Optimizing Kidney Exchange Programs with Budget and Time Constraints
Valeria Romanciuc
UMinho | 2019

Universidade do Minho
Escola de Ciências

Valeria Romanciuc

Optimizing Kidney Exchange Programs with Budget and Time Constraints

outubro de 2019

Universidade do Minho
Escola de Ciências

Valeria Romanciuc

Optimizing Kidney Exchange Programs with Budget and Time Constraints

Dissertação de Mestrado
Mestrado em Matemática e Computação

Trabalho efetuado sob a orientação do
Professor Doutor Filipe Pereira Pinto Cunha Alvelos
e da
Professora Doutora Ana Jacinta Pereira da Costa Soares

outubro de 2019

Direitos de Autor e Condições de Utilização do Trabalho por Terceiros

Acknowledgements

This dissertation would be impossible without the continued support of some amazing people in my life, to whom I would like to show my gratitude.

First and foremost, to my parents and brother, for always supporting me in my decisions and being so loving and understanding, even when we are so far away. I miss you all so much.

To my supervisors, Prof. Filipe and Prof. Ana Jacinta, for their copious amounts of patience and kindness, as well as all the opportunities they created for me this past year. I could not be more grateful for having been under your wing.

To Hélder and Speed, for carrying me in the beginning and teaching me tons of things about Software Engineering. To Luísa, for being a great friend and helping me keep some of my sanity when times were tough. To the guys at the lab, for adopting me and providing great life advice and pep talks whenever I needed.

Of course, to Leo, to whom thanks is not enough. For being there for me everyday, for always being able to make me feel better, for his unconditional support and for helping me from the first "Send" to the last. I could not be more lucky.

To all my friends, who always believed in me despite being there for the monthly existential crisis.

To everyone else who supported me during this process, you have my sincerest gratitude.

This work has been supported by FCT — Fundação para a Ciência e Tecnologia within the project UID/CEC/00319/2019, by the ERDF — European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme, by National Funds through the Portuguese funding agency, FCT, within project POCI-01-0145-FEDER-016677 and by the COST Action CA15210 - European Network for Collaboration on Kidney Exchange Programmes.

Statement of Integrity

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

Otimização em Programas de Doação Renal Cruzada com Restrições de Orçamento e Tempo

Resumo

A doença renal terminal afeta milhões de pessoas no mundo, sendo que a transplantação é o melhor tratamento disponível. A maioria dos rins para transplantes são obtidos de dadores cadavéricos, mas devido ao aumento do número de pacientes a necessitar de um transplante, esta oferta não é suficiente para responder à procura. Esta questão levou ao aparecimento de novas estratégias de transplantação, nomeadamente a obtenção de rins de dadores vivos. No entanto, mesmo quando existe um dador vivo disponível, em mais de 30% dos casos ele é fisiologicamente incompatível com o esperado recetor do transplante.

Os Programas de Doação Renal Cruzada emergiram de forma a oferecer uma solução alternativa para pacientes com dadores vivos disponíveis, mas incompatíveis, onde dois ou mais pares incompatíveis de dador-recetor podem encontrar um transplante compatível trocando de dador. Torna-se necessário definir um plano de transplantação num conjunto de pares incompatíveis dador-recetor, de forma a decidir quais as trocas a efetuar. Contudo, após a definição deste plano, são efetuados testes de compatibilidade mais precisos, podendo revelar novas incompatibilidades, levando ao cancelamento do plano. Além disso, devido à complexa logística associada a estes testes, existem também restrições a nível de tempo e de orçamento.

Nesta dissertação, pretendemos responder ao problema de decidir quais os testes que devem ser efetuados dentro de um conjunto de pares incompatíveis dador-recetor, tendo em conta probabilidades de falha, bem como restrições de tempo e orçamento. Assim, estudamos três problemas com diferentes níveis de restrição destes recursos no contexto dos Programas de Doação Renal Cruzada e apresentamos vários métodos de resolução para cada um, baseados em Programação Inteira.

Palavras-chave: Programas de Doação Renal Cruzada, Otimização, Programação Inteira

Optimizing Kidney Exchange Programs with Budget and Time Constraints

Abstract

End-stage renal disease affects millions of people worldwide, with transplantation being the best form of treatment available. Most kidneys for transplants are obtained from deceased donors, but due to the increasing number of patients in need of a transplant, this supply has not been enough to meet the demand. This issue forced new transplantation strategies to emerge, namely obtaining kidneys from living donors. However, even when there is a willing living donor, in over 30% of the cases they are incompatible with the intended recipient of the transplant.

Kidney Exchange Programs emerged to provide an alternative solution for patients with willing but incompatible donors, where two or more incompatible donor-patient pairs can find a compatible transplant by swapping donors. A transplantation plan needs to be defined in a pool of incompatible donor-patient pairs in order to decide which exchanges should take place. However, after the transplantation plan is defined, more accurate compatibility tests are performed, possibly revealing new incompatibilities and leading to the cancellation of the plan. Furthermore, due to the complex logistics associated with these tests, there is also usually a time and budget limitation.

In this dissertation we address the problem of deciding which tests to carry out in a pool of incompatible donor-patient pairs, taking into account probabilities of match failure, as well as budget and time constraints. We study three problems with varying degrees of limitation of resources in the context of Kidney Exchange Programs and present several methods for solving each one, based on Integer Programming.

Keywords: Kidney Exchange Programs, Optimization, Integer Programming

Contents

List of Figures

List of Tables

Chapter 1

General Introduction

1.1 Context

About one in a thousand people in Europe suffer from end-stage renal disease [1]. Between dialysis and transplantation, the latter is the preferred method of treatment and for the majority of people, kidneys are obtained from deceased donors. However, since the number of patients in need of a kidney continues to grow, it becomes increasingly difficult to meet the demand through this source alone. In an attempt to increase the number of kidney transplants, various strategies involving living donors emerged in recent times. Nevertheless, one issue with living donor transplantation is the fact that, even when a willing donor is available, in about 30% of the cases, the possible donor and patient are physiologically incompatible [1]. Kidney Exchange Programs arose to bypass these compatibility issues and increase the number of patients obtaining a transplant successfully.

The concept of a Kidney Exchange Program (KEP) was first proposed by Felix Rapaport in the 80's [2]. By considering two pairs consisting of a patient and a willing but incompatible donor, where the donor in one pair is compatible with the patient in the other, Rapaport proposed a donor swap between the two pairs. In a scenario where initially no transplants could be conducted due to incompatibilities, the proposed exchange would allow for both transplants to take place. This was called the "pairwise exchange", but was soon generalized to include more incompatible pairs in a single exchange.

Different approaches were developed in the context of KEPs in order to decide which transplants to conduct out of the pool of donor-patient pairs, namely mathematical techniques involving linear programming and optimization. In these approaches, a transplantation plan is defined and afterwards, compatibility tests are conducted in order to decide which transplants can actually be carried out. This uncertainty is an important aspect to consider in the context of these programs.

1.2 Goals

The motivation for our approach is two-fold: firstly, given the uncertainty described in KEPs, we are interested in studying the problem by taking this aspect into account and, for this reason, we will consider probabilities of compatibility failure; secondly, it is usually unfeasible to test the entire pool of incompatible donor-patient pairs to verify compatibility, as these tests are very resource intensive. If such was possible, we would have complete information of the network and would easily be able to decide which transplants to carry out. Thus, considering the complex logistics associated with performing the crossmatch tests, we will also assume there is a budget for the available number of tests. This is a main aspect of our approach, which differs from most approaches found in the literature. We consider the specific scenario where the maximization of the number of transplants is conducted in regard to a certain predefined budget, and we are interested in understanding what influence this limitation has on the results.

Besides the budget limitation on the number of crossmatch tests, in real-life scenarios there is often a time restriction as well. In these cases, the tests are conducted in *testing rounds* and the information about participating pairs is updated periodically, since some pairs may leave the program and new ones may join.

In this dissertation, we address the problem of deciding which possible transplants to test, in three different settings, according to a number of different methods. We take into consideration three aspects related to the operation of KEPs: uncertainty and limitation of resources and time. The first is addressed by associating a probability of compatibility failure; the second, by considering a budget for the number of compatibility tests that can be performed; the third, by considering testing rounds in which sets of tests are carried out.

Three different problems will be studied in the described context: the simultaneous problem, where a set of possible transplants is tested simultaneously (according to the budget); the unlimited problem, where the budget constraint is relaxed and we consider an unlimited number of tests, performed in an unlimited number of testing rounds; finally, the two-rounds problem, where a budget is again considered but instead of conducting the tests simultaneously, they are divided into two rounds. We are interested in measuring these methods' performance and see how they compare against one another.

1.3 Dissertation Outline

The dissertation is organized as follows.

Chapter 2 provides an overview of Kidney Exchange Programs, starting on basic concepts in the context of this subject. A summary of different modalities in KEPs is given, as well as current practices of operating KEPs around the world.

Chapter 3 describes the framework used in addressing this problem, as well as some mathematical techniques used, namely integer programming based approaches found in the literature. The concept of recourse is also presented and explained.

Chapter 4 is dedicated to describing the simultaneous problem and several methods to solve it. In the end, the conducted computational tests are presented and analyzed.

Chapter 5 provides a description of the unlimited problem, as well as the studied methods, which are divided into two categories. In one of them, the concept of probing is also presented and described. The final section is dedicated to discussing the obtained results.

Chapter 6 focuses on the two-round problem. After the problem description, three different policies are presented and their results are discussed.

Finally, chapter 7 contains general conclusions regarding the presented problems and methods, as well as final remarks about future work and the contribution of this dissertation.

Chapter 2

Kidney Exchange Programs

2.1 End-Stage Renal Disease

End-stage renal disease (ESRD) is characterized as an irreversible decrease in renal function which, if left untreated, leads to death [3]. This condition requires one of the two following alternatives of treatment: dialysis or kidney transplantation [4]. It has been shown that transplantation offers several advantages over dialysis, both in terms of the patient's quality of life and survival, on top of being much less costly [5, 6, 7].

Despite this, as of 2016, dialysis is still the most common form of treatment, with around 60% of patients in the EU undergoing some form of dialysis, as opposed to the other 40% who obtain a kidney transplant [8].

Being the most advantageous alternative, efforts have been made in order to provide more patients with a kidney transplant, which can be obtained from either deceased or living donors. In the first case, the patient in need of a transplant will be placed on a waiting list until a suitable deceased donor is found. However, as the number of ESRD patients steadily increases [9], the waiting times for a transplant also become increasingly longer and the number of deceased donors is far from being enough to provide every patient with a kidney [10]. In the second case, the patient has a donor (usually a relative) who is willing to provide a kidney for transplantation. Kidneys provided by living donors also have proved to yield better results when compared to the ones obtained from deceased donors, both in patient and graft survival [10, 11]. However, in over 30% of the cases, this transplant is not going to take place as the donor is physiologically incompatible with its intended recipient [12, 11]. Compatibility is determined based on blood types (ABO) and human leukocyte antigens (HLA) matching [12, 13].

In the first case, there are four different blood types: A, B, AB and O, each corresponding to the presence of protein A, B, both or neither, respectively [13]. A patient is incompatible with a donor who has one of the proteins which the patient does not. For instance, a patient with blood type A can only receive kidneys from donors with blood types A or O. Table 2.1 provides a map of the compatibilities between patients and donors of different blood types.

As for HLA incompatibility, it occurs when the patient has preformed antibodies against the donor's antigens. These antibodies can develop through previous transplants, blood transfusions or pregnancy [4]

Patient \ Donor	O	A	B	AB
O	•			
A	•	•		
B	•		•	
AB	•	•	•	•

Table 2.1: Map of compatibilities between the different blood types of patients and donors.

and their presence increases the likelihood of rejection of the graft. Their existence is determined through a test called *crossmatch*: if it is positive, it means the antibodies are present in the patient's system, so the transplant should not take place. If the crossmatch is negative, then there are no harmful antibodies present and the transplant may proceed.

As an effort to decrease the drawbacks of incompatibility, a third alternative arose in recent years based on the study of incompatible pairs and possible donor swaps between them: the so-called kidney exchange programs. We will now focus on this subject, as it is the theme of this dissertation.

2.2 Emergence of Kidney Exchange Programs

The basic concept of a kidney exchange was first presented in 1986 by Felix Rapaport [2]. The idea was to give incompatible donor-patient pairs an opportunity to find a reciprocally compatible donor among them and exchange one of their kidneys so that the transplants could be carried out.

The simplest type is the pairwise exchange, which can be formulated as follows: let us consider a patient P_1 with a willing but incompatible donor, D_1. Let us also consider another similar pair P_2 and D_2, where D_2 is an incompatible donor to P_2. In this scenario, no transplant would be conducted given the existing incompatibility between the pairs. However, if D_1 was compatible with P_2 and D_2 was compatible with P_1, we could simply swap donors and perform this procedure on both patients. A key factor in this exchange, albeit obvious, is that the donations may proceed only because each donor's associated patient will receive a kidney. This exchange, which can also be designated as "2-way exchange", is represented in figure 2.1.

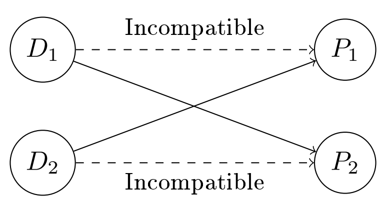

Figure 2.1: Representation of a pairwise kidney exchange. The dashed arrows represent incompatibility and the full arrows represent compatibility between donor (D_i) and patient (P_i).

2.3 Types of Kidney Exchanges

There are several other exchange modalities to a kidney exchange, which allow for different alternatives to be adopted in a program, some of which will be presented in this section.

2.3.1 *k*-way Kidney Exchange

This modality generalizes the pairwise exchange by involving k donor-patient pairs (see figure 2.2). On the one hand, it allows for a bit more flexibility, since the reciprocal compatibility between two pairs is no longer a requirement. On the other hand, from a logistical standpoint, it becomes increasingly difficult to conduct an exchange the larger the k is, as the surgeries on the patients involved in must be conducted simultaneously to decreases chances of withdrawal from the program by pairs after receiving a kidney. Therefore, the shorter the exchange, the smaller the amount of possible cancelled transplants due to these issues, which is why k is usually limited to two, three or four.

