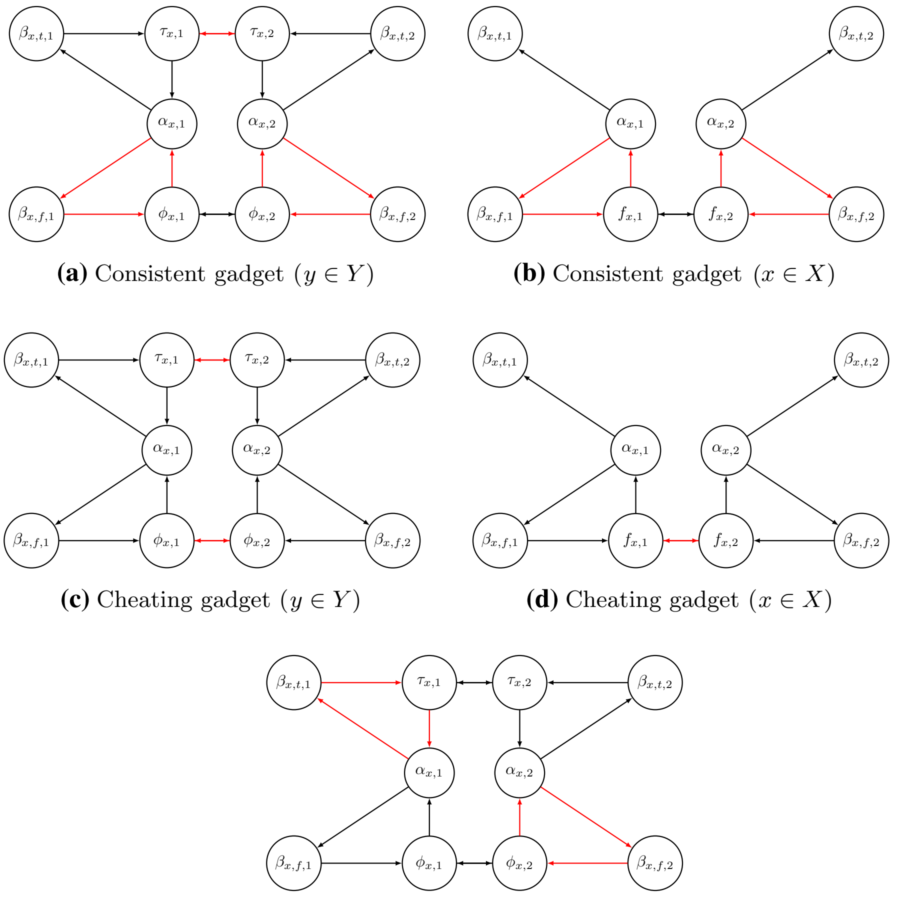
In words: for each variable $x \in X$ that is TRUE, $t_{x,i} \in S$ for $i = 1, 2$ and for each variable $x \in X$ that is FALSE, $f_{x,i} \in S$ for $i = 1, 2$.

Recall that $|L| = 4|X| + 1 = t$. Hence, we need to show that given this strategy $S$, each node of the leader is contained in a maximum size 3-cycle packing of the leader (i.e., a maximum size 3-cycle packing on $G[S]$), or in each maximum size 3-cycle packing of the follower (i.e., a maximum 3-cycle packing on $G[V \setminus S]$). By the choice of $S$, there are $2|X|$ nodes in $S$, all of which are contained in a maximum size 3-cycle packing of the leader. Indeed, such a maximum size 3-cycle packing of the leader consists of $|X|$ cycles of length 2, each containing a pair of $t$-nodes or a pair of $f$-nodes of the corresponding variable gadget. It remains to show that every maximum size 3-cycle packing for $G[V \setminus S]$ contains all $t$ and $f$-nodes not in $S$, as well as the $d$-node.

To do so, we now analyze the possible 3-cycle packings in $G[V \setminus S]$. Notice that any cycle in $G[V \setminus S]$ that contains nodes of different variable gadgets has length more than 3, and hence cannot be present in a 3-cycle packing. It follows that each cycle in the follower's 3-cycle packing

- consists of nodes all within a single variable gadget (called a cycle of Type 1), or
- consists of the nodes $\{\beta, \delta_c\}$ for some $\beta \in B, c \in C$ (called a cycle of Type 2), or
- consists of the nodes $\{\delta_c, d\}$ for some $c \in C$ (called a cycle of Type 3).

We now classify the variable gadgets with respect to the possible cycles of Type 1 contained in the variable gadget. The classification of these gadgets is illustrated in Fig. 4.



**(a)** Consistent gadget $(y \in Y)$ **(b)** Consistent gadget $(x \in X)$

**(c)** Cheating gadget $(y \in Y)$ **(d)** Cheating gadget $(x \in X)$

**(e)** Zigzag gadget $(y \in Y)$

**Fig. 4** Classification of cycle packings on variable gadgets in the follower's solution

**Definition 6** Given a solution to the follower's cycle packing problem, we call a gadget corresponding to variable $x \in X$

- *consistent* if either the two node-sets $\{\alpha_{x,i}, \beta_{x,f,i}, f_{x,i}\}$, $i = 1, 2$, or the two node-sets $\{\alpha_{x,i}, \beta_{x,t,i}, t_{x,i}\}$, $i = 1, 2$ each correspond to a cycle in the solution (Fig. 4b),
- *cheating* if the node-set $\{f_{x,1}, f_{x,2}\}$ or $\{t_{x,1}, t_{x,2}\}$ corresponds to a cycle in the solution (Fig. 4d).

