Pairwise Living Donor Kidney Exchange with Patient-Donor Preferences

---

Paarweiser Nieren-Austausch von lebenden Spendern mit Patient-Spender Präferenzen

von

Christian Glaser

Masterarbeit in Wirtschaftswissenschaften

vorgelegt der
Fakultät für Wirtschaftswissenschaften
der
Rheinisch-Westfälischen Technischen Hochschule Aachen

im Februar 2016

angefertigt am

Lehrstuhl für Management Science

Erstgutachter und Betreuer

Prof. Dr. Britta Peis
Lehrstuhl für Management Science
RWTH Aachen University

Zweitgutachter

Prof. Dr. Marco Lübbecke
Lehrstuhl für Operations Research
RWTH Aachen University

Contents

1 Introduction

Kidneys are by far the organ with the largest demand for transplants. In Germany roughly 8,000 patients are on the waiting list for a kidney donation [1]. Although roughly 2,500 kidney transplants per year took place between 2009 and 2014 [2], the number of patients on the waiting list remained the same and the average waiting time is 5 to 6 years [3]. In other countries the situation is even worse. Although the United States has only four times more inhabitants than Germany, more than 100,000 people are waiting for a kidney donation. In 2015, even 3,971 patients died waiting or have been removed from the waiting list because their health severely deteriorated [4].

A kidney is a very special organ. Humans have two kidneys, but one is sufficient to filter waste and remove excess fluid from the body. When one kidney fails due to disease, there is a high probability that the other kidney fails as well. Furthermore, advances in kidney retrieval surgery have resulted in that operations are almost always successful. Research showed that the risk of death from kidney retrieval surgery was 3 in 10,000 and that people with one kidney live as long as those with two [5]. So each healthy person could donate one of his kidneys without limitation of their future life.

Many patients find a relative, e.g., their spouse, who is willing to donate one of their kidneys but that alone is not enough. The donor kidney must be compatible with the patient, i.e., the blood type of patient and donor must be compatible and the patient's immune system must not reject the donor kidney. What exactly is considered as compatible differs from country to country, depending on the assessment of the attending doctors and the additional medical treatment a patient has to undergo before and after the transplantation.

In general one can say that the blood type according to the AB0 system should be compatible and that patient and donor should have a negative cross-match, i.e., the patient does not have antibodies against the donor's tissue or blood. However, these criteria are not valid anymore due to advances in immunosuppression treatment [6]. In Germany, 23% of the living donor kidney transplants were AB0 incompatible in 2014 [7].

Another criterion besides the blood type is the tissue type, that is the antigens of the cells which are categorized in the HLA (human leukocyte antigen) system. The antigens can be grouped in six main categories which have thousands of characteristics each. Thus, a perfect match of tissue type between patient and donor is basically impossible. As the tissue type is mostly hereditary, a good match between direct relatives is more probable.

However, the importance of a good HLA match differs from country to country. The European view is that the outcome of kidney transplants from unrelated live donors is strongly influ-

enced by the HLA compatibility [8] whereas a study from the US claims that the graft survival probability is not dependent on the closeness of the tissue type [9].

Independent of how strict a donor kidney is selected, there will always be a lack of compatible donors. An interesting and promising option to increase living donor kidney transplantations is the pairwise kidney exchange. Consider the following example: the pairing of a patient and a willing but incompatible donor. An exchange could be arranged with another incompatible patient-donor pair so that the donor from each pair gives his kidney to the patient of the other pair if the two pairs are crosswise compatible, i.e., the donor of patient A is compatible to patient B and the donor of patient B is compatible to patient A.

In principle, one can perform also three or four way exchanges or even arbitrarily longer circles. However, there is a strong ethical reason why this is disfavored. One can not make a contract that forces a person to donate his kidney. So after patient A got a kidney from the donor of patient B, the donor of patient A has no incentive to donate his kidney anymore because his relative or friend has already received a new kidney. Therefore, the operations are done simultaneously. In a pairwise exchange, four simultaneous operations with four complete surgical teams are already at the upper end of the logistical feasibility.

The legal situation differs from country to country. In all countries, except Iran, organ trade is forbidden. In the past, the pairwise kidney exchange was considered a money's worth trade and was therefore illegal in most countries. However, the situation is changing globally. In the USA, the national organ transplant act of 1984 clarified that paired exchanges do not violate federal laws against selling organs [5] and the United Kingdom recently passed a law that made kidney exchanges legal.

In Germany, the legal situation is more strict. In principle, a kidney donation is only allowed from a closely related person. This basically forbids pairwise exchanges as the two pairs do not know each other before they participate in a kidney exchange program. However, a ruling of the Federal Social Court from 2003 stated that it may be performed in exceptional cases. Both pairs have to become acquainted with each other and a psychologist has to certify the existence of an adequately intense and firm relationship. The fact that the two pairs have first met only for a kidney transplantation is not an argument against a close relationship [10, 11]. Thus in practice, living-donor kidney exchanges are also possible in Germany.

Another aspect that has to be considered in a kidney exchange program is that the treatment of patients and the registration of potential living donors is done by individual hospitals. Hospitals tend to first search for a compatible pair among their own patients before they widen their search and communicate it to the outside. Therefore, mainly "difficult-to-match-patients" participate in a wider exchange program which worsens the situation for all kidney patients. So one important demand for an algorithm to find compatible pairs is to be incentive compatible. So, a patient can only profit by providing more information. In contrast, by not revealing all his possible donors or compatibilities, the chance of getting a suitable match decreases.