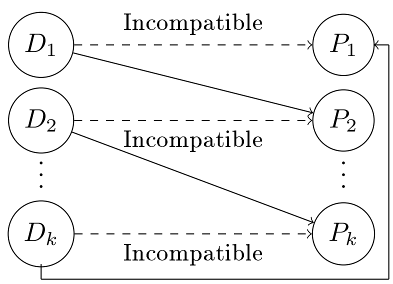

Figure 2.2: Representation of k-way kidney exchange. The dashed arrows represent incompatibility and the full arrows represent compatibility between donor (D_i) and patient (P_i).

2.3.2 Altruistic Donor Chains

Another inclusion to the set of donors to be considered is of *altruistic* donors, that is, donors who are willing to donate a kidney, despite not having an associated patient in need of a transplant [14]. These donors can either donate directly to a patient on the deceased donor waiting list, or transform the organization of an exchange of incompatible pairs into a chain (see figure 2.3). In the latter case, an altruistic donor donates a kidney to an incompatible pair, provided that this pair donates a kidney to another incompatible pair in the pool. This chain continues until the last pair in the exchange donates a kidney to a patient in the deceased donor waiting list. These donors introduce more flexibility into a program as well, since the requirement of simultaneity in procedures is relaxed, which helps increase the number of transplants carried out.

2.3.3 Inclusion of Compatible Donors

The donor pool can also include compatible pairs in an exchange. The benefits are two-fold: on the one hand, it increases the chance of incompatible pairs finding a compatible donor; on the other hand, it offers

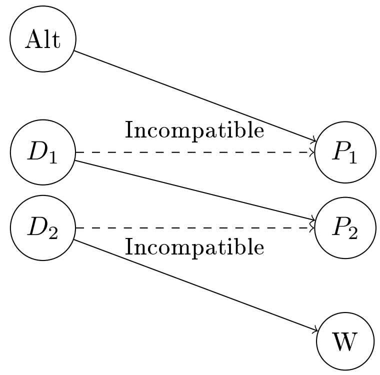

Figure 2.3: Representation of a “domino paired donation" with three transplants. The chain starts with an altruistic donor and ends with a patient on the deceased donor waiting list. The dashed arrows represent incompatibility and the full arrows represent compatibility between donor and patient.

compatible pairs the chance of finding a better quality kidney transplant or possibly a younger donor [15, 16].

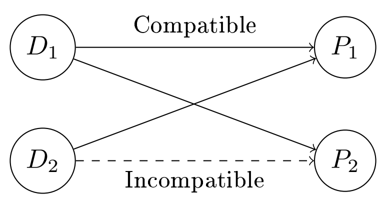

Figure 2.4: Representation of a kidney exchange including compatible pairs. The dashed arrows represent incompatibility and the full arrows represent compatibility between donor (D_i) and patient (P_i).

2.3.4 Patient Desensitization

Some patients are HLA-sensitized, which makes it difficult to find a compatible donor. However, in some cases of highly sensitized patients, a desensitization procedure may take place in order to remove some of the harmful antibodies present in the patient's system, thus decreasing the chances of rejection of the transplanted kidney. Nevertheless, a compatible transplant still leads to better outcomes [1].

2.3.5 Multiple Donors

To further increase the chances of identifying potential exchanges, a patient may have multiple (incompatible) donors associated in a KEP.

2.4 Kidney Exchange Programs in Operation

Kidney Exchange Programs have been implemented in several countries with different approaches. In this section, we will go over the approaches some countries have adopted and the effects each KEP had on the respective health care systems, focusing on the biggest implemented programs. Figure 2.5 presents a map with the current state of KEPs in Europe (see [1]).

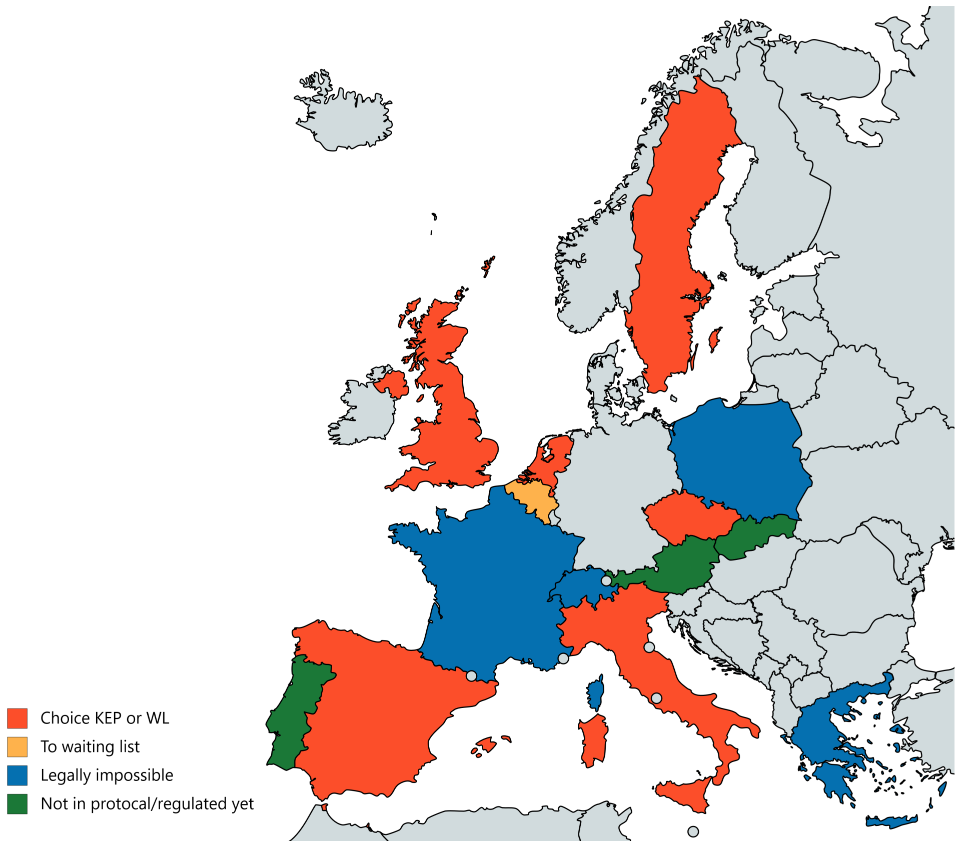

Figure 2.5: Development of KEPs by country [1].

2.4.1 The Netherlands

This program was established in 2004, the first one in Europe [1], and is organized by the Dutch Transplant Foundation. As of 2017, it had the highest number of living donor transplants [17]. Between 2004 and 2016, 284 transplants were performed from exchanges, comprising about 6% of all living donor national transplants. The maximum number of pairs allowed in an exchange is four and this program also includes altruistic donors and compatible pairs [1].

2.4.2 The United Kingdom

The National Living Donor Kidney Sharing Schemes (NLDKSS) is operating since 2007 on all four countries of the UK, and is organized by the National Health Service Blood and Transplant (NHSBT) [14]. As of 2017, it has reported an average of 135 transplants per year [1]. This program allows altruistic and compatible donors, patient desensitization, as well as multiple donors per patient. The number of pairs per exchange is restricted to three, although pairwise exchanges are prioritized [14], and there is a matching run every three months to determine the best exchange plan for every 300 pairs. According to [18], in 2017, 22% of adult kidney national transplants were possible because of this program.

2.4.3 Spain

Established in 2009 by *Organización Nacional de Transplantes* (ONT), it is the second largest KEP in Europe, with a total of 147 transplants reported (an average of 35 transplants per year) [1]. This program operates similarly to the UK's and it also includes altruistic donors, without a limit on the length of the chains.

2.4.4 Portugal

The Portuguese KEP was started in 2010 under the authority of the *Instituto Português do Sangue e da Transplantação*. It is still fairly small, with only nine transplants reported to have been performed through the program until the end of 2016. Only 2- and 3-way exchanges are allowed and matching runs are conducted twice a year [19]. Compatible pairs are not allowed in the donor pool and altruistic donors are not yet legally regulated.

2.4.5 Rest of Europe

There are other smaller programs implemented in European countries, namely in Austria, Belgium, Czech Republic, France, Italy and Poland, as well as some others in preparation, such as in Greece, Hungary, Slovakia and Switzerland [1]. They are all organized centrally but have different requirements and regulations. More detailed information can be found in [1].

2.4.6 The United States

The US's case is different than most countries in Europe. There is a main national program in place: the United Network for Organ Sharing (UNOS), which was established in 2010. However, there are at least two other national programs in operation, as well as other regional or single-center programs. Large centers conduct most of the exchanges by themselves, reporting only the hard-to-match patients to the national programs. Only 2-way or 3-way exchanges are allowed, but chains can have up to four exchanges [1].

The main problem in the US is the extremely high failure rate (over 90%), which is due to the fact that the KEP involves mainly hard-to-match patients, as well as the lack of standardized testing for HLA matching [1].

2.4.7 Rest of the World

Other countries with large established KEPs are Australia [20], Canada [21], and South Korea [22]. For more information, see [1].

2.5 International KEPs

In Europe, efforts have been made in order to establish joint programs involving different countries. This allows not only for an expansion of the donor pool, but also for more diversity of donors, which may help solve some issues unique to each country.

Taking into consideration all the requirements for participation in a KEP, it becomes even more difficult to meet them in procedures carried out between various centers and even more so countries. However, one essential constraint is to ensure that the number of transplants performed in each center when in a joint pool is at least the same as the number of transplants conducted by each center alone. This way, it is guaranteed that each center does not lose any transplant by participating in a joint program.

Austria and Czech Republic have established the first reported international pairwise exchange in 2016 [23]. Both extraction surgeries were performed simultaneously and the recipients reported no signs of rejection.

Denmark, Sweden and Norway have also joined pools, forming the Scandiatransplant Kidney Exchange Program. The first match run to find compatible exchanges was performed successfully in May of 2019. Although Estonia, Finland and Iceland are also part of Scandiatransplant, they do not have yet any incompatible pairs participating in the transnational program.

A joint program involving Italy, Spain and Portugal is being developed, having the first two countries already performed a transnational exchange [24].

Chapter 3

Literature Review and Preliminary Concepts

Kidney Exchange Programs can be addressed with different approaches [25, 26, 27, 28]. Although these approaches differ from one another, they share the same baseline structure. In this chapter we will introduce some concepts concerning this structure, as well as describe some of the current solutions being studied to implement KEPs.

3.1 A Graph-Theoretic Framework for KEPs

The results of the previously mentioned preliminary compatibility tests conducted between incompatible pairs are used to build a *compatibility graph*, which summarizes the possible swaps that can be performed in a set of incompatible pairs (which will be referred to as *the pool*). Thus, a network of incompatible pairs is represented via a directed graph $G = (V, A)$, where V is the set of vertices, each vertex representing an incompatible donor-patient pair, and A is the set of arcs, each arc representing compatibility between the donor of one pair and the patient of the other. As such given two vertices $i, j \in V$, the arc (i, j) exists in A if the donor in pair i is compatible with the patient in pair j.

A feasible exchange between pairs in this network is represented through a cycle in the graph, where its length indicates the number of participating pairs (so a cycle of length k (or a k-cycle) corresponds to a k-way exchange). If a program includes altruistic donors, chains are also included and they are represented as paths in the graph.

Finally, models for the kidney exchange problem are also characterized by an objective which maximizes social wellness, most commonly the total number of transplants. Nevertheless, other objectives can be considered depending on the features of the pool (see [26] for further information).

Figure 3.1 provides an example of a directed graph $G = (V, A)$ representing a KEP network. The set of vertices is $V = \{0, 1, 2, 3\}$ and the set of arcs is given by $A = \{(0, 2), (1, 0), (1, 3), (2, 1), (3, 1)\}$. The existing cycles (possible exchanges) are $c_1 = \{(0, 2), (2, 1), (1, 0)\}$ and $c_2 = \{(1, 3), (3, 1)\}$.

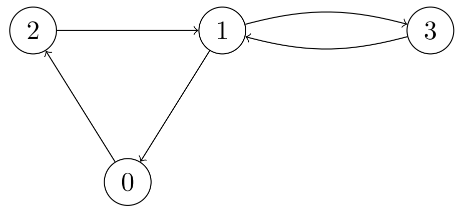

Figure 3.1: Example of a KEP instance with four incompatible pairs.

3.2 Deterministic Models

Several authors have modelled this problem through integer programming [26, 27, 29]. The kidney exchange problem is typically treated as a deterministic problem, meaning there is no uncertainty to the data and the number of transplants is most commonly maximized, as previously stated. Considering the constraint on the length of a cycle as being k, the main goal of this problem is to find a set of vertex-disjoint cycles, with length at most k, which has the maximum cardinality.

Next we will present the two baseline integer programs used in KEPs, which were first presented and described in [30] in detail.

3.2.1 The Cycle Formulation

Let C be the set of all cycles in the graph G. Let us consider $C(k) \subseteq C$, the set of all cycles of G with length at most k. For each cycle $c \in C(k)$, we define a decision variable z_c as:

$$z_c = \begin{cases} 1, & \text{if the cycle } c \text{ is selected} \\ 0, & \text{otherwise} \end{cases}$$

The weight w_c of each cycle $c \in C$ is defined as $w_c = \sum_{(i,j)\in c} w_{ij}$, where w_{ij} is the weight associated to each arc. When the goal is to maximize the number of transplants, the weight for all arcs is set to 1. This model maximizes the weight of a set of disjoint cycles, being formulated as follows:

$$\text{Max} \quad \sum_{c\in C} w_c z_c \tag{3.1}$$

Subject to:

$$\sum_{c:i\in c} z_c \leq 1, \quad \forall i \in V, \tag{3.2}$$

$$z_c \in \{0,1\}, \quad \forall c \in C. \tag{3.3}$$

The objective function 3.1 maximizes the number of transplants. Constraints 3.2 ensure that each vertex is selected at most once (that is, each donor can donate and each patient can receive at most one kidney). Constraints 3.3 define the domain of the decision variables.