Given a solution to the follower's cycle packing problem, we call a gadget corresponding to variable $y \in Y$

- *consistent* if either the node-sets $\{\alpha_{y,i}, \beta_{y,f,i}, \phi_{y,i}\}$, $i = 1, 2$ as well as the node-set $\{\tau_{y,1}, \tau_{y,2}\}$ each correspond to a cycle in the solution, or if the node-sets

$\{\alpha_{y,i}, \beta_{y,t,i}, \tau_{y,i}\}$, $i = 1, 2$ as well as the node-set $\{\phi_{y,1}, \phi_{y,2}\}$ each correspond to a cycle in the solution (Fig. 4a),

- *cheating* if the two node-sets $\{\tau_{y,1}, \tau_{y,2}\}$ and $\{\phi_{y,1}, \phi_{y,2}\}$ each correspond to a cycle in the solution (Fig. 4c),
- *zigzag* if the node-sets $\{\alpha_{y,1}, \beta_{y,t,1}, \tau_{y,1}\}$ and $\{\alpha_{y,2}, \beta_{y,f,2}, \phi_{y,2}\}$ each correspond to a cycle in the solution (Fig. 4e), or alternatively, the node-sets $\{\alpha_{y,2}, \beta_{y,t,2}, \tau_{y,2}\}$ and $\{\alpha_{y,1}, \beta_{y,f,1}, \phi_{y,1}\}$.

Note that these classifications concern only cycles of Type 1. Cycles of Type 2 can be added, covering any of the $\beta$ nodes in the gadget not yet covered by a cycle of Type 1, without affecting the packing classification.

Consistent gadgets will be used to reflect the truth value of the corresponding variables. We say the gadget is consistent with a TRUE value if the two node-sets $\{\alpha_{x,i}, \beta_{x,f,i}, f_{x,i}\}$ for $i = 1, 2$ each correspond to a cycle in the follower's cycle packing solution. Analogously, if in the follower's solution the node-sets $\{\alpha_{x,i}, \beta_{x,t,i}, t_{x,i}\}$ for $i = 1, 2$, each correspond to a cycle, we say the gadget is consistent with FALSE.

We use these definitions to characterize an optimal 3-cycle packing of the follower, as witnessed by the following lemma.

**Lemma 2** *Let $S$ be defined by (1). In each optimal 3-cycle packing on $G[V \setminus S]$, each variable gadget is either consistent, cheating or zigzag.*

***Proof*** We argue as follows. Consider a feasible solution to the follower's cycle packing problem such that there is a variable gadget which is neither consistent, nor cheating, nor zigzag. We show that that solution is not of maximum size by exhibiting the existence of a strictly better solution.

We first consider variable gadgets corresponding to variables $x \in X$. Without loss of generality, we assume that $t_{x,1}, t_{x,2} \in S$, as depicted in Fig. 4b and d. There are two cases.

Case 1: Consider a solution such that for some variable gadget corresponding to $x \in X$, at most one of $\{\beta_{x,f,1}, \beta_{x,f,2}\}$ is covered by a cycle of Type 2. Since, by assumption, the gadget is not consistent, at most three of its nodes are covered by a cycle of Type 1. By removing at most one cycle of Type 2 (thereby "freeing" a $\beta$ node), and introducing a cycle of Type 1, we have changed the state of this gadget to consistent. Moreover, the size of the packing has increased by at least $-2 + 3 = 1$.

Case 2: Consider a solution such that for some variable gadget corresponding to $x \in X$, both $\beta_{x,f,i}$, $i = 1, 2$ are covered by cycles of Type 2. Since, by assumption, the gadget is not cheating, it follows that no other nodes of the gadget are contained in a cycle. By simply adding the cycle consisting of the nodes $\{f_{x,1}, f_{x,2}\}$, the size of the packing increases by 2.

Now we consider variable gadgets corresponding to variables $y \in Y$. For ease of exposition, we assume that for such a gadget in the current solution, at least as many $\beta_{y,t,i}$ as $\beta_{y,f,i}$ $(i = 1, 2)$ are covered by cycles of Type 2. The same arguments hold, mutatis mutandis, if more $\beta_{y,f,i}$ than $\beta_{y,t,i}$ nodes are covered by Type 2 cycles. We consider the four $\beta$ nodes of the gadget, and make a case distinction based on whether these $\beta$ nodes are covered by cycles of Type 2.

Case 1: All four $\beta$ nodes of the gadget are covered by cycles of Type 2. Since, by assumption, the gadget is not cheating, one easily verifies that the size of the packing improves when the solution is changed such that the gadget becomes a cheating gadget.

Case 2: Three of the four $\beta$ nodes of the gadget are covered by cycles of Type 2. Hence, a single $\beta$ node is not covered by a cycle of Type 2, say $\beta_{y,f,2}$. It also follows that $\alpha_{y,1}$ is not covered. If $\phi_{y,1}$ is not covered, we remove the cycle of Type 2 covering $\beta_{y,f,1}$, and add the cycle of Type 1 containing nodes $\beta_{y,f,1}$, $\phi_{y,1}$ and $\alpha_{y,1}$, thereby increasing the size of the packing. If $\phi_{y,1}$ is covered, it is in a cycle with $\phi_{y,2}$, meaning that $\alpha_{y,2}$ is not covered. Then, we remove the cycle of Type 1 covering $\phi_{y,1}$ and $\phi_{y,2}$, as well as the cycle of Type 2 covering $\beta_{y,f,1}$, and we add the two cycles of Type 1 containing nodes $\beta_{y,f,i}$, $\phi_{y,i}$ and $\alpha_{y,i}$, $i = 1, 2$, again increasing the size of the packing.