In this thesis, I will concentrate on an algorithm for pairwise kidney exchanges. In the first chapter I will give a general introduction into matching algorithms and apply this concept to pairwise kidney exchanges in the next chapter. I will present a prescription to select compatible patient-donor pairs that takes into account the priority of the patients. Then, this prescription will be extended to also include patient-donor preferences. Finally, I will determine the benefit of an

exchange pool in a case study using simulated patient-donor data sets.

2 Matching Algorithm

The problem of finding a maximum number of pairwise compatible patient-donor pairs out of a pool of incompatible pairs is not trivial. In a brute-force approach all possible combinations between all pairs would need to be considered, which is only possible for a very small number of pairs.

However, more efficient algorithms for this problem already exist - so called matching algorithms. The problem can be modeled via a graph where the nodes are the patient-donor pairs and two nodes are connected with an edge if the two pairs are compatible to each other. A matching is then a set of edges without common vertices, i.e., each pair can only be connected to exactly one other pair. The so called *blossom algorithm* by Edmonds [12] is an efficient algorithm to find a maximum matching. We will not give an exact formal description of this algorithms as it is explicitly described in most textbooks (see [13] for example) but highlight its fundamental ideas and discuss certain steps in more detail if relevant for this thesis.

The algorithm is based on the following Lemma.

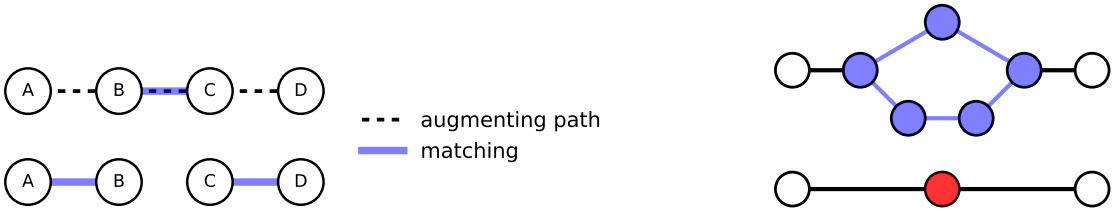
Lemma 1 (Berge's Lemma)**.** *A matching M in a graph $G = (V, E)$[1] is maximum if and only if there is no M-augmenting path [14].*

In an *alternating path* the edges belong alternatively to the matching and not to the matching. An *augmenting path* is an alternating path that starts and ends on free (i.e. unmatched) nodes. Fig. 2.1a illustrates that given an M-augmenting path in a graph G, the matching size $|M|$ can be increased by one edge by inverting the matching along the augmenting path. So every edge on the augmenting path that belongs to the matching is removed and every edge that does not belong to the matching is added to it. By finding all augmenting-paths a maximum matching can be computed.

Using Berge's Lemma, the basic idea for a matching algorithm is to start with any unmatched node, loop through the graph until an augmenting path is found and to use this path to increase the matching size by one. This is repeated until no further augmenting path can be found. The problem of finding a maximum matching is so transformed to the problem of efficiently finding all augmenting paths. The main challenge here is to deal with circuits in the graph.

Edmond's idea was to efficiently detect odd alternating circuits, which he called *blossoms*, and to shrink them into a new pseudo node (see Fig. 2.1b). Then, the matching algorithm can run on the shrunken graph. When all augmenting paths on the shrunken graph are found, the pseudo

[1] In graph theory $G = (V, E)$ denotes a graph G with a set of vertices or nodes V and a set of edges E. A matching M is a subset of the edges E without common vertices.



(a) Illustration of the usage of an augmenting path to increase the matching size. The upper graph shows the matching before and the lower graph shows the matching after the augmentation step.

(b) Shrinking of odd circuits: The odd circuit in the upper graph (highlighted in blue) is shrunk into one pseudo node (highlighted in red).

Figure 2.1: Examples of the augmentation and shrinking steps of the blossom algorithm.

nodes can be expanded again. As only odd circuits are shrunk, it is trivial to select the edges in the odd circuit that belong to the matching. So a maximum matching of the shrunken graph can be extended to a maximum matching of the initial graph.

The algorithm terminates after $\mathcal{O}(n^2)$ steps where n is the number of nodes. In more detail, it terminates after $\mathcal{O}(n)$ augmentations, $\mathcal{O}(n^2)$ shrinking, and $\mathcal{O}(n^2)$ tree extension steps [13]. Improvements in the implementation reduced the run-time to $\mathcal{O}(nm \log n)$, where m is the number of edges.

2.1 Gallai-Edmonds Decomposition

Before describing the algorithm in more detail, we discuss the *Gallai-Edmonds decomposition* of a maximum matching which is a direct outcome of the *blossom algorithm*. The graph can be decomposed into three disjoint sets of nodes. See Fig. 2.2 for an illustration.

1. **Inessential vertices**: This set is a set of odd sets in which all but one node can be matched by any maximum matching within the odd set. The remaining node can only be matched with a node from the set of *essential vertices*. Please note that an odd set can also consist of just one single node. In the context of kidney exchange, this set is denoted as *underdemanded patients* N^U.
2. **Neighbors of inessential vertices**: This is a set of nodes that are part of every maximum matching. The number of nodes in this set limits the matching size, i.e., if this set would consist of one more node, then also the matching size would increase by one. Each node of this set is matched with a node from the set of *inessential vertices*. In the context of kidney exchange, this set is denoted as *overdemanded patients* N^O.
3. **Rest**: This set contains all remaining nodes and consists of even sets of nodes that can be completely matched within each even set by any maximum matching. There is no connection between this set and the rest of the graph in a maximum matching. In the context of kidney exchange, this set is denoted as *perfectly matched patients* N^P.