3.2.2 The Edge Formulation

Let us consider a variable x_{ij} defined for each arc (i, j) in A as follows:

$$x_{ij} = \begin{cases} 1, & \text{if the donor in pair } i \text{ donates a kidney to the patient in pair } j \\ 0, & \text{otherwise} \end{cases}$$

The problem can be modelled as follows

$$\text{Max} \quad \sum_{c \in C} w_{ij} x_{ij} \tag{3.4}$$

Subject to:

$$\sum_{j:(j,i) \in A} x_{ji} = \sum_{j:(i,j) \in A} x_{ij}, \quad \forall i \in V, \tag{3.5}$$

$$\sum_{j:(i,j) \in A} x_{ij} \leq 1, \quad \forall i \in V, \tag{3.6}$$

$$\sum_{p:p \in \{1,k\}} x_{i_p i_{p+1}} \leq k - 1 \quad \forall \text{paths}\,(i_1, i_2, \ldots, i_k, i_{k+1}) \tag{3.7}$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A. \tag{3.8}$$

The objective function 3.4 maximizes the total number of transplants (in the case of unitary weights). Constraints 3.5 ensure the patient in pair i receives a kidney if, and only if, the donor in pair i donates a kidney. Constraints 3.6 make sure that the donor in pair i donates at most one kidney. Constraints 3.8 ensure that the cycles in the exchange do not exceed a length of k and constraints 3.8 define the domain of the decision variables.

These two models offer different advantages; the cycle formulation is known to be very efficient, but its number of variables grows exponentially with the number of pairs of an instance [30]. As for the edge formulation, despite having a polynomial number of variables, the number of constraints grows exponentially with the size of an instance [31]. This happens because of the constraints which limit the length of an exchange.

This trade-off between the number of variables and the number of constraints has been an ongoing topic of discussion, with computational experiments being made to assert which of the two formulations is the most efficient. It has also lead to the creation of "compact formulations", that is, formulations with a linear number of variables and constraints [27, 29, 31].

In [27], two compact formulations based on the edge formulation were presented, where both the number of variables and constraints are bounded by a polynomial in the size of the number of pairs: the *edge-assignment formulation* and the *extended edge formulation*. Both of them address the problem of limiting the length of a possible exchange with different approaches. For the first, new assignment variables are defined for each vertex according to the cycle it belongs to, so that cycle cardinality is written more easily. As for the second, the idea is to create copies of graph G, so that each copy contains a cycle with at most k arcs and each vertex can belong to at most one cycle. Again, this allows to express the cardinality constraints with a polynomial number of variables and constraints. However, both of these formulations have some drawbacks, namely the possibility of having multiple equivalent solutions, which can be a problem when it comes to solving these models using traditional methods (such as branch-and-bound) [27].

3.3 Probabilistic Models

3.3.1 Taking Uncertainty into Account

Although KEPs are commonly studied without considering uncertainty, in real-life situations the scenario is anything but deterministic. Even after a *transplantation plan* has been established (that is, the transplants to be carried out were selected), different situations may lead to transplant cancellations, such as withdrawals by pairs from a program, illness of a patient or donor, the sudden availability of a kidney transplant from the deceased donor waiting list and other last-minute unexpected issues [1].

On top of these issues, all the participating pairs of a transplantation plan must undergo a more precise test (the actual crossmatch) before the planned transplants are carried out, in order to verify the virtual compatibilities of the compatibility graph. Performing these tests is a complex task, both in terms of time and associated costs, which is the reason why they are conducted only on the pairs of a transplantation plan, instead of the entire pool. They may lead to the discovery of new incompatibilities between donors and patients (which is very likely to happen [31]), preventing the plan to move forward. Thus, it becomes necessary to take into account this uncertainty.

In the literature, this has been studied in [25, 26, 32, 33, 34] by associating a probability of failure to either each arc or each vertex and maximizing the expected number of transplants instead of the total number of transplants.

In [32], some algorithms are proposed to calculate the expectation of the number of transplants by associating probabilities of failure to both arcs and vertices. However, since these methods rely on the enumeration of all subsets of the set of arcs, they are limiting in terms of the size of the instance. As an attempt to circumvent this issue, the author proposes an intermediate stage where a database is constructed, where all possible cycle configurations for small graphs are associated to their corresponding expected values. Nevertheless, the approach was still computationally expensive and stays limited to cycles of length 3.

In [34], the authors deal with uncertainty with a two-stage integer problem model. The first stage addresses the decision problem, in which subsets of arcs should be tested for crossmatch. The uncertain variables, which values are revealed after the first stage decision is made, are the existence of the arcs. Given that each realization of the uncertain variables leads to a scenario, the resulting integer programming model is extremely large. The authors prove the complexity of the selection problem presented is NP-hard, even when the maximum cycle length is $k = 2$.

This approach differs from others for two main reasons: they impose almost no limitations to the selection problem, namely regarding the restriction of choosing among either disjoint cycles or chains; and they consider an upper bound for the number of crossmatch tests available. Indeed, the scenario chosen can not have a number of arcs higher than the number of tests available. This constraint is realistic since, as we have mentioned before, performing crossmatch tests is very resource-intensive and usually can not be done on the entire pool.

Although these approaches provide strong models for solving the kidney exchange problem, in practice they are difficult to use for instances with large pools of incompatible pairs, as it is the case with most national programs.

3.3.2 Recourse Policies

In the previously mentioned schemes, the authors assume that once a solution is established, it can not be altered. Thus, if issues arise after the creation of a transplantation plan that lead to cancellations, there is nothing to be done other than establishing a different plan.

However, these possible failures in a network can be addressed by considering *recourse policies*, where some part of a solution may still be used or some transplants can be rearranged. There are mainly two types of recourse being studied: *internal* recourse and *full* recourse. In the latter, if a failure occurs, a totally new solution can be considered.

Regarding internal recourse, we are looking for smaller cycles embedded in the cycles of our solution. Accordingly, if a transplant fails in a cycle but a shorter cycle within the first one can still be considered, we would be able to make use of part of the solution instead of wasting all the conducted tests.

This idea was first considered in [14] by Manlove et al., where the concept of *backarc* was introduced. A definition is presented next.

Definition 3.3.1. *Let $G = (V, A)$ be a directed graph, where V is the set of vertices and A is the set of arcs. Let us consider a 3-cycle c in G. Arc (i, j) is a backarc of c if $(j, i) \in c$.*

In example 3.2, a KEP network with three incompatible pairs is defined. The 3-cycle is composed by arcs $a = (1, 2), b = (2, 3)$ and $c = (3, 1)$, and $d = (1, 3)$ is a backarc of c. The 3-cycle is said to have an *embedded* 2-cycle (the latter being composed by arcs c and d).

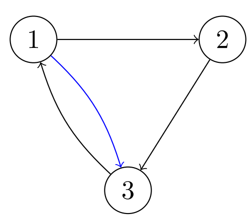

Figure 3.2: Example of a KEP instance with three incompatible pairs and one backarc.

Manlove et al. established multiple objectives for the problem, the first being to maximize the number of *effective* 2-cycles (which are either 2-cycles or 3-cycles with embedded 2-cycles). Among the objectives is also the maximization of the number of backarcs in the 3-cycles.

Klimentova et al. proposed new algorithms in [26] taking into account failure of vertices and arcs, for three different recourse schemes (no recourse, internal recourse and *subset recourse*).

A robust optimization approach is described in [35], being formulated as a two-stage optimization problem. The authors focus on maximizing the utility in the worst-case scenario of an instance. They consider three policies: *simple recourse*, where the costs of the cancelled transplants are taken into account; even though this policy does not enable the recovery of failing cycles or chains, it still allows for a more “informed" decision regarding the instance. An integer programming model is presented for the second policy, *backarcs recourse*, which enables for part of a failing cycle or chain to be recovered (considering only the pairs that were already involved). The last policy presented is full recourse, where other pairs in the instance are considered for the recovery, and a different integer programming model is presented as well.

In this approach, a referred advantage when compared to other stochastic models is the fact that no assumptions about the distribution of the probability of failure are needed. This allows to circumvent situations where modelling the probability distribution is either very difficult or outright impossible.

3.4 Budget Constraints

A central aspect of this dissertation is the focus on the budget limitation for crossmatch tests. As previously mentioned, there are complex logistics associated to KEPs, one of them being the limitation of the resources which can be spent performing crossmatch tests. Since this aspect is usually not considered in the literature, we are interested in studying a problem in the context of KEPs with this constraint.

Furthermore, we will also consider testing rounds, which simulate how tests are conducted in operating programs periodically. A round is characterized as a set of crossmatch tests conducted simultaneously, for which the results are known at the end of the round. The following example illustrates these concepts in the context of a KEP. Let us consider the instance in figure 3.3.

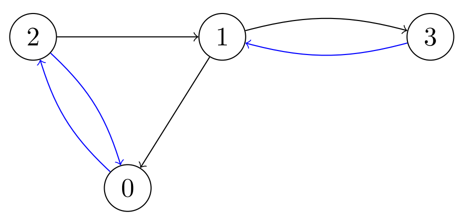

Figure 3.3: Example of a KEP instance with four incompatible pairs. The blue arcs represent the possible transplants.

In a round, we wish to select the set of arcs which maximizes the total number of transplants. Without any restrictions, we would choose the arcs in cycles $\{(1,3),(3,1)\}$ and $\{(0,2),(2,0)\}$, since they could potentially yield 4 transplants. However, if we consider a budget of 3, this set of arcs would no longer be eligible, as it has 4 arcs. For a budget of 3, the chosen arcs would be the ones in cycle $\{(1,0),(0,2),(2,1)\}$.

At the end of the round, we perform the crossmatch tests on the selected arcs. Supposing the blue arcs are the ones for which the crossmatch is negative (i.e., there is no incompatibility between the donor in one pair and the patient in the other), arcs $(2,1)$ and $(1,0)$ would be excluded from the network, since these transplants are not possible to be conducted. At the start of the next round, we continue this procedure on the updated network.

Chapter 4

Simultaneous Problem

In this chapter we will provide the definition and formulation of what we call the "simultaneous" problem, as well as several solution methods.

4.1 Problem Definition

In the simultaneous problem, we want to maximize the number of transplants, knowing there is a limited number of arcs that can be tested simultaneously. We will focus on deciding which arcs to test, respecting the budget limitation on the number of tests. Several approaches will be presented to carry out this selection. We also assume that only arcs which were previously tested can be selected for the final transplantation plan.

We start with the same framework as previously described in Chapter 3, a directed graph $G = (V, A)$, where V is the set of vertices and A is the set of arcs. Each possible exchange is represented by a cycle in the graph. The set of all cycles is denoted by C.

To each arc $(i, j) \in A$ we associate a weight p_{ij}, representing the probability of failure of that possible transplant.

With this, we may calculate the expected value of a cycle c with the following formula

$$E(c) = |A(c)| \prod_{(i,j)\in A(c)} (1 - p_{ij}), \quad \forall c \in C, \tag{4.1}$$

where $A(c)$ is the set of arcs of c and $|A(c)|$ is its cardinality.

4.2 Omniscient

For the purposes of comparison, we start by considering the so-called "omniscient" solution, which represents the maximum possible number of transplants that can be achieved with an unlimited budget and testing rounds. More specifically, after crossmatching all the arcs of an instance, the omniscient solution provides the solution with the highest number of transplants.

This solution can be interpreted as an "upper bound" for the methods being studied, as it is the best possible result for a given instance.

4.3 Solution Methods

4.3.1 Expectations Model

The expectations model is based on the cycle formulation, with additional budget constraints. To each arc $(i, j) \in A$ there are some associated parameters, namely the probability of failure p_{ij} as previously stated, as well as the cost of performing a test on that arc, b_{ij}. Associated with each cycle $c \in C$ is its expected value, denoted by $E(c)$, as well as a parameter a_{ij}^c, for each arc $(i, j) \in A$ and cycle $c \in C$, which is 1 if arc (i, j) belongs to cycle c and 0 otherwise. We also have a parameter B for the total available budget.

The decision variables are defined as follows:

- $y_{ij} = \begin{cases} 1, & \text{if arc } (i,j) \text{ is tested} \\ 0, & \text{otherwise} \end{cases} \qquad \forall (i,j) \in A$
- $w_c = \begin{cases} 1, & \text{if cycle } c \text{ is selected for the transplantation plan} \\ 0, & \text{otherwise} \end{cases} \qquad \forall c \in C$

The integer programming model is given by

$$\text{Maximize} \qquad z = \sum_{c \in C} E(c) w_c \tag{4.2}$$

$$\text{Subject to:} \qquad \sum_{c \in C} \sum_{(i,j) \in A} a_{ij}^c w_c \leq 1, \quad \forall i \in V \tag{4.3}$$

$$w_c - a_{ij}^c y_{ij} \leq 0, \quad \forall (i,j) \in A, \forall c \in C \tag{4.4}$$

$$\sum_{(i,j) \in A} b_{ij} y_{ij} \leq B \tag{4.5}$$

$$w_c, y_{ij} \in \{0, 1\}, \forall c \in C, \forall (i,j) \in A \tag{4.6}$$

The objective function (4.2) maximizes the expected number of transplants. Constraints (4.3) ensure that each vertex is in, at most, one selected cycle for the exchange. This is necessary because each pair can only donate (and consequently receive) at most one kidney. Constraints (4.4) ensure that only cycles where every arc is tested can be selected. Finally, constraints (4.5) ensure that the total available budget is not exceeded.

Since in a real-life scenario the length of an exchange is usually limited to three, we will only consider cycles of length two and three in our problems.

It should be noted that this model yields the optimal solution when the number of arcs in this solution is equal to the budget. However, if the budget exceeds the number of arcs in the solution, the number of transplants is not improved, thus we propose a heuristic to add more arcs besides the ones already obtained.

We start by calculating a solution, S_1, using the presented model (4.2-4.6). If the number of arcs in S_1 is smaller than the available budget, B, we exclude S_1 from the possible solutions by adding the following constraint to the model

$$\sum_{c \in C_1} (1 - w_c) + \sum_{c \in C_0} w_c \geq 1 \tag{4.7}$$

where C_1 is the set of indexes of decision variables in S_1 equal to 1, and C_0 is the set of indexes of decision variables in S_1 equal to 0. This constraint ensures that a given solution (and only that one) is excluded from the set of possible solutions.

We then re-optimize the model to obtain a new solution, S_2. Any arc in S_2 that is not in S_1 is then added to S_1, and S_2 is excluded from the possible solutions. We repeat this process until our constructed solution has B arcs. Algorithm 1 formalizes this procedure.