Case 3: Two of the four $\beta$ nodes are covered by cycles of Type 2. If these two nodes are $\beta_{y,t,i}$, $i = 1, 2$, then it is easy to see that modifying the solution such that the gadget becomes consistent (in fact, consistent with a TRUE value) is the only possibility since all nodes of the gadget are then covered. If $\beta_{y,t,1}$ and $\beta_{y,f,1}$ are covered by cycles of Type 2, it follows that at most 5 nodes of the gadget can be covered by cycles of Type 1. By removing one cycle of Type 2, this solution can be improved such that the gadget becomes consistent. The number of nodes covered by cycles of Type 1 rises to 8 and one of the cycles of Type 2 covering $\beta_{y,t,1}$, $\beta_{y,f,1}$ can still be used. If $\beta_{y,t,1}$ and $\beta_{y,f,2}$ are covered by cycles of Type 2, and since, by assumption, the gadget is not zigzag, at most 5 nodes of the gadget are covered by cycles of Type 1. By switching to a zigzag packing, 6 nodes are covered by cycles of Type 1, and no existing cycles of Type 2 are impacted. The size of the packing increases.

Case 4: At most one $\beta$ node is covered by a cycle of Type 2. Since, by assumption, the gadget is not consistent, we can improve the solution by modifying this gadget to be consistent.

As we have covered all cases, the proof of Lemma 2 is now complete. □

Thus, we have proven that in any optimum 3-cycle packing of the follower, each gadget is either consistent, cheating or zigzag. Given this structure of any optimal solution of the follower, we will now argue that the leader's strategy $S$ ensures that all $4|X| + 1$ leader nodes will be covered (if the truth assignment on $X$ is such that there exists no truth assignment on $Y$ satisfying $E$). Observe that, given the two possible packings on variable gadgets corresponding to variables $x \in X$, all leader nodes within the variable gadgets are covered. Thus, we have covered already $4|X|$ nodes of the leader. We must now show that the $d$ node will also be covered in any maximum size cycle packing of the follower.

By Lemma 2, we know that an optimal solution of the follower either contains a variable gadget that is cheating or zigzag, or all variable gadgets are consistent. We proceed to argue that in each of these two cases, any optimal solution of the follower covers node $d$.

**Lemma 3** *Let $S$ be defined by (1). In each optimal 3-cycle packing on $G[V \setminus S]$, node $d$ is covered.*

***Proof*** We distinguish two cases. In both cases, we argue by contradiction, i.e., we argue that if node $d$ is not covered in a 3-cycle packing, that 3-cycle packing is not of maximum size.

Case 1: The follower's solution uses a gadget that is either cheating or zigzag. Let us further suppose that this follower's solution is such that node $d$ is not covered.

- Cheating gadget: the cheating gadget covers 4 nodes by cycles of Type 1 (2 if the gadget corresponds to $x \in X$), compared to 8 (6) nodes in a consistent gadget. Switching to a consistent packing is thus strictly better unless we need to break two cycles of Type 2 to perform this switch. This can only be the case if all four $\beta$-nodes are covered by cycles of Type 2 in the cheating gadget. Indeed, if less than four are covered, the packing can be made consistent by breaking at most one cycle of Type 2. However, even if all four $\beta$-nodes are covered by cycles of Type 2, this still only leads to parity between the consistent and cheating packing (since they both cover 12 (10) nodes over the variable and linked clause gadgets combined). Thus, if the $d$ node is uncovered, one of the cycles of Type 2 can be replaced by the cycle $(\delta_c, d)$ of Type 3, and the consistent packing achieves 14 (12) nodes covered over the variable and linked clause gadgets.
- Zigzag gadget: by an analogous argument, it can be shown that if $d$ is uncovered, switching from a zigzag to a consistent packing increases the number of covered nodes by two.

Case 2: The follower's solution uses only consistent gadgets. Let us further suppose that this follower's solution is such that node $d$ is not covered. Recall that the truth assignment on $X$ used to build the strategy $S$ is such that there does not exist any truth assignment on $Y$ satisfying $E$.

Since node $d$ is not covered, it follows immediately that in this cycle packing, each $\delta_c$ is covered by a cycle of Type 2. If this were not the case, the cycle $(\delta_c, d)$ would strictly increase the size of the packing. Now let us build a truth assignment for the (2,2)-SAT instance based on the packing. If the packing restricted to a variable gadget is consistent with TRUE (FALSE), set that variable to TRUE (FALSE). Clearly, for each variable $x \in X$ this truth assignment is the same as the original truth assignment used to construct the strategy $S$. We claim that this truth assignment satisfies each clause. Indeed, for a given clause $c$ let there be (wlog) the cycle $(\delta_c, \beta_{y,t,i})$. By construction of the DEC-S-KEP instance these arcs only exist if the clause is satisfied by a value of TRUE for $y$. Furthermore, by construction of the truth assignment, we have set $y$ to TRUE. Thus, the truth assignment is such that every clause is satisfied. Thus, we have arrived at a contradiction.

Since Lemma 2 implies there are no other cases, the proof of Lemma 3 is now complete. □

Summarizing, if there exists a truth assignment to $X$ such that there does not exist a truth assignment to $Y$ satisfying $E$, the leader can construct a strategy $S$. This strategy is such that if the follower's solution uses only consistent gadgets, there is at least one $\delta_c$-node that cannot be covered by a cycle of Type 2, and will thus be covered by a cycle with node $d$, i.e., a cycle of Type 3. Alternatively, if the follower's solution contains a cheating or a zigzag gadget then it must also be the case that node $d$ is covered. The leader is thus guaranteed that the strategy $S$ implies that all its $4|X|+1$ nodes are covered.