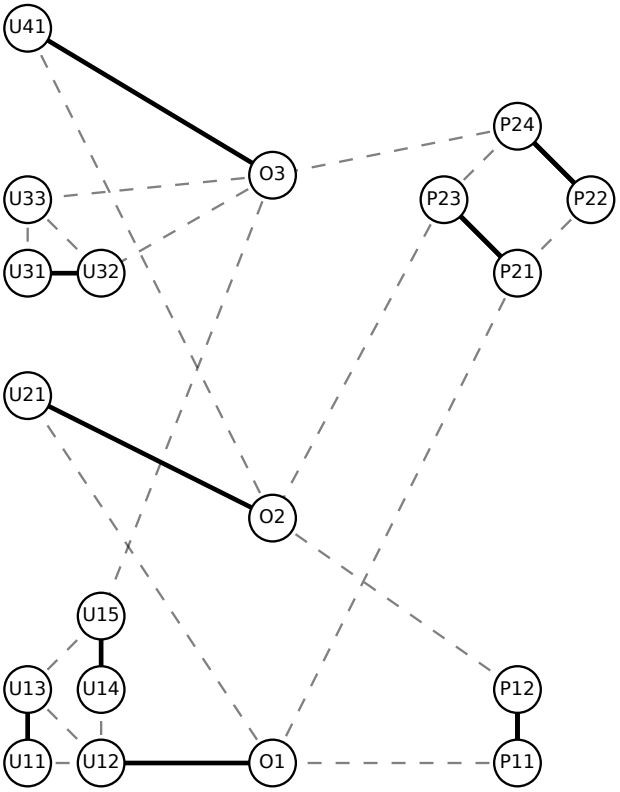


Figure 2.2: Graph to illustrate the *Gallai-Edmonds decomposition*. The dashed lines are the edges of the graph and the solid lines are the edges that belong to the maximum matching. The layout of the graph already highlights the *Gallai-Edmonds decomposition*. The left group of nodes labeled with "U" are the *underdemanded patients*, the nodes in the middle labeled with "O" are the *overdemanded patients* and the nodes on the right labeled with "P" are the rest, the *perfectly matched patients*.

One essential part of the *blossom algorithm* is needed if the *Gallai-Edmonds decomposition* is determined from an already existing maximum matching. We will use the graph of Fig. 2.2 to illustrate the process. The following description is primarily based on the textbook by Cook et al. [13]. Using a matching M of a graph $G = (V, E)$ and an M-exposed node r, i.e., a node that does not belong to the matching M, we build up an alternating tree that consists of sets A and B of nodes, such that each node in A is the other end of an odd-length M-alternating path beginning at r, and each node in B is the other end of an even-length M-alternating path beginning at r (see Fig. 2.3).

The alternating tree T can be constructed using the following prescription: Starting with an unmatched node r and setting $A = \emptyset$ and $B = \{r\}$ the sets A and B can be build up with the following rule:

$$\text{If } vw \in E,\ v \in B,\ w \notin A \cup B,\ wz \in M, \text{ then add } w \text{ to } A,\ z \text{ to } B. \tag{2.1}$$

Fig. 2.3 shows the alternating tree T that is build up from the graph in Fig. 2.2 if started

with node $U33$. The idea behind this structure is that if we find an edge vw such that $v \in B$ and $w \notin A \cup B$, then the M-alternating path from r to v together with vw is an M-augmenting path. However, as we already start with a maximum matching M, we will not find any M-augmenting path but we can explore the *shrinking of odd circuits*, another essential part of the *blossom algorithm*, that will result in the *Gallai-Edmonds decomposition*.

Odd circuits can be detected if an edge vw with $v, w \in B$ is found. Then the path in the alternating tree T from v to w together with the edge vw forms an odd circuit that can be shrunk to a new pseudo node. In our example this is the edge $U33$ to $U32$ and the corresponding path in T ($U33 \rightarrow U31 \rightarrow U32$). This odd circuit can be replaced by a new pseudo node $S1$ resulting in a derived graph G' and which is then used for starting over again.

Fig. 2.3b-d shows the remaining shrinking steps. The *Gallai-Edmonds decomposition* is obtained by identifying the *underdemanded patients* with B and the *overdemanded patients* with A. This process (the creation and shrinking of alternating trees) is repeated until all M-exposed nodes have been considered. All nodes not in A or B belong to the third part of the *Gallai-Edmonds decomposition*.

All calculations in this thesis are implemented using the programming language *Python* [15]. We use the package *Network X* [16] to model graphs and to compute a maximum (weight) matching. For the determination of the *Gallai-Edmonds decomposition* we use our own implementation. All plots have been generated using the *matplotlib* [17] package.