Algorithm 1: Selecting a set of B arcs

Initialize $S = \varnothing$
while $|S| < B$ **do**
  Optimize the expectations model and obtain a solution S_1
  foreach *Arc* x *in* S_1 **do**
    if $x \notin S$ *and* $|S| < B$ **then**
      Add x to S
  Exclude S_1 from the possible solutions

Let us consider the example of a KEP network shown in figure 4.1. This instance has four incompatible donor-patient pairs, represented by the set of vertices $V = \{0, 1, 2, 3\}$. The possible transplants are given by the set of arcs $A = \{(0,2), (1,0), (1,3), (2,0), (2,1), (3,1)\}$ and the cycles in this network are $c_0 = \{(0,2), (2,0)\}$, $c_1 = \{(1,3), (3,1)\}$ and $c_2 = \{(0,2), (2,1), (1,0)\}$, with expected values of 1.425, 1.05 and 2.180, respectively (all calculated with the formula in 4.1). We also consider that the cost of performing a test on each arc is 1.

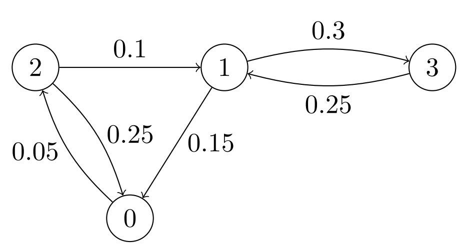

Figure 4.1: Example of a KEP instance with four incompatible pairs and probabilities of failure.

Recalling that we only consider solutions where all arcs can be tested, if $B = 2$, the optimal solution would be to test the arcs of cycle c_0, as it has the highest expected value of the 2-cycles. If $B = 3$, the optimal solution would be the cycle c_2. However, if $B = 4$, the optimal solution is the set of cycles c_0 and c_1, as their combined expected number of transplants is equal to 2.475. Thus, four is the maximum number of arcs we can select with the expectations model.

If we consider $B = 5$, following algorithm 1, we would start by obtaining the solution c_0 and c_1 with the expectations model. Then, after excluding it from the set of possible solutions, we re-optimize for the expected number of transplants and get cycle c_2, out of which we select arc $(2, 1)$ or $(1, 0)$ at random. Thus, one solution for $B = 5$ is $\{c_0, c_1, (2, 1)\}$.

4.3.2 Expectations with Backarcs

An improvement to our previous model is to consider *internal recourse* which, as explained in Chapter 3, consists of including *backarcs*. This allows us to rearrange part of a solution in case of failure after new incompatibilities are revealed. By doing this, instead of only considering simple cycles, we must consider *configurations*, which are defined in [26] as follows:

Definition 4.3.1. *A configuration of a cycle of size k is an equivalence class of isomorphic graphs with k vertices containing at least a cycle of size k.*

Thus, we will consider all possible configurations of a 3-cycle with zero, one (figure 4.2), two (figure 4.3) or three backarcs. Figure 4.4 shows the representatives of each equivalence class.

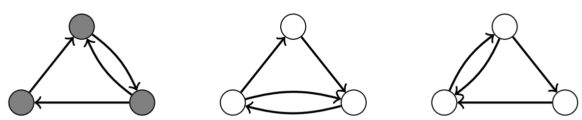

Figure 4.2: The three possible configurations of a 3-cycle with one backarc (one equivalence class) and the representative on the left.

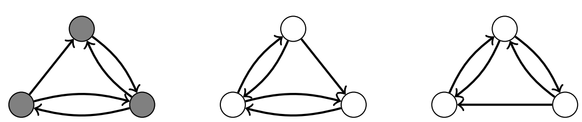

Figure 4.3: The three possible configurations of a 3-cycle with two backarcs (one equivalence class) and the representative on the left.

In the specific case of our model, this translates to considering a different set of cycles for the definition of the decision variables w_c, as they are no longer associated only to 2-cycles and 3-cycles, but also to all possible configurations of a 3-cycle. As such, we will additionally define the following sets: C_3^0 as the set of all 3-cycle configurations with no backarcs, C_3^1 as the set of all configurations with one backarc, C_3^2 as the set of all configurations with two backarcs and finally C_3^3 as the set of all configurations with three backarcs.

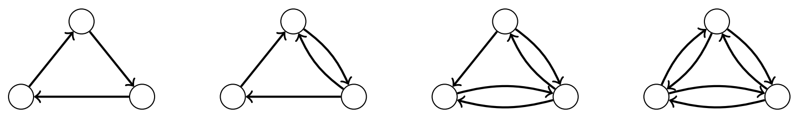

Figure 4.4: All four possible configurations (representatives of each equivalence class) of a 3-cycle.

We will extend the model (4.2-4.6) by considering C to be the union of C_3^0, C_3^1, C_3^2 and C_3^3, along with the set of 2-cycles (C_2): $C = C_2 \cup C_3^0 \cup C_3^1 \cup C_3^2 \cup C_3^3$. The expected value of a configuration with backarcs is calculated according to the formulae given in [32].

Let us consider once again the instance in figure 4.1. In this case, we have the configurations $c_0 = \{(0,2),(2,0)\}$, $c_1 = \{(1,3),(3,1)\}$, $c_2 = \{(0,2),(2,1),(1,0)\}$ and $c_3 = \{(0,2),(2,1),(1,0),(2,0)\}$ (with expected value of 2.515), which allow us to define the following sets: $C_2 = \{c_0, c_1\}$, $C_3^0 = \{c_2\}$ and $C_3^1 = \{c_3\}$. As configuration c_3 has a higher expected value than c_0 and c_1 combined, the arcs to be tested would be those in configuration c_3. This illustrates how including backarcs can give us better results for a limited number of crossmatch tests.

Similarly to the expectations model, the expectations model with backarcs provides an optimal solution for a certain budget when the number of arcs obtained is equal to the budget. If the budget is bigger, the solution is not improved by testing more arcs. Thus, we will once again consider algorithm 1 in order to add more arcs to the solution, with the difference that in line 1, the optimization is carried out with the expectations model with backarcs. We are now able to improve on the size of a set of arcs obtained with previous approach for a budget B.

4.3.3 Optimistic Approach

We also consider the approach where uncertainty is not taken into account when selecting the arcs to be tested, which we refer to as the "optimistic approach". The solution of this method corresponds to the number of planned transplants if all arcs of an instance existed. This approach uses a particular case of the model (4.2-4.6), where the probability of failure of all arcs in A is set to 0, which is equivalent to assuming there are no incompatibilities between the pairs involved. The expected value of a cycle is then simply its length, so the objective function maximizes the total number of transplants.

Since this model ignores probabilities of failure, it can be used to analyze the impact of including uncertainty in KEPs. As we will see when interpreting the computational results, considering only the deterministic problem (or that all arcs exist) leads to a significant reduction in the actual number of transplants obtained through this model and significantly less actual transplants than in the other approaches.

4.4 Computational Results

Computational tests were conducted on 50 instances of 20, 30, 40 and 50 pairs each, generated by the instances' generator described in [36], taking into account the probabilities of blood type and tissue incompatibility. The probabilities of a positive crossmatch were obtained according to [37]. The model was implemented using C++ and CPLEX on a machine with a Intel Core i7 CPU @ 2.3 GHz and 6 GB RAM.

In this section we will analyze the performance of the optimistic approach, the expectations model and the expectations with backarcs approach, and establish comparisons among them. For all instances, we assume that conducting each test costs 1 and that the budget per instance varies from 2 to the number of vertices.

Optimistic Approach

Figures 4.5 and 4.6 show the results for the optimistic approach, compared to the actual number of transplants obtained after crossmatching the arcs of the solution. Furthermore, we also consider the omniscient solution as a reference for the upper bound.

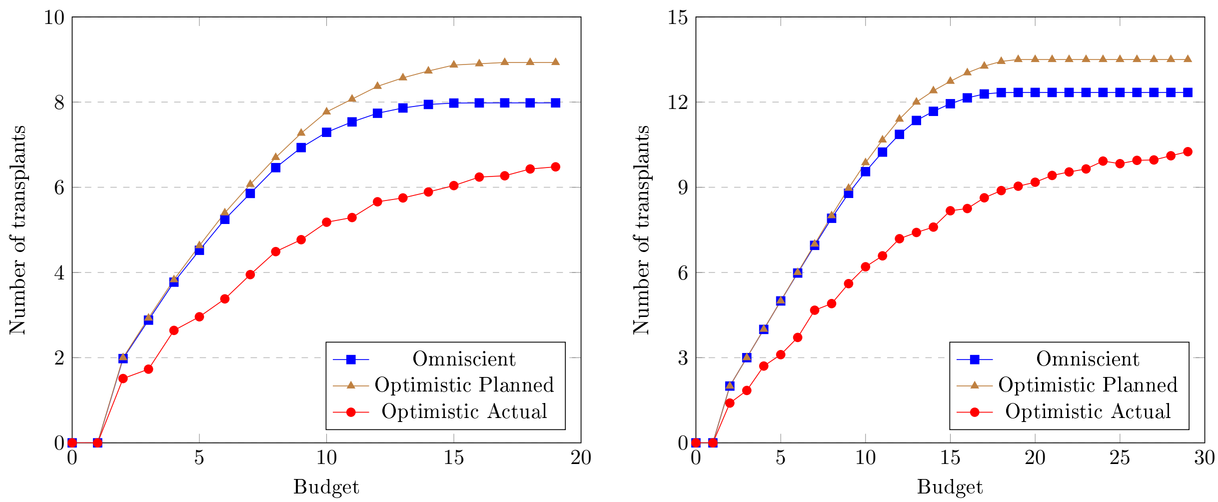

Figure 4.5: Results of the optimistic approach, compared to the crossmatched solution, as well as the omniscient solution for instances of 20 (left) and 30 pairs (right).

We can observe that for instances of 20 pairs and for a budget between 2 and 10, with each unitary increase in the budget, the increase in the number of planned transplants is of, on average, 0.72 transplants. For a budget between 11 and 17, the increase is of 0.14 transplants, and for a budget of 18 or higher, there is no increase in the number of planned transplants (no additional information is gained).

Considering the same ranges for the budget, the increase in the number of actual transplants is of approximately 0.46 transplants and 0.16 transplants, respectively. For a budget of 18 or higher, the number of actual transplants continues to increase, although at a lower rate.

For 30 pairs, for a budget between 2 and 13, there is a gain of approximately 0.91 transplants with each unitary increase in the budget. Between a budget of 14 and 19, the increase is of about 0.22 transplants per one additional test. For a budget of 19 or higher, the number of transplants remains the same.

As for the actual number of transplants, for a budget between 2 and 13, the increase in the number of transplants for each additional test is of 0.55; for a budget between 14 and 19 there is an increase of 0.29

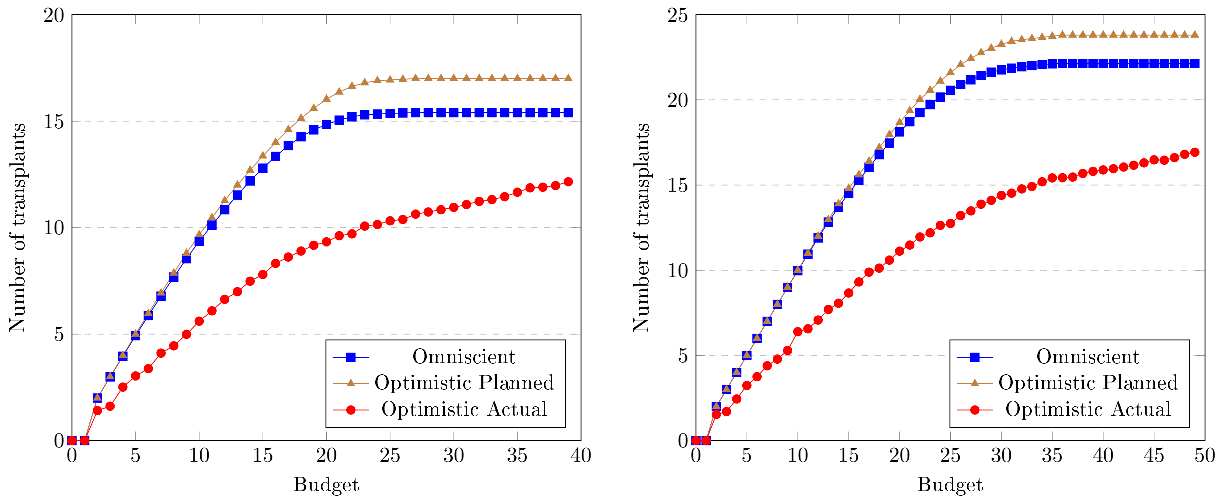

Figure 4.6: Results of the optimistic approach, compared to the crossmatched solution, as well as the omniscient solution for instances of 40 (left) and 50 pairs (right).

transplants; for a budget between 20 and 29, the increase is of 0.12 transplants.

For instances of 40 pairs, for a budget between 2 and 19, there is an increase of about 0.82 transplants with each unitary increase in the budget. Between a budget of 20 and 27, the increase is of about 0.14 transplants and for a budget of 28 or higher, the number of transplants does not change.

Regarding the actual number of transplants, for a budget between 2 and 19, the increase in the number of transplants for each additional test is of 0.46; for a budget between 20 and 27 there is an increase of 0.19 transplants; for a budget between 28 and 39, the increase is of 0.13 transplants.

For instances of 50 pairs and for a budget between 2 and 21, with each unitary increase in the budget, the increase in the number of planned transplants is of, on average, 0.91 transplants. For a budget between 22 and 36, the increase is of 0.27 transplants, and for a budget of 37 or higher, there is no increase in the number of planned transplants (no additional information is gained).

Considering the same ranges for the budget, the increase in the number of actual transplants is of approximately 0.52 transplants and 0.25 transplants, respectively. For a budget of 37 or higher, with each additional test there is an increase of approximately 0.13 actual transplants.

We can conclude that regardless of the size of the instance, on average, the number of transplants increases at a similar rate in relation to the increase in the budget.

For instances of 20 pairs, the average number of transplants obtained without considering failure is of 8.93 transplants for a budget of 19 tests (see table 4.1), about 11.2% higher when compared to the omniscient solution (the maximum possible). However, the number of transplants after crossmatching the arcs of the solution was of 6.48, at least 27.4% below the initial solution. This means that out of the initial selected arcs, at least 27.4% of them end up failing, which supports the idea that this approach leads to a high number of failures and, consequently, to a small number of transplants.