$\Leftarrow$ Suppose that there exists a strategy $S$ such that the leader can guarantee that all its $4|X|+1$ nodes are covered. We will show that the existence of such a strategy implies that there exists a truth assignment for $X$ such that there is no truth assignment for $Y$ satisfying the expression $E$.

We first analyze the structure of any strategy $S$ that guarantees covering all $4|X|+1$ nodes of the leader; we use $\mathcal{S}$ to denote the collection of strategies that guarantee that all nodes of the leader are covered, i.e., the set of optimal strategies.

Observe that for each $S \in \mathcal{S}$ it must hold that either both $t_{x,1}$ and $t_{x,2}$, or none of $t_{x,1}$ and $t_{x,2}$ are in $S$ for each $x \in X$. Indeed, if $S$ contains exactly one node from $\{t_{x,1}, t_{x,2}\}$ for some gadget corresponding to $x \in X$, it is impossible to cover that node of the leader. Thus, $\mathcal{S}$ contains only strategies $S$ for which either both $t_{x,1}$ and $t_{x,2}$, or none of $t_{x,1}$ and $t_{x,2}$ are in $S$ for each $x \in X$. The same statement holds for the nodes $f_{x,1}$ and $f_{x,2}$, for some $x \in X$: either both nodes $f_{x,1}, f_{x,2}$ are in $S$ or none of them, for each $x \in X$.

Further, we call a strategy $S$ *nice* if, for each $x \in X$, either $t_{x,1}, t_{x,2} \in S$, or $f_{x,1}, f_{x,2} \in S$ but not both.

The following lemma describes the presence of this property in optimal strategies.

**Lemma 4** *There exists an optimal strategy $S \in \mathcal{S}$ that is nice.*

***Proof*** First, remark that we can generalize the statement of Lemma 2 for all nice strategies $S$, since the proof does not use any particular property of (1) except that it is nice. Therefore, it follows that any nice strategy guarantees that $4|X|$ leader nodes are covered, either in the leader's or the follower's cycle packing; an optimal 3-cycle packing on $G[V \setminus S]$ guarantees a consistent or cheating packing on every $X$-gadget, covering all $4|X|$ leader nodes in these gadgets. This also immediately implies that, in case a nice strategy $S$ does not guarantee $4|X|+1$ covered leader nodes, there must exist an optimal 3-cycle packing on $G[V \setminus S]$ that does not cover the final leader node $d$.

We now argue by contradiction. Assume, that each $S \in \mathcal{S}$ is not nice. Since each nice strategy guarantees $4|X|$ covered leader nodes, the strategies $S \in \mathcal{S}$ should guarantee $4|X|+1$ covered leader nodes. We will show this cannot be the case.

Let $S \in \mathcal{S}$ be a strategy that is not nice. It follows that there exists a set of gadgets corresponding to $W = W_1 \cup W_2 \subseteq X$ such that

1. $W_1 := \{x \in X \mid t_{x,1}, t_{x,2}, f_{x,1}, f_{x,2} \notin S\}$, and
2. $W_2 := \{x \in X \mid t_{x,1}, t_{x,2}, f_{x,1}, f_{x,2} \in S\}$.

Let strategy $S'$ be identical to $S$ except that, for each $x \in W$, we set $t_{x,1}, t_{x,2} \in S'$ and $f_{x,1}, f_{x,2} \notin S'$. Clearly, it follows that $S'$ is nice. The strategy $S'$ being nice, and the cycle packing on $G[V \setminus S']$ being of maximum size while not covering $d$, imply the following:

1. Each clause node $\delta_c$ ($c \in C$) is in a cycle of Type 2 with a $\beta$-node, and
2. each gadget is consistent; this follows from Lemma 2, and the proof of Lemma 3, case 1, as the arguments used to prove both apply to any nice strategy.

Given the cycle packing on $G[V \setminus S']$, we now construct a maximum size cycle packing on $G[V \setminus S]$ which also does not cover $d$. First, these packings are identical with respect to the cycles of Type 2, and the cycles of Type 1 in the gadgets corresponding to $y \in Y$ and $x \in X \setminus W$. Note that these gadgets are all consistent. Next, for gadgets corresponding to a variable $x \in W_1$, choose cycles such that the gadget is consistent with TRUE. For the gadgets of variables $x \in W_2$, the follower cannot choose any cycle of Type 1 in the gadget.

The size of the cycle packing on $G[V \setminus S]$ as just described is $2|C|+8|Y|+8|W_1|+6|X \setminus W|$. This is also an upper bound on the size of any maximum cycle packing on $G[V \setminus S]$. Indeed, since each cycle of Type 2 or Type 3 has a length of 2 and covers one node $\delta_c$, $c \in C$, their combined size is at most $2|C|$. The size of cycles of Type 1 per gadget is similarly bounded by 8 for gadgets corresponding to $y \in Y$ and $x \in W_1$, 6 for gadgets corresponding to $x \in X \setminus W$, and no cycles can be chosen in gadgets corresponding to $x \in W_2$. Since the size of the constructed cycle packing on $G[V \setminus S]$ matches the upper bound, it is a maximum size cycle packing. Note that this cycle packing does not cover the node $d$. Since there is a maximum size cycle packing not covering $d$, the strategy $S$ does not guarantee $4|X|+1$ covered leader nodes and either we have $S \notin \mathcal{S}$, or an optimal strategy can at most guarantee $4|X|$ covered leader nodes and each nice strategy $S' \in \mathcal{S}$. Either way, we obtain a contradiction, and the proof of Lemma 4 is complete. □

By Lemma 4, we are ensured that there is a nice strategy $S$ within the class $\mathcal{S}$. Given a nice strategy $S$, we formulate the solution to Adversarial (2,2)-SAT accordingly: if $t_{x,1}, t_{x,2} \in S$ and $f_{x,1}, f_{x,2} \notin S$, set $x \in X$ to TRUE, and conversely, if $t_{x,1}, t_{x,2} \notin S$ and $f_{x,1}, f_{x,2} \in S$, set $x \in X$ to FALSE. We claim that this truth assignment for $X$ is such that there does not exist a truth assignment for $Y$ satisfying $E$. We will now argue by contradiction, and show that if there exists a truth assignment for $Y$ satisfying $E$, then the strategy $S$ only guarantees $4|X|$ covered leader nodes.