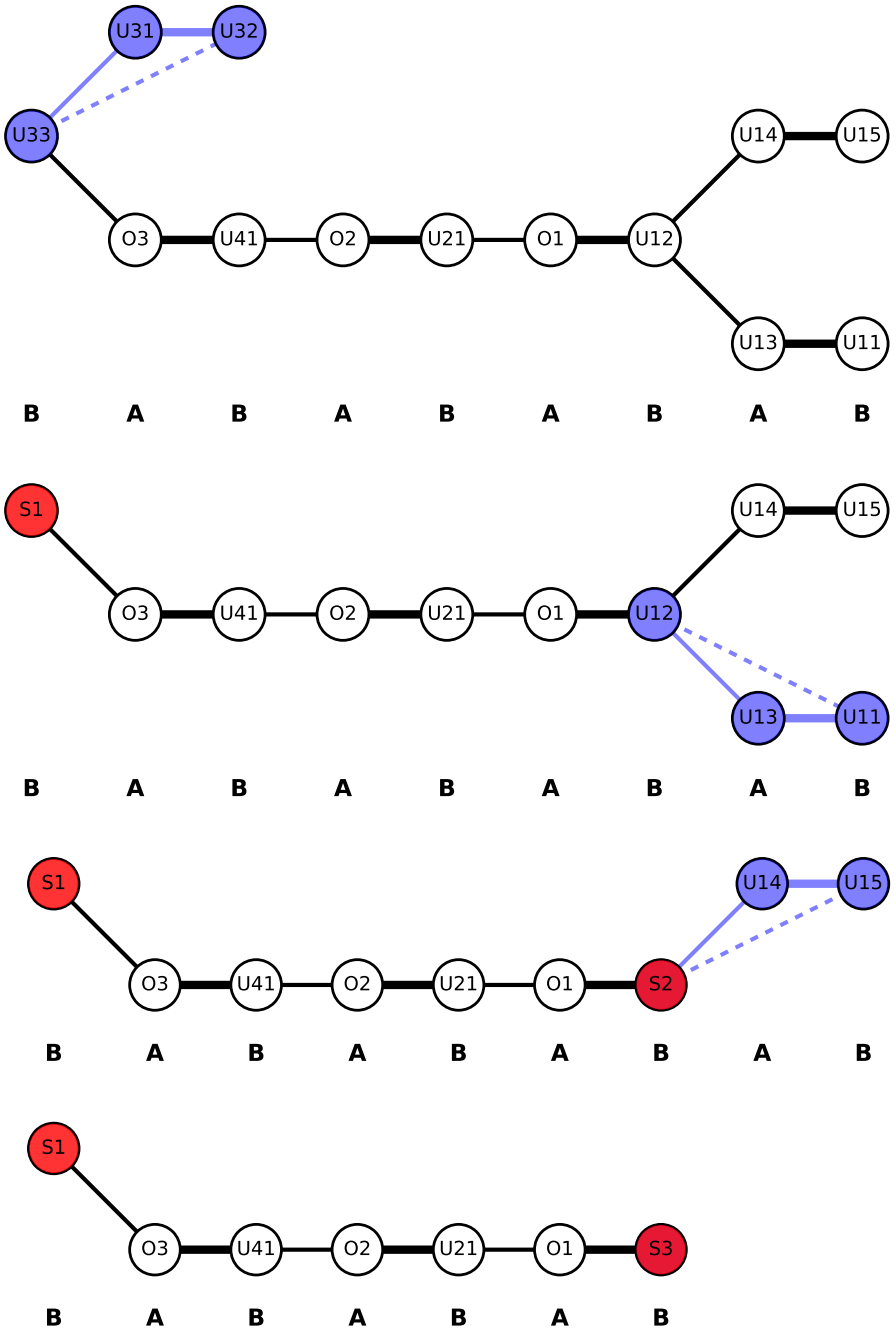


Figure 2.3: Illustration of the four shrinking steps needed to compute the *Gallai-Edmonds decomposition* of the graph shown in Fig. 2.2. The alternating tree starts at the node on the left side. The odd circuit that is shrunk is highlighted in light blue. The resulting pseudo node is highlighted in red.

3 Priority Mechanism

In this section, the before presented concepts from graph theory will be applied to pairwise kidney exchanges. In the case of kidney exchange, the nodes in the graph are the (incompatible) patient-donor pairs and the edges denote a crosswise compatibility between two pairs. Particular interest will be laid on the question of which maximum matching should be used among the potential large number of maximum matchings. The first part of this section is a short summary of the ideas of Roth et al. [4]. Then, an alternative implementation of their prescription to match the patient-donor pairs will be presented which allows an extension with additional patient-donor preferences.

Note that each patient can also have more than just one donor. This results in additional compatibilities with other patients and is modeled with additional edges between the nodes. After a matching has been computed, one can infer from the edges in the matching which donor of the patient is selected.

A maximum matching can be computed efficiently with the blossom algorithm that has been presented in the previous section. However, the critical question is which maximum matching should be selected. Roth et al. [4] give a prescription how to handle this issue which they called *priority mechanism* and which will be shortly summarized here.

3.1 Priority Mechanism using the Gallai-Edmonds Decomposition

A common procedure in medicine is to give each patient a priority, e.g., according to health status, age and time on the waiting list. This is already done for the distribution of cadaver kidneys. However, starting with the patient with the highest priority and then adding patients to the matching with decreasing priority as long as all patients can be matched will not result in a maximum matching. The solution to this challenge lies in the structure of a maximum matching, the *Gallai-Edmonds decomposition*.