For the remaining instances, similar results were attained, where the failing rate for the optimistic approach is between 24% and 29%.

Budget/Instance Size	Omniscient	Optimistic	Optimistic Actual
20	7.982	8.93	6.48 (-27.4%)
30	12.340	13.5	10.254 (-24%)
40	15.399	17	12.156 (-28.5%)
50	22.133	23.8	16.923 (-28.9%)

Table 4.1: Results of the optimistic approach for instances of $n = 20$, 30, 40 and 50 pairs, for $B = n$, respectively.

Expectations vs Expectations with Backarcs

As for the model with expectations and the expectations with backarcs approach, figures 4.7 and 4.8, contain the results for all instances for a budget between 2 and the number of vertices.

For the expectations approach, we have the following results:

- For instances of 20 pairs, the expectations approach yields an increase of about 0.25 transplants for each unitary increase in the budget. As for the actual number of transplants, there is approximately 0.3 more transplants with each additional budget unit.
- For 30 pairs, there are approximately 0.27 more expected transplants and about 0.33 more actual transplants for a unitary increase in the budget.
- For 40 pairs, with each additional budget unit there is an increase of approximately 0.24 expected transplants and 0.3 actual transplants.
- Regarding instances of 50 pairs, the increase of the budget yields an increase of about 0.3 expected transplants and 0.34 actual transplants.

In regard to the expectations with backarcs approach, the results obtained are described next.

- For instances of 20 pairs, the expectations approach yields an increase of about 0.29 transplants for each unitary increase in the budget. As for the actual number of transplants, there is approximately 0.32 more transplants with each additional budget unit.
- For 30 pairs, there are approximately 0.31 more expected transplants and about 0.32 more actual transplants for a unitary increase in the budget.
- For 40 pairs, with each additional budget unit there is an increase of approximately 0.28 expected transplants and 0.31 actual transplants.
- Regarding instances of 50 pairs, the increase of the budget yields an increase of about 0.33 expected transplants and 0.36 actual transplants.

We can observe that, once again, the increase of the number of transplants in relation to the increase in the number of tests is similar for instances of all sizes.

Moreover, comparing the expectations model with the model with additional arcs for instances of size 20, there is an increase of about 13.7% in the number of transplants from the former (with 6.05 transplants) to the latter (with 6.88 transplants). For the remaining instances, the increase is similar: for 30 pairs, the increase is of 16.4%, for 40 pairs, of 18.6% and for 50 pairs, the increase is of 15%.

For the expectations model with backarcs, the increase in the actual number of the solution when adding more arcs is of approximately 6%, 7%, 9.1% and 7.4% for instances of 20, 30, 40 and 50 pairs, respectively.

Regardless of the instance size, we can also see that the expectations model with backarcs provides better results than the model without backarcs, ranging from an increase of about 1.9% in the size of a solution (for 30 pairs), to about 4.9% (for 50 pairs). Thus, we can conclude that the expectations model with backarcs (with additional arcs) leads to better results when compared to the model without backarcs (with additional arcs, respectively).

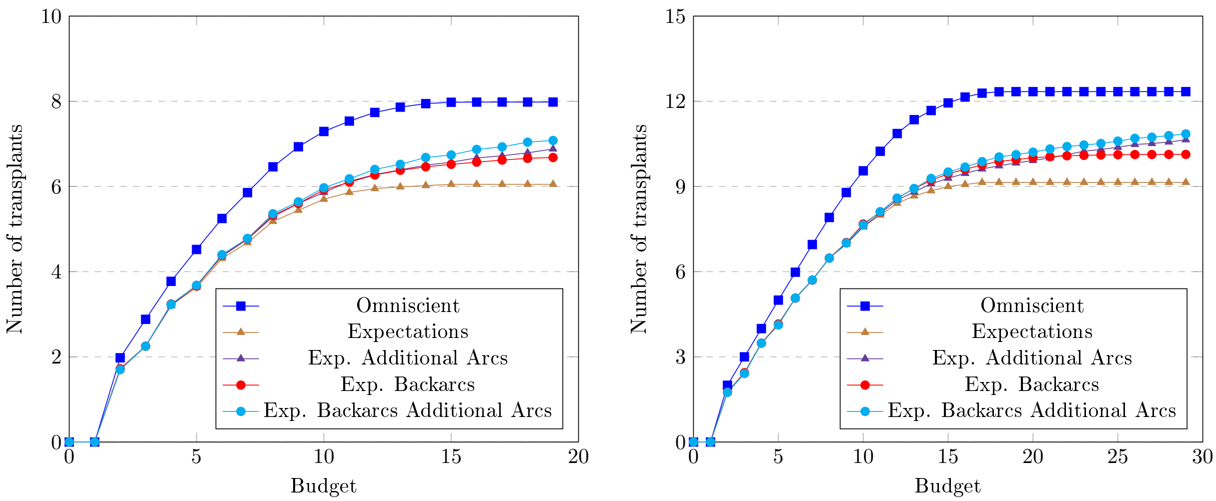

Figure 4.7: Results of the expectations model compared to the model with additional arcs, and the expectations with backarcs model, compared to the same model with additional arcs, as well as the omniscient solution for instances of 20 (left) and 30 pairs (right).

Optimistic vs Expectations Model

As we have concluded with the two previous analyses, although the optimistic approach had a higher number of planned transplants, around 24% of those transplants end up not taking place. This is even more evident when comparing to the expectations model with additional arcs, since the solutions obtained with the latter approach have, on average, approximately 5% more transplants that the actual solutions of the optimistic. Figures 4.9 and 4.10 and table 4.3 show the aforementioned results.

This is helpful in understanding how considering the uncertainty present in Kidney Exchange Programs

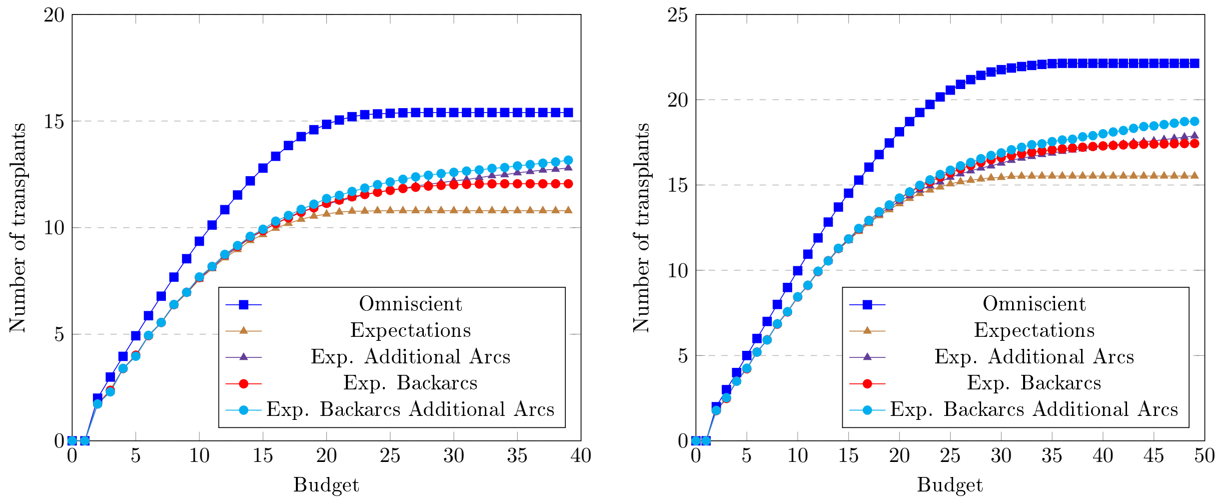

Figure 4.8: Results of the expectations model compared to the model with additional arcs, and the expectations with backarcs model, compared to the same model with additional arcs, as well as the omniscient solution for instances of 40 (left) and 50 pairs (right).

Budget	Exp.	Exp. Additional Arcs	Exp. Backarcs	Exp. Backarcs Additional Arcs
20	6.05	6.88	6.68	7.08
30	9.143	10.644	10.130	10.851
40	10.793	12.8	12.058	13.165
50	15.525	17.876	17.436	18.732

Table 4.2: Results of the expectations approach and the expectations with backarcs for instances of $n = 30$, 40 and 50 pairs, where $B = n$, respectively.

affects the quality and size of the obtained solutions.

Budget	Optimistic Actual	Exp. Backarcs Add. Arcs
20	6.48	6.68
30	10.254	10.644
40	12.156	12.8
50	16.923	17.876

Table 4.3: Comparison between the optimistic approach and the expectations with backarcs approach for instances of $n = 20$, 30, 40 and 50 pairs, where $B = n$, respectively.

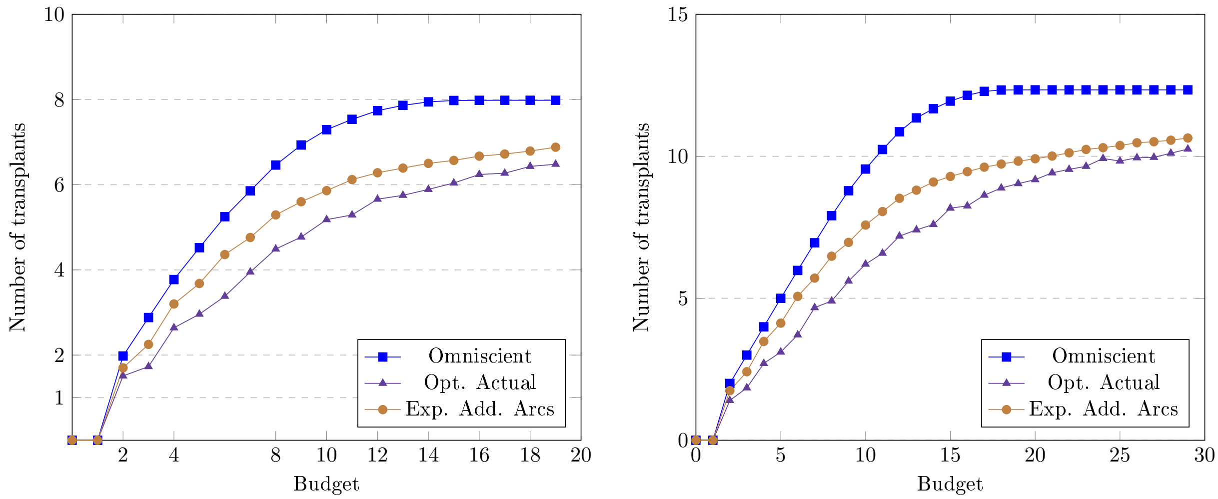

Figure 4.9: Comparison between the optimistic model and the expectations model (with additional arcs), as well as the omniscient solution for instances of 20 and 30 pairs, respectively.

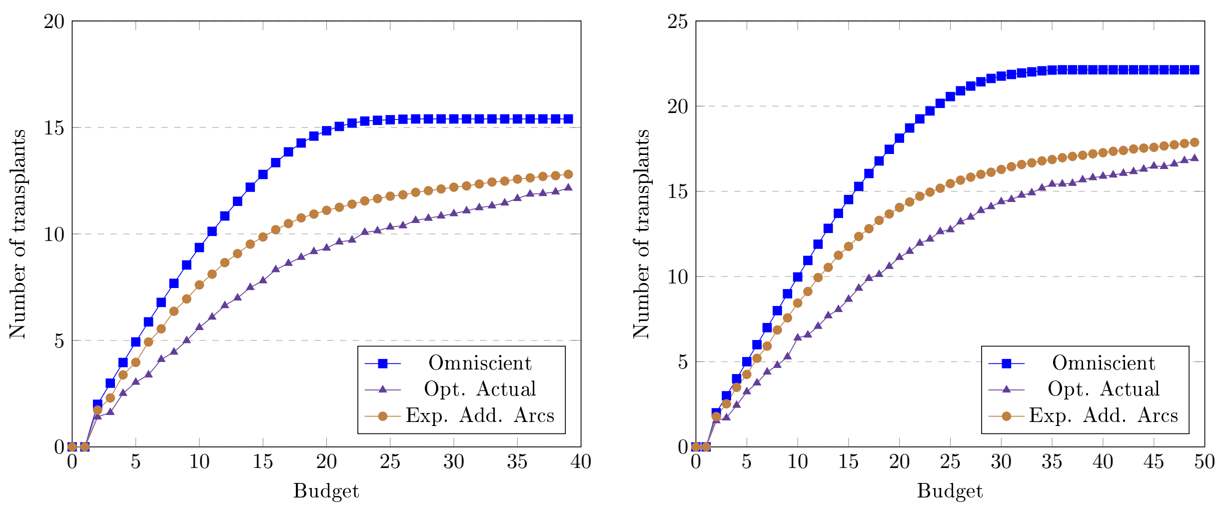

Figure 4.10: Comparison between the optimistic model and the expectations model (with additional arcs), as well as the omniscient solution for instances of 40 and 50 pairs, respectively.

Chapter 5

Unlimited Problem

In this chapter we will describe a second problem which can be studied in the context of Kidney Exchange Programs, which we call the "unlimited" problem.

5.1 Problem Description

The unlimited problem is characterized as having an unlimited budget for testing arcs in an unlimited number of rounds. In each round, the goal is to select the best set of arcs to be tested according to a certain method. We want to reach the omniscient solution in the smallest number of rounds with the smallest budget. Again, we assume that a solution must have all of its arcs tested.

This problem may be characterized as the opposite of the simultaneous problem, as it will be the most unrestricting one.

The framework to be considered is the same as for the simultaneous problem, a directed graph $G = (V, A)$, where V is the set of vertices and A is the set of arcs.

Two types of methods will be studied: the solution-based ones and the probing-based ones. They will be described in the following sections.

5.2 Solution-Based Methods

In this section we will consider five solution-based approaches: optimistic, expectations pure, expectations modified, union pure and union modified. They all start by considering a solution obtained through the optimization of the IP model and proceed from there.

5.2.1 Optimistic

This approach is described in [38] (where it is called "reoptimization").

We first obtain an optimistic solution, i.e., assume that all arcs of the instance exist. In each round, we are going to maximize the number of transplants and crossmatch all arcs of the solution. If any arc fails the test, we restart the process. As such, we will optimize the model until we find a solution for which all arcs exist.