Given a truth assignment for $Y$ satisfying $E$, we construct a maximum size cycle packing only covering $4|X|$ leader nodes as follows. In each variable gadget, select a packing consistent with the truth value of the corresponding variable. This adds $8|Y| + 6|X|$ to the size of the cycle packing. For each clause, select a cycle of Type 2. This requires that for each clause, there is at least one uncovered $\beta$ node connected through a 2-cycle. This is guaranteed in the following way. Let $x$ be a variable satisfying clause $c$ in the truth assignment. Without loss of generality, we assume it occurs in $c$ unnegated, and that it is the first unnegated occurrence. Then we have chosen a packing consistent with TRUE for the variable gadget of $x$, and $\beta_{x,t,1}$ remains uncovered. The cycle between $\delta_c$ and $\beta_{x,t,1}$ can be added to the cycle packing. In this way, we add $2|C|$ to the size of the packing. Note that $d$ is left uncovered in this cycle packing. The cycle packing is also of maximum size, $8|Y| + 6|X| + 2|C|$, since it reaches the upper bound on packing size. This can be easily checked, since the upper bound of the cycle packing restricted to cycles of Type 1 is $8|Y| + 6|X|$, i.e. consistent packings in every gadget. The upper bound of the packing restricted to cycles of Type 2 and 3 is

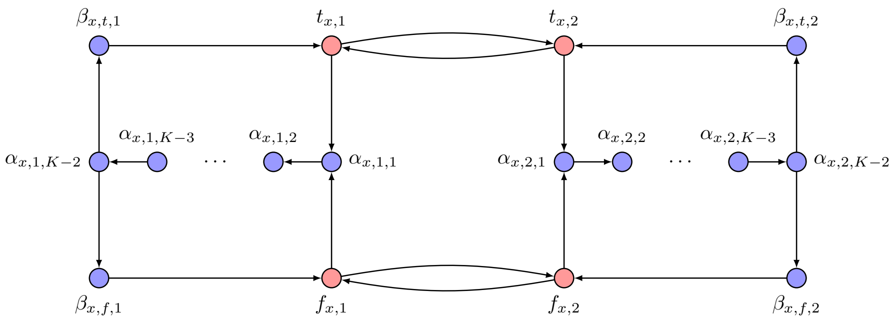



**Fig. 5** The modified gadget corresponding to a variable $x \in X$ for some fixed $K \geq 3$

$2|C|$, since each cycle of these types is of length 2 and requires a node $\delta_c$, of which there are exactly $|C|$. The existence of a truth assignment for $Y$ satisfying $E$ thus means the strategy $S$ did not guarantee $4|X|+1$ covered leader nodes, a contradiction. We conclude that a truth assignment on $Y$ satisfying $E$ does not exist, i.e., that the existence of a strategy $S$ guaranteeing $4|X|+1$ covered leader nodes guarantees the existence of a solution $X$ to Adversarial (2,2)-SAT. $\square$

The result of Theorem 3 can actually be generalized through a slight modification of the variable gadgets.

**Corollary 1** *DEC-S-KEP is $\Sigma_2^P$-complete for each fixed $K \geq 3$.*

***Proof*** Fix some maximum cycle length $K \geq 3$ and a variable $x \in X \cup Y$. We modify the variable gadget as follows: for $i = 1, 2$, we replace the node $\alpha_{x,i}$ by $K-2$ nodes $\alpha_{x,i,j}$ with $j = 1, \ldots, K-2$. All arcs with endpoint $\alpha_{x,i}$ in the case of $K = 3$ now have endpoint $\alpha_{x,i,1}$. All arcs with origin $\alpha_{x,i}$ in the case of $K = 3$ now have origin $\alpha_{x,i,K-2}$. Additionally, in case $K \geq 4$, there is a path of length $K-3$ consisting of arcs $(\alpha_{x,i,j}, \alpha_{x,i,j+1})$ for $j = 1, \ldots, K-3$. Figure 5 illustrates the modified variable gadget. Notice that for $K = 3$ we indeed return to our original variable gadget $G_x$. The proof of the corollary for arbitrary fixed $K \geq 3$, with the same target $t = 4|X|+1$, is almost completely analogous to the proof of Theorem 3. The path between $\alpha_{x,i,1}$ and $\alpha_{x,i,K-2}$ ensures that each cycle satisfying the maximum cycle length is either of Type 1, Type 2 or Type 3. The proof thus goes through unchanged, except for presence of cycles of length $K$ within the variable gadgets. In case $K \geq 4$, cheating packings are now strictly dominated by consistent packings, hence, in this case, the proof of Lemma 3 reduces to a smaller number of cases. $\square$

## 4 The Stackelberg KEP game with $K = 2$

In this section, we consider the Stackelberg KEP game for $K = 2$. We will show that DEC-S-KEP for $K = 2$ is polynomially solvable, i.e., we can compute an optimal strategy $S$ for the leader in polynomial time.