All patients in the set of *perfectly matched patients* N^P are matched among themselves in all maximum matchings and are completely decoupled from the rest of the graph. All patients in the set of *overdemanded patients* N^O are matched with a patient in the set of *underdemanded patients* N^U in each maximum matching as well. The assignment of a patient in N^O to a patient in N^U can be different in each maximum matching. The only set of the *Gallai-Edmonds decomposition* in which not all patients get matched is the set of *underdemanded patients* N^U. So the

problem is reduced to the problem of finding a maximum matching that maximizes the priorities of underdemanded patients as all other patients get matched anyway.

The set N^U consists of disjoint sets D_i of odd cardinality. In each odd component D_i all but one patient get matched within D_i. The remaining patient can only be matched with a patient in N^O, i.e., the distribution of overdemanded patients to underdemanded patients determine which odd components D_i are fully matched.

We order the odd components decreasingly according to the priority of their lowest priority patient. Then, let $\mathcal{D} = \{D_1, D_2, D_3, ..., D_N\}$ be the ordered set of odd components with $N^U = \bigcup_i D_i$. The procedure is to first match the patients of D_1 and then to continue with the next odd component. The patients of the next odd component get fully matched only if the previous odd components can still be matched. If one odd component can not be matched it will be skipped. Please note that the mapping between a patient of D_i to a patient of N^O can change in each step.

A formal description of this algorithm is given in [4] as follows:
For each $\mathcal{J} \subseteq \mathcal{D}$ and $\mathcal{I} \subseteq N^O$ the neighbors of the set of odd components $\mathcal{J}$ among overdemanded patients in $\mathcal{I}$ is defined as

$$\mathcal{C}(\mathcal{J}, \mathcal{I}) = \left\{ i \in \mathcal{I} : \exists J \in \mathcal{J} \text{ with } \tilde{r}_{i,J} = 1 \right\} , \tag{3.1}$$

where $\tilde{r}_{i,J}$ denotes the existence of a link between patient i and set J and is defined as

$$\tilde{r}_{i,J} = \begin{cases} 1 & \text{if } \exists j \in J \text{ s.t. } r_{i,j} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

and $r_{i,j}$ is 1 if the patient-donor pair i is compatible with the patient-donor pair j and 0 otherwise.

The odd components that will be fully matched under the priority mechanism can be computed using the following procedure.

Step 1: If $|\mathcal{C}(\{D_1\}, N^O)| \geq |\{D_1\}| = 1$, then let $\mathcal{J}_1 = \{D_1\}$. This means that the odd component D_1 has a connection to N^O and in this case all of its members will be matched. If $|\mathcal{C}(\{D_1\}, N^O)| < |\{D_1\}| = 1$, then let $\mathcal{J}_1 = \emptyset$. In this case all members of D_1 but its lowest priority patient will be matched.

Step k: If $|\mathcal{C}(\mathcal{J} \cup \{D_k\}, N^O)| \geq |\mathcal{J} \cup \{D_k\}|$ for every $\mathcal{J} \subseteq \mathcal{J}_{k-1}$, then let $\mathcal{J}_k = \mathcal{J}_{k-1} \cup \{D_k\}$. In this case all members of D_k will be matched. If $|\mathcal{C}(\mathcal{J} \cup \{D_k\}, N^O)| < |\mathcal{J} \cup \{D_k\}|$ for some $\mathcal{J} \subseteq \mathcal{J}_{k-1}$, then let $\mathcal{J}_k = \mathcal{J}_{k-1}$. In this case all members of D_k but its lowest priority patient will be matched.

The result of the priority mechanism, i.e., the transformation of a maximum matching to the maximum matching under the priority mechanism is shown in Fig. 3.1.

It was not explicitly discussed in [4] how to implement the algorithm to guarantee a polynomial runtime. If the procedure is modeled in a straightforward way it is computationally very expensive. In each step all subsets $\mathcal{J} \subseteq \mathcal{J}_{k-1}$ need to be considered. Thus, the runtime scales exponentially. This is one motivation for a different implementation of the priority mechanism which will be described in the next section.