The algorithm for this method is as shown in algorithm 2.

Algorithm 2: Obtain a solution with the optimistic method

```
Optimize the optimistic model and obtain a solution S
Test all arcs in S
if All arcs in S exist then
    Stop. The output is S
else
    Exclude from the network the arcs which do not exist
    go to 1
```

In a given round, if there are several solutions with the same objective value, we select the one which has the highest number of arcs tested in previous rounds.

Given the fact that there is an unlimited budget and an unlimited number of testing rounds, this method converges to the omniscient solution (since, in the worst-case scenario, we can test all arcs of the instance).

5.2.2 Expectations Pure

For this method, we will define a model similar to the one used in Chapter 4. The parameters remain the same:

- p_{ij}: the probability of failure of arc $(i.j) \in A$
- a_{ij}^c: a parameter which is 1 if arc (i, j) belongs to cycle c and 0 otherwise

The decision variables are also maintained.

- $y_{ij} = \begin{cases} 1, & \text{if arc } (i,j) \text{ is tested} \\ 0, & \text{otherwise} \end{cases} \qquad \forall (i,j) \in A$
- $w_c = \begin{cases} 1, & \text{if cycle } c \text{ is selected for the transplantation plan} \\ 0, & \text{otherwise} \end{cases} \qquad \forall c \in C$

In each round, we will use the following integer programming model to make the selection of the arcs:

$$\begin{aligned}
\text{Maximize} \quad & z = \sum_{c \in C} E(c) w_c && (5.1)\\
\text{Subject to:} \quad & \sum_{c \in C} \sum_{(i,j) \in A} a_{ij}^c w_c \leq 1, \quad \forall i \in V && (5.2)\\
& w_c - a_{ij}^c y_{ij} \leq 0, \quad \forall (i,j) \in A, \forall c \in C && (5.3)\\
& w_c, y_{ij} \in \{0,1\}, \forall c \in C, \forall (i,j) \in A && (5.4)
\end{aligned}$$

The objective function (5.1) maximizes the expected number of transplants. Constraints (5.2) ensure that each vertex is in, at most, one selected cycle for the exchange. Constraints (5.3) ensure that only cycles where every arc is tested are selected. Finally, constraints 5.4 define the domain of the decision variables.

After we maximize the expected number of transplants and get a solution, we test all its arcs. If at least one arc does not exist, it is excluded from the network and the model is re-optimized. This procedure is done until we obtain a solution with all tested arcs.

One thing should be remarked about this approach: it does not necessarily converge to the omniscient solution. Since the approach with expectations provides a solution with the most reliable arcs *on average*, it can differ from the omniscient solution.

The algorithm for this method is the same as 2, except in line 1, where the expectations model is optimized instead of the optimistic model. In the next section, we will present an example which illustrates why this method may not converge to the omniscient solution.

5.2.3 Expectations Modified

This method is based on the previous one, but aims at improving the fact that the expectations pure approach may not converge to the omniscient solution.

In this procedure, we start with the solution given by the expectations model and analyze other, possibly better, solutions. We start by optimizing the expectations model, testing all of the arcs of the obtained solution until reaching one for which all arcs exist, S_e. Then, we optimize the optimistic model, obtaining a solution S_o and testing all of its untested arcs. If all arcs exist, we have arrived at an omniscient solution, S_o. Otherwise, we go back to optimizing the optimistic model (see algorithm 3).

On the one hand, this method allows us to search for solutions that are more sturdy and reliable (calculated with the expectations model), and on the other hand, test solutions which are the most promising in terms of the number of transplants (through the optimistic model), in the same iteration. As such, we start with a reliable solution and then move on to a possibly better one by switching methods.

Let us consider an example to illustrate this point. Figure 5.1 shows a KEP instance with four incompatible pairs. We are going to assume all arcs in this instance exist.

The cycles in this instance and their respective expected values are:

- $c_1 = \{(0,2),(2,1),(1,0)\}$ and $E(c_1) = 2.43675$.
- $c_2 = \{(0,1),(1,0)\}$ and $E(c_2) = 1.26$.

Algorithm 3: Expectations modified heuristic

```
1  Get a solution S_e with the expectations model
2  Test untested arcs in S_e
3  Update network
4  if All arcs in S_e exist then
5  |   Re-optimize and get a solution S_o with optimistic model
6  |   if All arcs in S_o exist then
7  |   |   Stop. The solution is S_o
8  |   else
9  |   |   go to 5
10 else
11 |   go to 1
```

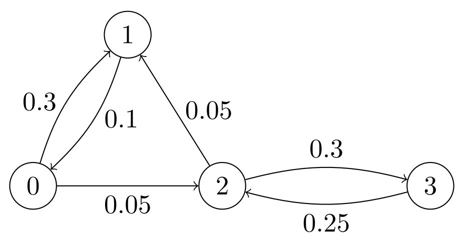

Figure 5.1: Example of a KEP instance with four incompatible pairs and probabilities of failure.

- $c_3 = \{(2,3),(3,2)\}$ and $E(c_2) = 1.05$.

By optimizing the network with the expectations model, we would obtain the solution $S_1 = \{(0,2),(2,1),(1,0)\}$, as it has the highest value out of all the cycles and their combinations (combining cycles c_2 and c_3 yields an expected value of 2.31). However, assuming all arcs in this instance exist, the omniscient solution would be equal to cycles c_2 and c_3 combined, so $S_2 = \{(0,1),(1,0),(2,3),(3,2)\}$.

With the expectations modified approach, after reaching S_1 we would optimize the optimistic model and obtain solution S_2. After testing all of its arcs and confirming they all exist, we could conclude the latter solution is better.

5.2.4 Union Pure

This approach is similar to the previous one, but is meant to speed up the convergence between the expectations solution and the solutions obtained with the optimistic model. The goal is to try the optimistic solution (S_o), and in case at least one of its arcs does not exist, we test the expectations solution (S_e) in the same iteration. Thus, in each iteration, we are testing the most promising solution and the solution which is most likely to not have any failures.

For this, we start by calculating a solution S_e with the expectations model and another solution S_o with the optimistic model. We then test all the arcs of both. If all arcs of solution S_o exist, then we can stop, as we have arrived at the omniscient solution (S_o). Otherwise, we re-optimize both models and repeat the process.

Algorithm 4 summarizes this approach.

Algorithm 4: Union Pure Heuristic

1 Obtain a solution S_e with the expectations model
2 Obtain a solution S_o with the optimistic model
3 Test untested arcs in S_e
4 Test untested arcs in S_o
5 **if** *All arcs in* S_o *exist* **then**
6   **Stop.** The solution is S_o
7 **else**
8   **go to** 1

5.2.5 Union Modified

For this approach, we start again by considering the solutions of both the optimistic model and the expectations model in each iteration. However, instead of testing all arcs of the optimistic solution, we select arbitrarily a number of arcs of this solution to be tested (as described in algorithm 5).

Algorithm 5: Union Modified heuristic

1 Get a solution S_e with the expectations model
2 Get a solution S_o with the optimistic model
3 Test untested arcs in S_e
4 Test $|S_o| - |S_e|$ arcs (chosen arbitrarily) of S_o
5 **if** *All arcs in* S_o *exist* **then**
6   **Stop.** The solution is S_o
7 **else**
8   **go to** 1

5.3 Probing-Based Methods

This section provides a description of three probing-based approaches. First, we are going to define the concept of *expected value with probing*, as was described in [39].

The expected value with probing of an arc is defined as the expected number of transplants of a network if the arc x is selected to be tested.

Let us consider the expected number of transplants if the result of the test on arc x is v (where v is the value of a realization of the tested random variable, which is 0 if arc x exists and 1 otherwise), which we denote by $E(A, x, v)$.

The expected number of transplants with probing for arc x, denoted by $E(A, x)$, is defined as

$$E(A, x) = (1 - p_x)E(A, x, 0) + p_x E(A, x, 1), \quad x \in A, \tag{5.5}$$

where p_x is the probability of failure of arc x. $E(A, x, v)$ is calculated by updating p_x to v and solving for the maximum expected number of transplants.

Example 5.2 illustrates the concept of probing.

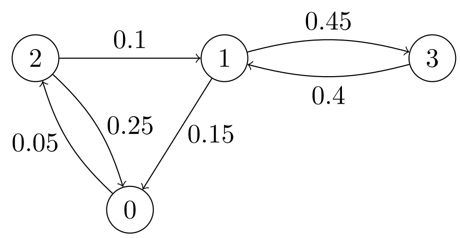

Figure 5.2: Example of a KEP instance with four incompatible pairs and probabilities of failure.

Table 5.1 contains the expected value with probing (EVP) for each arc of the instance in figure 5.2. We can observe that arc $(1, 0)$ has the highest EVP, at 2.493, which is greater than the expected value calculated for any of the cycles of the instance.

x	$E(A, x, 0)$	$E(A, x, 1)$	$E(A, x)$
$(0, 2)$	2.295	0.66	2.21325
$(1, 0)$	2.565	2.085	**2.493**
$(1, 3)$	2.625	2.18025	2.4248625
$(2, 0)$	2.56	2.18025	2.4650625
$(2, 1)$	2.4225	2.085	2.38875
$(3, 1)$	2.525	2.18025	2.3871

Table 5.1: Expected value with probing for each arc of instance in figure 5.2.

This concept can be extended to a set of any number of arcs; however, the number of integer programming models to be solved for each subset of T arcs for an instance with n arcs is 2^T, and a total of $2^T \binom{n}{T}$ for each instance. As this is very computationally heavy even for small instances, we restrict T to 3, since it is usually also the maximum size of a cycle considered in practical applications.

5.3.1 Expected Value with Probing for Arcs

In this method, we are interested in probing the arcs of a solution obtained with the optimistic approach. We start by getting the solution with the highest number of transplants and then calculate the expected value with probing for each one of its arcs (see algorithm 6). In each round, we select the arc with the highest EVP to be tested.

Algorithm 6: Expected Value with Probing for Arcs

```
do
    Obtain a solution S with the optimistic model
    Calculate the expected value with probing for all arcs of S
    Select the arc with the highest EVP and test it
while The solution has untested arcs
```

5.3.2 Expected Value with Probing for Cycles

This method differs from the previous one in the fact that instead of calculating the EVP for each arc of a solution, we calculate it for each cycle. In each round, we conduct the optimization for the optimistic approach (thus obtaining the solution with the maximum number of transplants) and then calculate the EVP for each cycle, selecting the one with the highest EVP. These steps are formalized in algorithm 7.

For instance, for a 2-cycle composed by arcs x and y, the EVP is calculated as follows:

$$\begin{aligned} E(A,x,y) =&(1-p_x)(1-p_y)E(A,x,y,00) + (1-p_x)p_y E(A,x,y,01) + \\ &+ p_x(1-p_y)E(A,x,y,10) + p_x p_y E(A,x,y,11) \quad \forall x,y \in A, \end{aligned} \tag{5.6}$$

where p_x and p_y is the probability of failure of arcs x and y, respectively. $E(A,x,y,v_1v_2)$ is calculated by updating p_x to v_1 and p_y to v_2 and solving for the maximum expected number of transplants.

Algorithm 7: Expected Value with Probing for Cycles

```
do
    Obtain a solution S with the optimistic model
    Calculate the expected value with probing for all cycles of S
    Select the cycle with the highest EVP and test it
while The solution has untested arcs
```

5.3.3 Expected Value with Probing and Alternative Solutions

In this approach, we use the various alternative solutions obtained for the optimistic method. We start by calculating n solutions with the highest number of transplants (where n is the number of pairs of the instance) and for each one, we calculate the EVP for each cycle. Then, after calculating the average EVP per cycle for each solution, we choose the one with the highest average EVP. Finally, we test the untested arcs of this solution and repeat this process until a solution with tested arcs is reached.

Algorithm 8: Expected Value with Probing for Alternative Solutions

```
1 do
2 |   Obtain n solutions with the optimistic model
3 |   Calculate the EVP for every cycle of every solution
4 |   Select the solution which has the highest average EVP to be tested and test all of its arcs
5 while The solution has untested arcs
```

5.4 Computational Results

Computational tests were conducted for 50 instances of 20, 30, 40 and 50 pairs each, obtained as described in Section 4.4 of Chapter 4.

For instances of 20 pairs, we start by analyzing the four convergent solution-based methods. Regarding the number of rounds, as shown in table 5.2, the fastest method is “Union Pure", attaining convergence in 1.842 rounds. The slowest is “Expectation Modified", with 4.797 rounds, which is approximately 2.6 times higher than the former approach.

As for the number of tests, the least expensive method is the “Optimistic", with 13.554 tests. Conversely, the approach which needed the highest number of tests to converge is the “Union Pure", with 15.958, approximately 45% more tests when compared to the Optimistic.

Concerning the probing-based approaches, the fastest one is the “EVP for Alternative Solutions", with 2.178 rounds, and the slowest one is the “EVP for Arcs", needing over 5 times more rounds (11.377) to attain convergence. However, when analyzing the number of tests, the roles for these two methods are reversed: the former is the most expensive, with 13.109 tests, while the latter is the least expensive, with 11.377 tests.

Figure 5.3 contains the visualization of table 5.2. If we consider this problem as a multi-criteria approach with two minimization criteria (number of tests and number of rounds), we can observe that the two most efficient methods are the “Union Pure" and the "EVP for Alternative Solutions".

Regarding execution times, the probing-based approaches are usually slower when compared to the solution-based ones.

Similar results were observed for instances of 30 pairs, which are summarized in table 5.3. Solution-based methods are also faster, on average, when compared to the probing-based approaches, but also more expensive. The “Union Pure" method was again the fastest overall (with 2.317 rounds), with “EVP for Alternative Solutions" being the fastest among the probing-based approaches (with 2.804 rounds). As for the number of tests, “EVP for Arcs" was the approach which needed the least amount of tests to converge overall (with 17.159 tests), whereas the Optimistic method was the least expensive among the solution-based approaches.

	Solution-based					Probing-based		
Average	Exp. Pure*	Opt.	Exp. Mod.	Union Pure	Union Mod.	Arcs	Cycles	Alt. Sol.
Sol. Value	7.601	7.982	7.982	7.982	7.982	7.982	7.982	7.982
No. Rounds	1.791	2.277	4.797	**1.842**	2.547	11.377	4.357	**2.178**
No. Tests	10.997	**13.554**	14.813	15.958	13.673	**11.377**	12.992	13.109

Table 5.2: Computational results for the unlimited problem for instances of 20 pairs. *Exp. Pure does not converge to an omniscient solution.