The proof of this claim heavily relies on the results by [14] and [13]. We will show that the leader's optimal strategy can be determined by solving the problem of computing a player's best reaction in an $N$-KEG game, whenever the strategies of the other $N-1$ players are considered fixed. Note that in an $N$-KEG game, the players are restricted to play a strategy in which they contribute all internally unmatched pairs to the common pool. This stands in contrast to the setting of the Stackelberg KEP game, where the leader is allowed to withhold unmatched nodes from the follower.

In the following lemma, we show that contributing an extra node to the common pool never decreases the minimum number of leader nodes matched in a maximum size matching. As a result, strategies where the leader does not contribute one or more nodes not covered by the internal packing are (weakly) dominated by the strategy with an identical internal packing where the leader contributes all nodes that are not covered. Thus, there always exists an optimal strategy where the leader contributes all nodes that are not covered to the common pool.

For ease of notation, we reduce the directed compatibility graph to an undirected graph $G = (V, E)$, where $E$ consists of edges $\{u, v\}$ for which the arc set of the directed counterpart contains both $(u, v)$ and $(v, u)$.

**Lemma 5** *Let $G = (V = L \cup F, E)$ be an undirected graph, $S \subseteq L$ a strategy of the leader and $u \in S$ a node. Then:*

$$w_K^L(G[V \setminus S]) \leq w_K^L(G[(V \setminus S) \cup \{u\}]).$$

***Proof*** Clearly, as $w_K(G[V \setminus S]) \leq w_K(G[(V \setminus S) \cup \{u\}]) \leq w_K(G[V \setminus S]) + 1$, we can restrict ourselves to a case distinction with two cases:

Case 1: $w_K(G[(V \setminus S) \cup \{u\})]) = w_K(G[V \setminus S]) + 1$. In this case, any maximum matching of $G[(V \setminus S) \cup \{u\}]$ matches $u$. Take an arbitrary maximum matching $M_u$ of $G[(V \setminus S) \cup \{u\}]$, let $e = \{u, v\} \in M_u$ for some $v \in V \setminus S$ be the edge that matches $u$. The matching $M_u \setminus e$ is a maximum matching of $G[V \setminus S]$ with fewer leader nodes (as $u$ is a leader node). Thus, any maximum matching of $G[(V \setminus S) \cup \{u\}]$ corresponds to a maximum matching of $G[V \setminus S]$ covering fewer leader nodes. In particular, this implies $w_K^L(G[V \setminus S]) < w_K^L(G[(V \setminus S) \cup \{u\}])$, meaning that we actually increase our objective value by submitting $u$.

Case 2: $w_K(G[(V \setminus S) \cup \{u\}]) = w_K(G[V \setminus S])$. Clearly, we have that $w_K^L(G[V \setminus S]) \geq w_K^L(G[(V \setminus S) \cup \{u\}])$: any maximum matching of $G[V \setminus S]$ is also a feasible maximum matching for $G[(V \setminus S) \cup \{u\}]$. We will now show by contradiction that also $w_K^L(G[V \setminus S]) \leq w_K^L(G[(V \setminus S) \cup \{u\}])$ must hold.

Suppose that

$$w_K^L(G[(V \setminus S) \cup \{u\}]) < w_K^L(G[V \setminus S]). \tag{2}$$

In that case, let us consider a maximum matching $M_u$ of $G[(V \setminus S) \cup \{u\}]$ covering $w_K^L(G[(V \setminus S) \cup \{u\}])$ leader nodes. Then $u$ is matched in $M_u$, as otherwise $M_u$ would be a maximum matching on $G[V \setminus S]$ with fewer than $w_K^L(G[V \setminus S])$ covered leader nodes. Let $e = \{u, v\} \in M$ for some $v \in V \setminus S$. The matching $M' = M_u \setminus e$ is a $(w_K(G[V \setminus S]) - 1)$-cardinality matching in $G[V \setminus S]$, thus non-maximum. This implies that there exists an $M'$-augmenting path in $G[V \setminus S]$ due to [6].

We claim that any $M'$-augmenting path starts in $v$; if not, let $P$ be an $M'$-augmenting path in $G[V \setminus S]$ not starting in $v$. Then: $M' \oplus P$ is a $w_K(G[V \setminus S])$-cardinality matching in $G$, where $A \oplus B := (A \setminus B) \cup (B \setminus A)$ denotes the symmetric difference of $A$ and $B$. Then, the nodes $u$ and $v$ are still both unmatched, as we assumed $P$ does not contain $v$. This implies that $\big(M' \oplus P\big) \cup \{\{u, v\}\}$ is a $(w_K(G[V \setminus S]) + 1)$-cardinality matching in $G[(V \setminus S) \cup \{u\}]$. This contradicts our assumption that adding $u$ to the graph does not increase the size of a maximum matching.

Therefore, any $M'$-augmenting path $P = \{v = v_0, v_1, \dots, v_k = w\}$ in $G[V \setminus S]$ must start in $v$. The set of matched nodes in the $w_K(G[V \setminus S])$-cardinality matching $M' \oplus P$ is almost the same as the set of matched nodes in $M_u$, except $u$ in $M_u$ is exchanged for node $w$ (the endnode of $P$ unequal to $v$). This means $M' \oplus P$ is a maximum matching on $G[V \setminus S]$ and covers either $w_K^L(G[(V \setminus S) \cup \{u\}])$ (when $w \in L$) or $w_K^L(G[(V \setminus S) \cup \{u\}]) - 1$ (when $w \in F$) leader nodes, which gives a contradiction with the assumption made in Inequality (2).