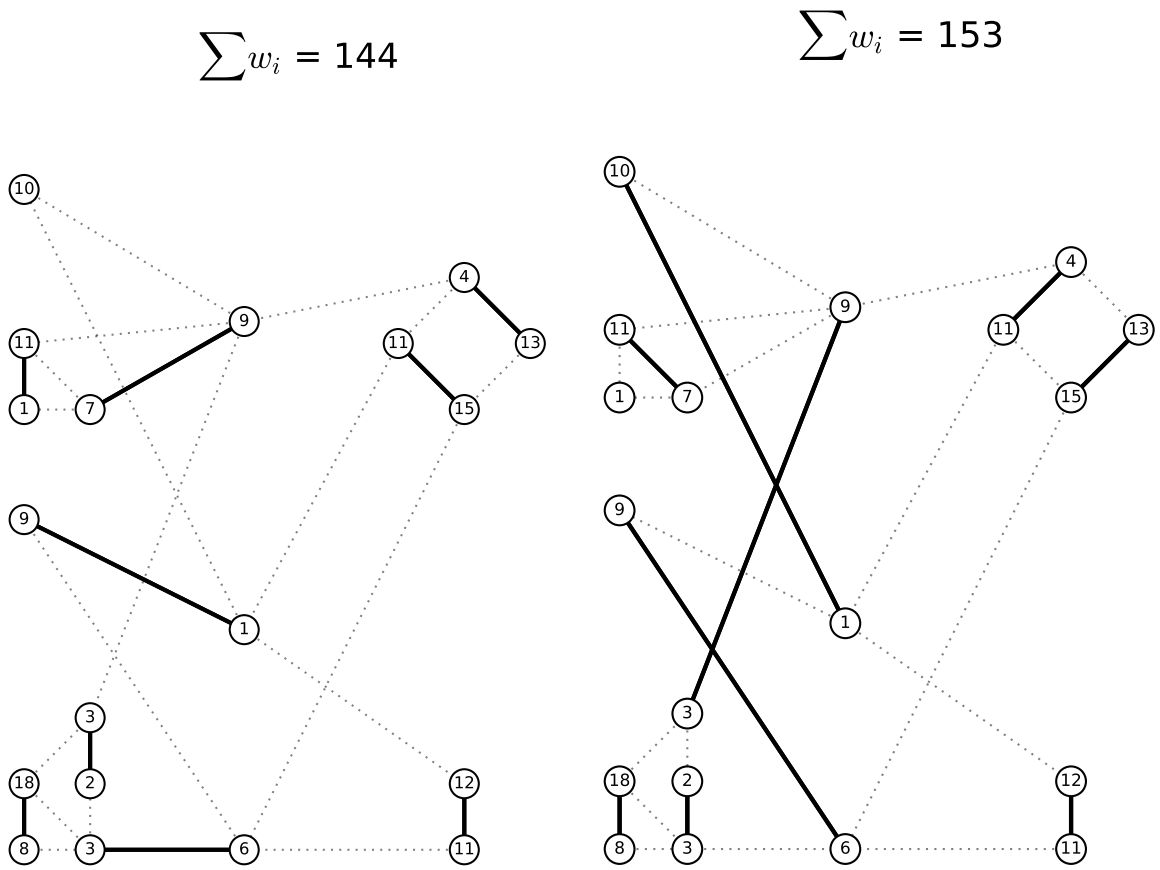


Figure 3.1: (left) Maximum cardinality matching. (right) Maximum cardinality matching under the priority mechanism. The numbers in the nodes denote the priority of the patients. The dashed lines are the edges and the bold solid lines the edges that belong to the matching.

3.2 Formulation as Maximum Weight Matching

The priority mechanism can be formulated as a maximum weight matching. This is a matching in which not the cardinality of the matching is maximized but the sum of the weights of the edges participating in the matching. Maximum cardinality of the matching can be achieved as well by adding a large constant c to each edge weight. In other words, among the maximum cardinality matchings the matching with the largest sum of edge weights is selected.

An extension of the blossom algorithm which uses a *primal-dual* method for finding a matching of maximum weight can be used. This algorithm is also developed by Edmonds [18] and runs in $\mathcal{O}(n^3)$ time (n is the number of nodes) [19] which is a significant improvement compared to the exponential scaling of the run-time found in the previous section.

In the previous sections no edge weights are used and in the priority mechanism the nodes but not the edges have a priority. We give the following prescription to transform node priorities

to edge weights. The edge weight is defined as

$$w_e = w_{a,b} = p_a + p_b + c\,, \tag{3.3}$$

where p_a and p_b are the priorities of the nodes a and b and c is a large constant with $c > \sum_{v \in V} p_v$. With this definition of edge weights the following statement holds.

Lemma 2. *The maximum weight matching is equal to the maximum cardinality matching under the priority mechanism.*

Proof. Let M_1 be a matching of size $|M_1| = k$ and let M_2 be a matching of size $|M_2| = k+1$, then

$$\begin{aligned}
\sum_{e \in M_1} w_e &= k \cdot c + \sum_{v \in M_1} p_v \\
&< k \cdot c + c && \text{because of } c > \sum_{v \in M_1 \subseteq V} p_v \\
&< (k+1) \cdot c + \sum_{v \in M_2} p_v = \sum_{e \in M_2} w_e\,,
\end{aligned}$$

where $\sum_{v \in M}$ denotes a sum over all vertices of the matching M. Hence, due to the addition of the large constant c to each edge weight, a matching with larger cardinality has also a larger sum of edge weights than any matching with smaller cardinality. Thus, the maximum weight matching with edge weights as defined in Eq. (3.3) is a maximum cardinality matching.

To prove the rest of the lemma we can again use the knowledge of the structure of the graph from the *Gallai-Edmonds decomposition* and consider one component after the other.

N^P *(perfectly matched patients):* These nodes will always be matched. Thus, they have always the same contribution to the sum of weights $\sum_i w_i$. Therefore, they have not to be taken into account in further considerations.

N^O *(overdemanded patients):* Here, the same argument as for $\mathrm{N^P}$ holds. In addition, it is irrelevant with which node from N^U a node from N^O is matched.

N^U *(underdemanded patients):* For each odd component D_k at least $|D_k| - 1$ nodes will be matched. If one node $\in D_k$ remains unmatched, it is the node with the lowest priority, because this maximizes the sum of weights.

Only the lowest priority node of each odd component needs to be considered (all other nodes get matched anyway). We denote by C the collection of these lowest priority nodes. With our definition of the edge weights, all nodes contribute linearly with their priority to the sum of weights. Therefore, maximizing the sum of priorities of the patients $v \in C$ used in the matching is equivalent to maximizing the overall sum of weights.

$$\sum_{e \in M} w_e = \underbrace{|M| \cdot c + \sum_{v \in N^P} p_v + \sum_{v \in N^O} p_v + \sum_{v \in N^U \setminus C} p_v}_{\text{constant}} + \sum_{v \in C \cap M} p_v\,, \tag{3.4}$$

where p_v is the priority of node (patient) v and $C \cap M$ denotes the nodes $v \in C$ that get matched in the matching M.