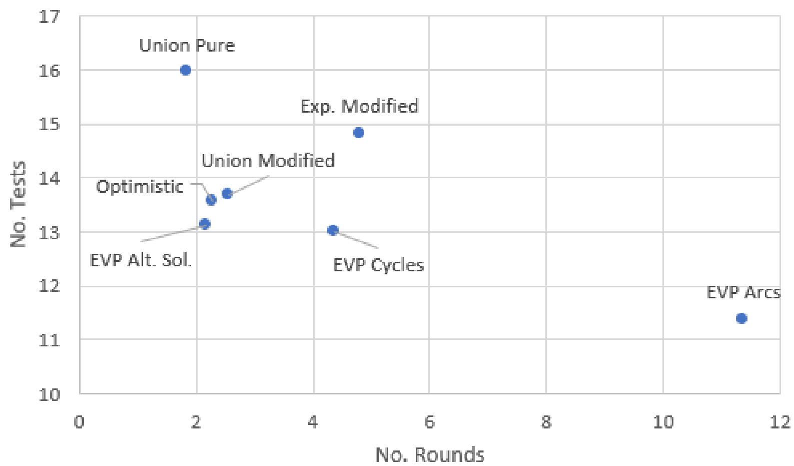

Figure 5.3: Scatter plot comparing the number of rounds and the number of tests for instances of 20 pairs.

In figure 5.4, we can observe that once again, the non-dominated solutions are the Union Pure and the EVP for Alternative Solutions.

	Solution-based					Probing-based		
Average	Exp. Pure	Opt.	Exp. Mod.	Union Pure	Union Mod.	Arcs	Cycles	Alt. Sol.
Sol. Value	11.764	12.340	12.340	12.340	12.340	12.340	12.340	12.340
No. Rounds	2.282	2.934	8.288	**2.317**	3.415	17.159	6.845	**2.804**
No. Tests	17.438	**21.125**	25.097	25.675	21.831	**17.159**	20.020	20.753

Table 5.3: Computational results for the unlimited problem for instances of 30 pairs.

Tables 5.4 and 5.5 provide the results obtained for instances of 40 and 50 pairs, except for approach “EVP for Alternative Solutions", which proved to be too computationally heavy. Again, similar conclusions can be drawn from these results, where the fastest and least expensive approaches are the same as for the

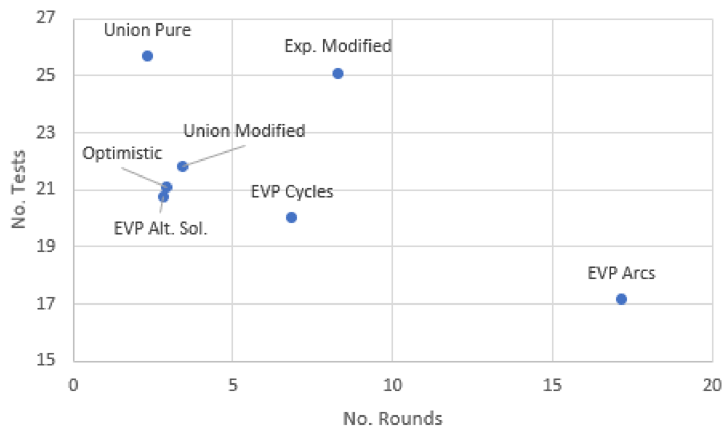

Figure 5.4: Scatter plot comparing the number of rounds and the number of tests for instances of 30 pairs.

previous instances.

It should be noted that for this problem and for instances of 50 pairs, the “Union Pure" approach yields better results when compared to the Reoptimization approach described in [38], as it achieved the optimal solution with less tests and in less rounds.

	Solution-based					Probing-based	
Average	Exp. Pure	Opt. Pure	Exp. Mod.	Union Pure	Union Mod.	Arcs	Cycles
Sol. Value	14.453	15.399	15.399	15.399	15.399	15.399	15.399
No. Rounds	2.626	3.500	13.630	**2.770**	4.325	23.269	9.680
No. Tests	22.723	29.110	37.860	34.727	29.795	**23.269**	27.582

Table 5.4: Computational results for the unlimited problem for instances of 40 pairs.

	Solution-based					Probing-based	
Average	Exp. Pure	Opt.	Exp. Mod.	Union Pure	Union Mod.	Arcs	Cycles
Sol. Value	16.494	17.212	17.212	17.212	17.212	17.212	17.212
No. Rounds	2.496	3.384	14.248	**2.466**	3.94	22.848	9.06
No. Tests	23.62	28.696	38.744	34.756	29.362	**22.848**	26.962

Table 5.5: Computational results for the unlimited problem for instances of 50 pairs.

From these results, we can observe that there is a clear trade-off between speed (number of rounds) and cost (number of tests). According to the resource that is most available to us, we are able to choose the best policy.

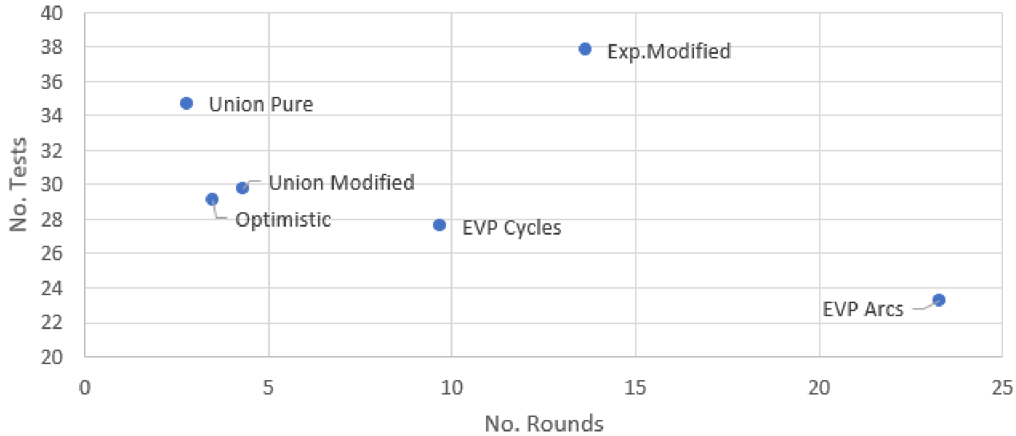

Figure 5.5: Scatter plot comparing the number of rounds and the number of tests for instances of 40 pairs.

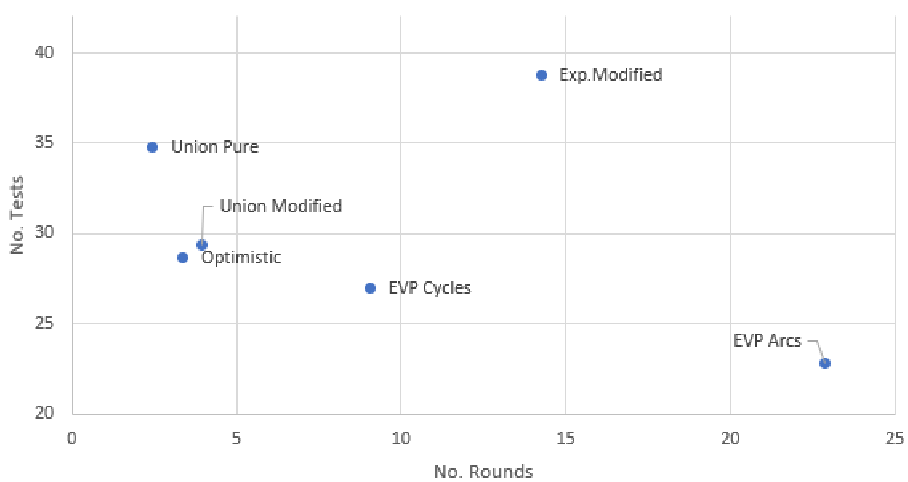

Figure 5.6: Scatter plot comparing the number of rounds and the number of tests for instances of 50 pairs.

Chapter 6

Two-Rounds Problem

The final problem being studied in this dissertation is the two-rounds problem, for which a certain budget and two-rounds will be considered.

6.1 Problem Definition

This problem is a particular case of the unlimited problem presented in Chapter 5 with two rounds. There is also an additional budget constraint for both rounds.

A budget B for the crossmatch tests is considered for the two rounds (combined). In the first round, a set of arcs is selected according to a certain method and is tested at the end of the round. The network is then updated and we proceed to the second round, where again a new set of arcs is selected and is tested. In each round, we are solving a simultaneous problem, as described in Chapter 4.

Since this problem can be interpreted as a "middle-ground" between the simultaneous and the unlimited problem, it will be interesting to compare its results to the previous two. We will be able to understand how many transplants are lost by considering less rounds than in the unlimited problem (and if it is significant) and the gain in comparison to performing all the tests in one round, as is done in the simultaneous problem.

Three methods for the selection of the set of arcs will be presented in the following section: Left, Middle and Right.

6.2 Selection Policies

In each round a set of arcs is to be selected, we need to define which method will be used for this selection. Based on the results obtained in the previous chapters, for the first round, we will use the optimistic approach described in Section 4.3.3 of Chapter 4. Since there are still arcs to be tested in the second round, we can try to get the highest number of arcs in the first round and rearrange the solution in the second round in case of failure. In the second round we will use the expectations approach with backarcs (Section 4.4 of Chapter 4), as there is no possibility of recourse afterwards. Thus, we want to obtain the most reliable set of arcs in this final round.

Since the considered budget must be divided between the two rounds, we will consider three policies in which the budget is distributed differently.

For the “Left" policy, the first round will have approximately 2/3 of the budget allocated to it and will be solved with the optimistic model. After getting a solution with all tested arcs, we will move on to the second round, where the budget is equal to 1/3 of the initial budget and the approach used to calculate the solution will be the expectations with backarcs. The reason for this is to ensure that the obtained solution is as sturdy as possible, considering that the second round is the final one, thus no future repairs can be made to the network.

For the “Middle" policy, the difference is that the budget will be divided (approximately) evenly between the two rounds. Once again, in the first round we use the optimistic approach, whereas in the second, we use the expectations with backarcs and additional arcs model.

For this final policy, we consider the same algorithm as for the left policy, with the difference that the budget is divided 1/3 for the first round and 2/3 for the second round. As such, we are leaving more tests to be performed at a later stage of the program run.

Algorithm 9 described these methods.

Algorithm 9: Two-rounds heuristic

First round: Optimize the optimistic approach for a portion of the budget B (divided according to the selected method) and obtain a set of arcs to test S
Test all arcs in S
Update network
Second round: Optimize the expectations model with backarcs and added arcs for the remaining budget (taking into account some arcs were already tested) and obtain a set of arcs S_e
Test all untested arcs in S_e
Exclude arcs which do not exist from the network
Maximize the number of transplants for the existing arcs

6.3 Computational Results

Computational tests were conducted for 50 instances of 20 and 30 pairs each, generated as described in Section 4.4 of Chapter 4. Figure 6.1 shows the results for each policy for a budget of 2 to 19 for instances of 20 pairs, and for a budget of 2 to 29 to instances of 30 pairs. We can observe that, in general, there are no significant differences among the three policies. Nevertheless, for a smaller budget, the Right policy generally yields better results, while for a bigger budget, the Left and Middle policies tend to be slightly better.

For instances of 20 pairs, the average increase in the number of transplants per unitary increase in the budget is of approximately 0.313 transplants for the Left policy, 0.314 for the Middle Policy and 0.31 for the Right policy.

Regarding instances of 30 pairs, the average increase in the number of transplants with each additional unit in the budget is of about 0.344 transplants for the Left policy, 0.339 for the Middle policy and 0.329 transplants for the Right policy.

This is more evident in figure 6.2 and backed-up by table 6.1.

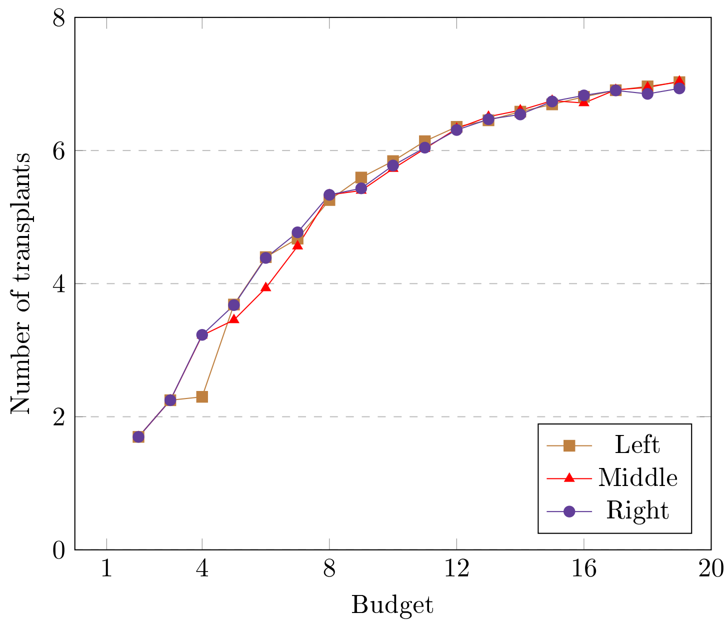

Figure 6.1: Results for instances of 20 pairs.

As for instances of 30, figure 6.2 shows that the left policy was the dominant one, except, once again, for smaller budgets.

No. Pairs/Tests	Left	Middle	Right
20	7.027	7.038	6.933
30	11.044	10.906	10.631

Table 6.1: Average number of transplants for each method, for instances of 20 and 30 pairs.

As previously mentioned, we were also interested in comparing the results of the simultaneous problem with the ones of the current two-round problem.

The best approach in the simultaneous problem was the expectations model with backarcs, which, for instances of 20 pairs with a budget of 20, yielded a solution of 7.08 transplants. Comparing to the two-phased Middle policy which yielded the best result for this budget, with 7.038 transplants, we can conclude that the simultaneous approach provided a better solution.