Thus, in both cases, we obtain $w_K^L(G[V \setminus S]) \leq w_K^L(G[(V \setminus S) \cup \{u\}])$, which finishes the proof. □

Lemma 5 shows that whenever strategy $S \subseteq L$ is chosen and $u \in S$ is unmatched with respect to a maximum matching on $G[S]$, there is no incentive for the leader to hide node $u$ from the follower. Therefore, the leader can restrict itself to strategies $S \subseteq L$ for which $G[S]$ allows a perfect matching.

Furthermore, we notice that in contrast to the setting of the $N$-KEG problem in [13] where the independent agent is not allowed to use edges between nodes of the same player (internal edges) the Stackelberg KEP game does not have this restriction. Once again, we claim that for any strategy $S' \subseteq L$ for which the follower will choose an internal edge $\{u, v\} \subseteq L$ in the second phase of the Stackelberg KEP game, there exists a weakly dominating strategy $S \subseteq L$ for which the follower will not pick internal leader edges on the maximum size matching on $G[V \setminus S]$.

**Lemma 6** *Let $G = (V = L \cup F, E)$ be an undirected graph. There exists an optimal strategy $S \subseteq L$ such that the follower chooses a maximum size matching on $G[V \setminus S]$ with no internal leader edges.*

***Proof*** Let $S' \subseteq L$ be an arbitrary feasible leader strategy. Let $M$ be a maximum size matching on $G[V \setminus S']$ covering exactly $w_K^L(G[V \setminus S'])$ leader nodes. Let $N = M \cap E(G[L])$ be the submatching of $M$ consisting of the internal leader edges. Consider now the feasible strategy $S = S' \cup V(N)$. Strategy $S$ has the same guaranteed objective value as $S'$, but now the follower does not pick any internal leader edges anymore. This shows that there exists an optimal strategy for the leader in which the follower will not pick any internal leader edges. □

Notice that whenever we impose the hard constraint that the follower is not allowed to use internal leader edges, the minimum number of covered leader nodes in a maximum size follower matching can never increase. Therefore, it also follows that the strategic options of the follower are equivalent to those of an independent agent in a suitably constructed $N$-KEG game. Together with this observation, we now have all the necessary tools to derive the complexity of DEC-S-KEP restricted to pairwise exchanges only.

**Theorem 4** *DEC-S-KEP is polynomially solvable if the maximum cycle length $K = 2$.*

***Proof*** We consider a Stackelberg KEP game on the graph $G = (V = L \cup F, A)$ with maximum cycle length $K = 2$. We now construct a $(|F| + 1)$-KEG with maximum cycle length $K = 2$ as follows: let player 1 have the control over the leader node set $L$, while the remaining $|F|$ players each have control over a unique node from $F$. We show that finding an optimal strategy in DEC-S-KEP corresponds to player 1's strategy in an arbitrary Nash equilibrium on $(|F| + 1)$-KEG.

Notice that the $|F|$ players with only one node in the $(|F| + 1)$-KEG only have one feasible strategy, namely to contribute their only node. Therefore, finding a Nash equilibrium in $(|F| + 1)$-KEG reduces to finding a best response for player 1 in $(|F| + 1)$-KEG. As a result, each equilibrium yields the same number of transplants involving donor-recipient pairs of player 1.

Let us compute an arbitrary Nash equilibrium of $(|F| + 1)$-KEG. [13] show that such an equilibrium exists, and can be computed in polynomial time. We show that the strategy of player 1 in this equilibrium is an optimal strategy for DEC-S-KEP. In DEC-S-KEP, the leader can deploy a strategy $S \subseteq L$ where internal nodes are hidden and unmatched, and the independent agent is allowed to use all edges in $G[V \backslash S]$ for its global cycle packing. However, it follows from Lemma 5 and Lemma 6 that for DEC-S-KEP with $K = 2$, there exists an optimal leader strategy $S \subseteq L$ such that $G[S]$ allows a perfect matching, i.e. no nodes are kept internally while being unmatched, and such that the follower can choose a maximum matching in $G[V \backslash S]$ covering the fewest nodes from $L$ while using only edges between nodes from different players. The strategy of player 1 in the Nash equilibrium on $(|F|+1)$-KEG is therefore also an optimal strategy for DEC-S-KEP. Thus, the optimal strategy for the leader in DEC-S-KEP with $K = 2$ can be computed in polynomial time by using the results from [13]. □

## 5 Conclusion

Collaboration between countries, as well as cooperation among hospitals (both referred to as agents), has the potential to improve the lives of patients in need of a kidney transplant. Indeed, both the number of realized transplants and the quality of the matches found, can be increased. However, when joining a collaborative effort, an agent, in control of a number of recipient-donor pairs, faces the choice which pairs to contribute, and which pairs to withhold. Different properties (strategy-proofness, rationality) of various mechanisms have been investigated in the literature; here, we have shown that for an individual agent the problem can be modelled as a Stackelberg KEP game. We proved that in case the maximum cycle length is bounded by $K \geq 3$, the corresponding problem is $\Sigma_2^p$-complete, while for $K = 2$ the problem is solvable in polynomial time. Thus, we have shown that, in general, for an individual agent, the problem of maximizing the number of their own transplants can be very complex computationally. This result holds even in the absence of relevant real-life issues in KEPs, such as non-directed donor chains (taking into account the presence of altruistic donors), the lack of information on the compatibilities present in the pool, and strategies of other agents. Analyzing the implications of these issues in Stackelberg KEP games is interesting. Our result has the potential to weaken the need for deploying mechanisms that are strategy-proof, as individual agents will find it challenging to identify their own optimal strategy if $K \geq 3$. So far, in our setting with a single leader and a single follower, the case of $K = 2$ is easy, implying that a player can efficiently choose an optimal strategy if the strategies of all other players are fixed. However, interesting questions for multi-player settings remain. For instance, when the strategies of other agents are unknown (and may depend on the first agent's strategy), can the first agent identify a strategy yielding the first agent a given number of transplants?