The priority mechanism sorts all nodes $v \in C$ according to their priority and loops through the nodes starting with the highest priority node. A node is added to the matching if all previously matched nodes $\in C$ can still be matched. This also means that a node that is not matched in the priority mechanism could only be matched at the expense of a node with higher priority, thus, reducing the overall sum of weights. Hence, the priority mechanism automatically generates a matching of maximum weight according to our definition of edge weights.

□

3.3 Inclusion of Directly Compatible Pairs

If a patient finds a compatible donor among his family or friends, he normally does not participate in a kidney exchange program as he already has a compatible donor. However, his participation would help all other patients in the kidney exchange program.

Consider the following example. Patient A has blood type "A" and his donor has blood type "0". A person with blood type "0" is the universal donor as his blood is compatible to all other blood types, but this is not true for the other direction. A patient with blood type "0" can only receive organs from donors that also have blood type "0". So let's assume that there is another patient B with blood type "0" that has a donor with blood type "A". In this case his donor is incompatible but if patient A and B would exchange their donors, both could receive a new kidney. Especially patients with blood type "0" would benefit from the participation of directly compatible pairs as otherwise there is always a lack of blood type "0" donors.

The benefit of this idea will be quantized in the case study of the next chapter. Here, the extension of the matching algorithm to also include directly compatible pairs will be presented.

The matching algorithm used so far does not allow a matching of a node with itself which would correspond to the assignment of a patient with its own donor. This case is needed because it can of course happen that the directly compatible pair does not find a match with another pair in the kidney exchange pool. Then, the patient should be matched with its own donor.

We solve this problem by adding a dummy node for every directly compatible pair to the graph and connect each of these pairs with an edge of weight zero to its dummy node. Then, this dummy edge would only participate in a maximum cardinality maximum weight matching if it increased the cardinality of the matching, i.e., only if the directly compatible pair can not be matched with another incompatible pair, it will be matched with its own donor. See Fig. 3.2 for an illustration.

3.4 Inclusion of Additional Patient-Donor Preferences

The formulation of the priority mechanism as maximum weight matching, allows the consideration of additional patient preferences in the matching algorithm. Consider the following example: A patient has two willing but incompatible donors. One donor is his spouse and the other a second cousin. The patient would prefer to have his spouse as a donor rather than his relatively young second cousin. This wish can now be modeled by choosing the edge weights accordingly.

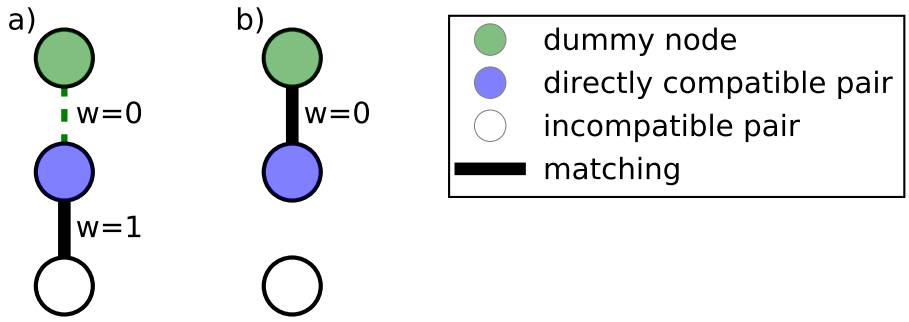


Figure 3.2: Illustration of the usage of dummy nodes. a) The directly compatible pair is matched with an incompatible pair. b) The directly compatible pair can not be matched with an incompatible pair, so it is matched with its dummy node.

His wish expressed more formally is: "I want my second cousin as a donor only if there exists no maximum cardinality matching with my spouse as a donor."

This can be achieved by adding patient-donor preferences to the edge weights and setting the scales of the patient priorities and the patient preferences at different orders of magnitude. One possible extension of the edge weight definition from Eq. (3.3) would be

$$w_{ab} = 10 \cdot w_{\text{preference}} + p_a + p_b + c \tag{3.5}$$

where p_a and p_b are the priorities of patient a and b and $w_{\text{preference}}$ is the additional patient-donor preference. c is again a large enough constant to guarantee maximum cardinality of the matching. $w_{\text{preference}}$, p_a and p_b is restricted to the interval $[1,2]$.

In our example, we set $w_{\text{preference}} = 1$ for the edge with the cousin as donor and we set $w_{\text{preference}} = 2$ for the edge with the spouse as donor, thus giving his spouse a higher weight. This example is illustrated in Fig. 3.3.

In practice it would be beneficial to only allow binary patient preferences, i.e., $w_{\text{preference}} \in \{1,2\}$. This is because a patient reduces his probability to get matched if he reduces his preference for a donor from 2 to 1. Consider two patients A and B that can only be matched with the same patient C. Patient A has priority 2 and patient B has priority 1. If both would set the preference of their donor to 2, patient A would be matched but if patient A would set the preference of his donor to 1, patient B would be matched. See Fig. 3.4 for an illustration.

The restriction to binary preferences is reasonable because a patient should not win against another patient because he increased his preferences by a very small number, e.g., from 1 to 1.001. Furthermore, the default patient donor preference should be set to 2 and only patients that come with multiple donors should decrease the preference of some of their donors to set a rank order among their donors.