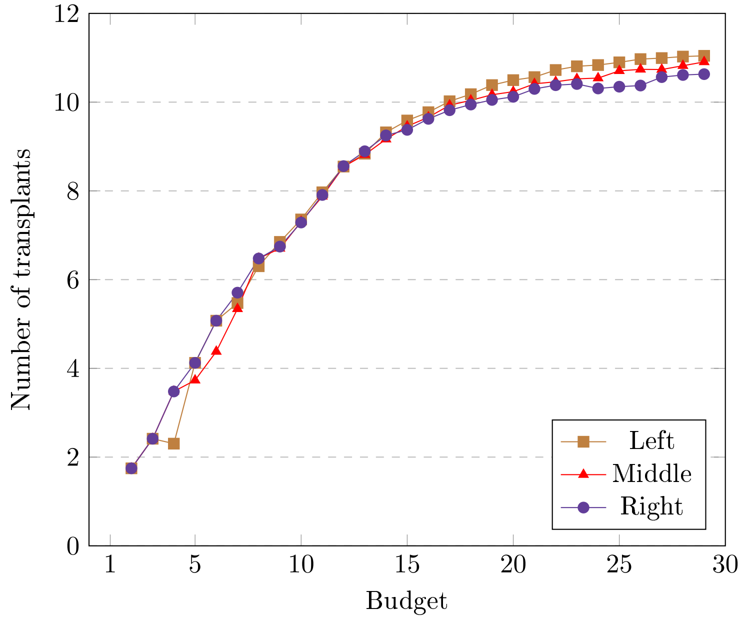

Figure 6.2: Results for instances of 30 pairs.

However, for instances of 30 pairs (considering a budget of 30), we observed that the expectations model with backarcs resulted in 10.851 transplants, whereas the best policy in the two-phased problem was the Left, yielding 11.044 transplants.

Chapter 7

Conclusions

7.1 Discussion

Kidney Exchange Programs play a crucial role in providing more patients suffering from end-stage renal disease with kidney transplants, improving significantly their quality of life. Several optimization approaches have been developed in order to increase the number of transplants being carried out in these programs.

Because of the compatibility issues that may arise in KEPs, the inclusion of probabilities of failure in the optimization techniques leads to better results. Furthermore, for logistical reasons, it is also important to consider the time (in terms of testing rounds) and budget limitations (in terms of the number of the available number of tests) that exist in operating KEPs.

This dissertation focused on the study of three different problems in the context of KEPs: the simultaneous problem (chapter 4), the most restricting one, where there is both a budget and time constraint; the unlimited problem (chapter 5), where there are an unlimited budget and time; finally, the two-rounds problem (chapter 6), where there is a budget and two testing rounds.

In the simultaneous problem, we studied both the deterministic and the probabilistic approach, as well as a method involving backarcs. We were able to conclude that the probabilistic approach with backarcs yielded the best results, providing, on average, a solution with approximately 7% more transplants than the second best approach, for all instances.

Regarding the unlimited problem, where we assume there is no budget or number of rounds limitation, two types of methods were studied: the solution-based and the probing-based approaches. Despite the fact that there is a trade-off between the number of rounds and number of tests needed for each method (so the fastest methods are usually the most expensive ones), we concluded that the solution-based methods tend to require less rounds, whereas the probing-based methods need a lower budget when compared to the solution-based ones. For this problem, we improved the best results from the literature as the best solution-based method leads to less tests and less rounds than the one from [38].

Finally, for the two-rounds problem, we considered an optimistic (deterministic) approach for the first round and an expectations (probabilistic) approach for the second with backarcs. The budget was divided between the two rounds according to three policies: Left (more tests in the first round), Middle (divided equally) and Right (more tests in the second round). The computational tests showed that, on average, Right policy yielded better results for small budgets, whereas the Left and Middle policies lead to the best outcomes for bigger budgets. Comparing these approaches to the ones studied in the simultaneous problem, we concluded that for instances of 30 pairs and a budget of 30, the Left policy performed better when compared to the Expectations Model with Backarcs.

7.2 Future Work

As end-stage renal disease is an issue with great social impact, new methods are being continuously tested and developed in hopes of improving the current system, including in the context of Kidney Exchange Programs.

Regarding our contribution specifically, there are methods that can be further explored. One of them is the number of alternative solutions tested in the "EVP for Alternative Solutions", described in Section 5.3.3 of Chapter 5. By considering more alternative solutions, we could potentially increase the objective function value.

Furthermore, the approaches studied in chapter 6 could also have a potential of providing better results than the ones obtained by trying different methods for the selection of the tests in the first round. With a different method, the set of selected arcs could be more reliable or lead to a higher number of transplants.

Bibliography

[1] P. Biró, L. Burnapp, B. Haase, A. Hemke, R. Johnson, J. van de Klundert, D. Manlove, G. Böhmig, G. Bond, D. Kuypers, B. Smeulders, F. Spieksma, J. Fronek, M. Salonen, C. Jacquelinet, M.-A. Macher, G. Menoudakou, E. I. Ásgeirsson, N. Wilson, P. Di Ciaccio, V. Sparacino, R. Kieszek, C. Bolotinha, T. Baltesová, K. Cechlárová, M. d. l. O. V. Munoz, T. Andersson, K. Hadaya, and R. Leishman. Kidney exchange practices in europe: First handbook of the cost action CA15210: European network for collaboration on kidney exchange programmes (ENCKEP), 2017.

[2] Felix Rapaport. The case for a living emotionally related international kidney donor exchange registry. In *Transplantation proceedings*, volume 18, 1986.

[3] Maaz Ahmed Abbasi, Glenn M Chertow, and Yoshio N Hall. End-stage renal disease. *BMJ clinical evidence*, 2010, 2010.

[4] John Hopkins Medicine. End stage renal disease (esrd). `https://www.hopkinsmedicine.org/health/conditions-and-diseases/end-stage-renal-failure`. Accessed: 2019-07-11.

[5] Andreas Laupacis, Paul Keown, Nancy Pus, Hans Krueger, Beryl Ferguson, Cindy Wong, and Norman Muirhead. A study of the quality of life and cost-utility of renal transplantation. *Kidney international*, 50(1):235–242, 1996.

[6] C. Bradley Wallis, Kannan P. Samy, Alvin E. Roth, and Michael A. Rees. Kidney paired donation. *Nephrology Dialysis Transplantation*, 26(7):2091–2099, 03 2011.

[7] Mohammed A Kaballo, Mark Canney, Patrick O'Kelly, Yvonne Williams, Conall M O'Seaghdha, and Peter J Conlon. A comparative analysis of survival of patients on dialysis and after kidney transplantation. *Clinical Kidney Journal*, 11(3):389–393, 10 2017.

[8] European Renal Care Providers Association. Facts and figures. `http://ercpa.eu/index.php/facts-figures/`. Accessed: 2019-06-20.

[9] James Heaf. Current trends in European renal epidemiology. *Clinical Kidney Journal*, 10(2):149–153, 02 2017.

[10] Connie L Davis and Francis L Delmonico. Living-donor kidney transplantation: a review of the current practices for the live donor. *Journal of the American Society of Nephrology*, 16(7):2098–2110, 2005.

[11] E. Nemati, B. Einollahi, M. Lesan Pezeshki, V. Porfarziani, and M. Reza Fattahi. Does kidney transplantation with deceased or living donor affect graft survival? *Nephro-Urology Monthly*, 6(4):e12182, 2014.

[12] Dorry L Segev, Sommer E Gentry, Daniel S Warren, Brigitte Reeb, and Robert A Montgomery. Kidney paired donation and optimizing the use of live donor organs. *Jama*, 293(15):1883–1890, 2005.

[13] Alvin E Roth, Tayfun Sönmez, and M Utku Ünver. Efficient kidney exchange: Coincidence of wants in markets with compatibility-based preferences. *American Economic Review*, 97(3):828–851, 2007.

[14] David Manlove and Gregg O'Malley. Paired and altruistic kidney donation in the uk: algorithms and experimentation. *Lecture Notes in Computer Science*, 7276:271–282, 2012.

[15] S. E. Gentry, D. L. Segev, M. Simmerling, and R. A. Montgomery. Expanding kidney paired donation through participation by compatible pairs. *American Journal of Transplantation*, 7(10):2361–2370, October 2007.

[16] Kristiaan Glorie. *Clearing Barter Exchange Markets: Kidney Exchange and Beyond*. PhD thesis, Erasmus Research Institute of Management, Erasmus University Rotterdam, 2014.

[17] European Commission. Journalist workshop on organ donation and transplantation: Recent facts & figures. `https://ec.europa.eu/health//sites/health/files/blood_tissues_organs/docs/ev_20141126_factsfigures_en.pdf`. Accessed: 2019-06-20.

[18] National Health Service Blood and Transplant. Uk's living kidney sharing scheme to hit 1000th transplants milestone this week. `https://www.organdonation.nhs.uk/get-involved/news/uk-s-living-kidney-sharing-scheme-to-hit-1000th-transplants-milestone-this-week/`. Accessed: 2019-06-20.

[19] Portal da Diálise. Programa nacional de doação renal cruzada. `https://www.portaldadialise.com/articles/programa-nacional-de-doacao-renal-cruzada-pndrc`. Accessed: 2019-06-20.

[20] Linda Cantwell, Claudia Woodroffe, Rhonda Holdsworth, and Paolo Ferrari. Four years of experience with the australian kidney paired donation programme. *Nephrology*, 20(3):124–131, February 2015.

[21] Shafi Malik and Edward Cole. Foundations and principles of the canadian living donor paired exchange program. *Canadian Journal of Kidney Health and Disease*, 1:6, May 2014.

[22] Kiil Park, Jang Il Moon, Soon Il Kim, and Yu Seun Kim. Exchange donor program in kidney transplantation. *Transplantation*, 67(2):336–338, 1999.

[23] Georg A. Böhmig, Jiří Fronek, Antonij Slavcev, Gottfried F. Fischer, Gabriela Berlakovich, and Ondrej Viklicky. Czech-austrian kidney paired donation: first european cross-border living donor kidney exchange. *Transplant International*, 30(6):638–639, April 2017.

[24] European Kidney Health Alliance. Italy and spain achieve first international cross-kidney transplantation. `http://ekha.eu/blog/italy-and-spain-achieve-first-international-cross-kidney-transplantation/`. Accessed: 2019-06-20.

[25] Filipe Alvelos and Ana Viana. Kidney exchange programs with a priori crossmatch probing. In Natalia Kliewer, Jan Fabian Ehmke, and Ralf Borndörfer, editors, *Operations Research Proceedings 2017*, pages 363–368, Cham, 2018. Springer International Publishing.

[26] Xenia Klimentova, João Pedro Pedroso, and Ana Viana. Maximising expectation of the number of transplants in kidney exchange programmes. *Computers & Operations Research*, 73:1–11, 2016.

[27] Miguel Constantino, Xenia Klimentova, Ana Viana, and Abdur Rais. New insights on integer-programming models for the kidney exchange problem. *European Journal of Operational Research*, 231(1):57–68, 2013.

[28] Miquel Bofill, Marcos Calderón, Francesc Castro, Esteve Del Acebo, Pablo Delgado, Marc Garcia, Marta García, Marc Roig, María O Valentín, and Mateu Villaret. The spanish kidney exchange model: study of computation-based alternatives to the current procedure. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 272–277. Springer, 2017.

[29] Filipe Alvelos, Xenia Klimentova, Abdur Rais, and Ana Viana. A compact formulation for maximizing the expected number of transplants in kidney exchange programs. *Journal of Physics: Conference Series*, 616:012011, May 2015.

[30] David J Abraham, Avrim Blum, and Tuomas Sandholm. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *Proceedings of the 8th ACM conference on Electronic commerce*, pages 295–304. ACM, 2007.

[31] John P Dickerson, David F Manlove, Benjamin Plaut, Tuomas Sandholm, and James Trimble. Position-indexed formulations for kidney exchange. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 25–42. ACM, 2016.

[32] Joao Pedro Pedroso. Maximizing expectation on vertex-disjoint cycle packing. In *International Conference on Computational Science and Its Applications*, pages 32–46. Springer, 2014.

[33] John P Dickerson, Ariel D Procaccia, and Tuomas Sandholm. Failure-aware kidney exchange. In *Proceedings of the fourteenth ACM conference on Electronic commerce*, pages 323–340. ACM, 2013.

[34] Valentin Bartier, Bart Smeulders, Yves Crama, and Frits CR Spieksma. Recourse in kidney exchange programs. 2019.

[35] Kristiaan Glorie, Margarida Carvalho, Miguel Constantino, Paul Bouman, and Ana Viana. Robust models for the kidney exchange problem. Technical report, Working paper, 2015.

[36] Susan L Saidman, Alvin E Roth, Tayfun Sönmez, M Utku Ünver, and Francis L Delmonico. Increasing the opportunity of live kidney donation by matching for two-and three-way exchanges. *Transplantation*, 81(5):773–782, 2006.

[37] Kristiaan Glorie. Estimating the probability of positive crossmatch after negative virtual crossmatch. Technical report, Econometric Institute, Erasmus University Rotterdam, 2012.

[38] Márton Dücső. Testing re-optimisation strategies in kidney exchange programmes. unpublished, 2018.

[39] Filipe Alvelos, Ana Viana, and Xenia Klimentova. Probing for maximizing the expected number of transplants. In *XIX CLAIO, Latin-Iberoamerican Conference on Operations Research Proceedings. September 24-27*, Lima, Perú, 2018.

[40] American Kidney Fund. Kidney disease: Kidney failure/esrd. `http://www.kidneyfund.org/kidney-disease/#kidney_failure_esrd`. Accessed: 2019-03-14.

[41] Robert A Wolfe, Valarie B Ashby, Edgar L Milford, Akinlolu O Ojo, Robert E Ettenger, Lawrence YC Agodoa, Philip J Held, and Friedrich K Port. Comparison of mortality in all patients on dialysis, patients on dialysis awaiting transplantation, and recipients of a first cadaveric transplant. *New England Journal of Medicine*, 341(23):1725–1730, 1999.

[42] David W Gjertson and J Michael Cecka. Living unrelated donor kidney transplantation. *Kidney international*, 58(2):491–499, 2000.

[43] Kiil Park, Jang Il Moon, Soon Il Kim, and Yu Seun Kim. Exchange donor program in kidney transplantation1. *Transplantation*, 67(2):336–338, 1999.