**Supplementary Information** The online version contains supplementary material available at https://doi.org/10.1007/s10107-021-01748-6.

## References


1. Abraham, David J., Blum, Avrim, Sandholm, Tuomas: Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *Proceedings of the eighth ACM Conference on Economic Commerce*, pages 295–304. ACM, (2007)
2. Agarwal, N., Ashlagi, I., Azevedo, E., Featherstone, C.R., Karaduman, Ö.: Market failure in kidney exchange. Am. Econ. Rev. **109**(11), 4026–4070 (2020)
3. Arora, S., Barak, B.: Computational complexity: a modern approach. Technical report, Cambridge University Press (2009)
4. Ashlagi, I., Roth, A.E.: Free riding and participation in large scale, multi-hospital kidney exchange. Theor. Econ. **9**(3), 817–863 (2014)
5. Ashlagi, I., Fischer, F., Kash, I.A., Procaccia, A.D.: Mix and match: a strategyproof mechanism for multi-hospital kidney exchange. Games Econom. Behav. **91**, 284–296 (2015)
6. Berge, C.: Two theorems in graph theory. Proc. Natl. Acad. Sci. USA **43**(9), 842–844 (1957)
7. Biró, P., Haase-Kromwijk, B., Andersson, T., Ásgeirsson, E.I., Baltesová, T., Boletis, I., Bolotinha, C., Bond, G., Böhmig, G., Burnapp, L., Cechlárová, K., Di Ciaccio, P., Fronek, J., Hadaya, K., Hemke, A., Jacquelinet, C., Johnson, R., Kieszek, R., Kuypers, D., Leishman, R., Macher, M.-A., Manlove, D., Menoudakou, G., Salonen, M., Smeulders, B., Sparacino, V., Spieksma, F., de la Oliva Valentín Muñoz, M., Wilson, N., van de Klundert, J.: Building kidney exchange programmes in Europe – an overview of exchange practice and activities. Transplantation, 103:1514–1522 (2019)
8. Biró, P., Kern, W., Pálvölgyi, D., Paulusma, D.: Generalized matching games for international kidney exchange. In Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems, pp. 413–421. International Foundation for Autonomous Agents and Multiagent Systems (2019)
9. Biró, P., Gyetvai, M., Klimentova, X., Pedroso, J.P., Pettersson, W., Viana: Compensation scheme with Shapley value for multi-country kidney exchange programmes, Ana (2020)
10. Modelling and optimisation in european kidney exchange programmes: Biró, Péter., van de Klundert, Joris, Manlove, David, Pettersson, William, Andersson, Tommy, Burnapp, Lisa, Chromy, Pavel, Delgado, Pablo, Dworczak, Piotr, Haase, Bernadette, et al. Eur. J. Oper. Res. **291**, 447–456 (2021)
11. Blum, A., Caragiannis, I., Haghtalab, N., Procaccia, A.D., Procaccia, E.B., Vaish, R.: Opting into optimal matchings. In Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 2351–2363. SIAM (2017)
12. Caragiannis, I., Filos-Ratsikas, A., Procaccia, A.D.: An improved 2-agent kidney exchange mechanism. Theoret. Comput. Sci. **589**, 53–60 (2015)
13. Carvalho, M., Lodi, A.: Game theoretical analysis of kidney exchange programs. arXiv preprint arXiv:1911.09207 (2019)
14. Carvalho, M., Lodi, A., Pedroso, J.P., Viana, A.: Nash equilibria in the two-player kidney exchange game. Math. Program. **161**(1–2), 389–417 (2017)

15. Hajaj, C., Dickerson, J.P., Hassidim, A., Sandholm, T., Sarne, D.: Strategy-proof and efficient kidney exchange using a credit mechanism. In Twenty-Ninth AAAI Conference on Artificial Intelligence (2015)
16. Johannes, B.: New classes of complete problems for the second level of the polynomial hierarchy. Technical report, PhD thesis of TU Berlin (2011)
17. Klimentova, X., Viana, A., Pedroso, J.P., Santos, N.: Fairness models for multi-agent kidney exchange programmes. Omega 102333 (2020)
18. Lodi, A., Ralphs, T., Woeginger, G.: Bilevel programming and the separation problem. Math. Program. **146**(1–2), 437–458 (2014)
19. Reese, P., Boudville, N., Garg, A.: Living kidney donation: outcomes, ethics, and uncertainty. The Lancet **385**, 2003–2013 (2015)
20. Smeulders, B., Blom, D.A.M.P., Spieksma, F.C.R.: The Stackelberg kidney exchange problem is $\Sigma_2^p$-complete. In Proceedings of SAGT 2020, p. 342. Springer (2020)
21. Toulis, P., Parkes, D.C.: Design and analysis of multi-hospital kidney exchange mechanisms using random graphs. Games Econom. Behav. **91**, 360–382 (2015)
22. Valentín, M.O., Garcia, M., Costa, A.N., Bolotinha, C., Guirado, L., Vistoli, F., Breda, A., Fiaschetti, P., Dominguez-Gil, B.: International cooperation for kidney exchange success. Transplantation **103**(6), e180–e181 (2019)
23. Woeginger, G.J.: The trouble with the second quantifier. 4OR, 1–25 (2021)