This is a clear improvement with respect to the previous situation, where the patient's only option to test if his preferred donor would get him matched was to conceal some of his donors.

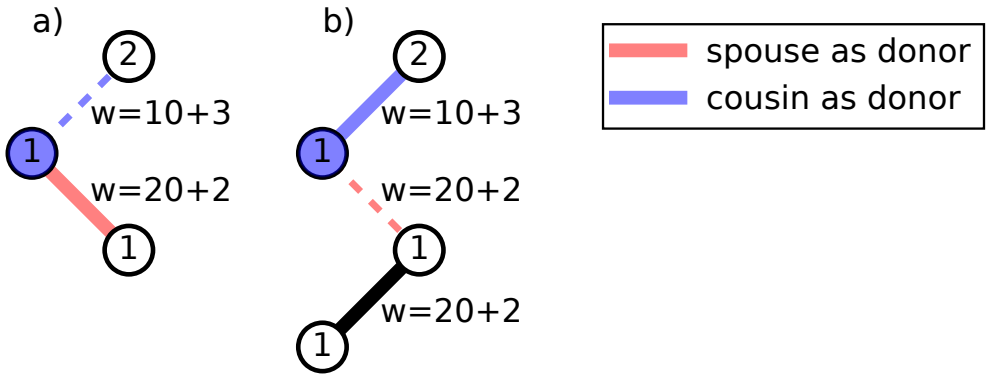


Figure 3.3: Illustration of usage of additional patient-donor preferences. The blue circle is the patient with two different donors that have different weights. The numbers in the circles are the priorities of the patients. a) The donor with the larger priority (spouse) can be used to achieve a maximum cardinality matching. b) Only with the donor with lower priority (cousin) a maximum cardinality matching can be achieved.

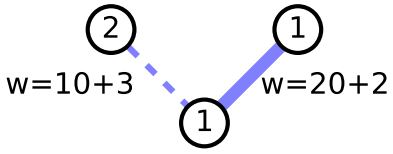


Figure 3.4: The upper left patient has the highest priority but as he reduces the preference of his donor from 2 to 1 he will not be matched.

Hence, our extension of the algorithm with additional patient-donor preferences even improves the incentive compatibility of the algorithm. A patient can now name all of his donors and can still incorporate his wish of a preferred donor.

Another usage of edge weights is to specify different levels of compatibility. Currently, a donor kidney can be either compatible or not compatible to a patient. With edge weights, this binary compatibility can be replaced by several levels of compatibility, e.g., from “not compatible” over “mostly compatible” to “perfectly compatible”. However, such an extension is problematic for the incentive compatibility of the matching algorithm. A patient has a higher probability to get matched if all his potential donors are rated as “perfectly matched”. Hence, the patient or the attending doctors could have an incentive to not assess the patient’s health status correctly.

4 Case Study

In this chapter, the matching algorithm will be applied to simulated patient-donor data sets. The goal of this case study is to answer questions like “How much does the number of matched patients increase with the overall number of patients participating in the exchange pool?”, “What is the optimal size of an exchange pool?” and “What is the influence of the blood type in a maximum matching?”.

4.1 Simulation of Patient-Donor Data Set

The patient-donor data set is simulated to best reproduce the true situation. Each patient and donor has two properties, its blood type and its tissue type. The distribution of blood types is known so we can randomly chose a blood type for each patient and donor according to this distribution. As the distribution of blood types differ from country to country, the distribution of blood types as in Germany will be used which is shown in Fig. 4.1.

To characterize the blood type the AB0 system including the Rhesus (Rh) factor is used, i.e., the four main groups “A”, “B”, “AB” and “0” can each have a positive or a negative Rh factor. A positive Rh factor implies that the red blood cells have the additional “D” antigen on their surface. Therefore in general, a patient with negative Rh factor can donate his blood to a patient with positive Rh factor but not the other way around.

The compatibilities between the blood types are listed in Tab. 4.1. A person with blood type “0” is a universal donor as his blood is compatible to all other blood types, but this is not true for the other direction. A patient with blood type “0” can only receive organs from donors that also have blood type “0”. In contrast, a person with blood type “AB” is the universal recipient. He can receive organs from all blood types but give his organs only to patients with the same blood type.

So for each patient-donor pair we randomly choose a blood type for the patient and for the donor according to the distribution in Germany. In the next step, we randomly choose a tissue type. As discussed in the introduction, the importance of a good match in the tissue type is assessed differently in different countries. Hence, we consider different scenarios. We start with ignoring the patients’ and donors’ tissue type, i.e., the compatibility of a patient with a donor is only determined by their blood types. Then, we increase the importance of a good match of the tissue types of the patient and its donor. The four scenarios we consider are:

- All patients and donors have the same tissue type, i.e., the compatibility between a patient

and its donor is only due to compatible blood types according to Tab. 4.1.

- 4 different tissue types, i.e., the probability of a compatible tissue type is $p = 25\%$.
- 10 different tissue types, i.e., the probability of a compatible tissue type is $p = 10\%$.
- 100 different tissue types, i.e., the probability of a compatible tissue type is $p = 1\%$.

If a patient-donor pair is directly compatible (according to its blood and tissue type), we have two options: The first option is to discard this pair and to simulate a new one. This would correspond to the actual situation in hospitals today. If a patient already has a compatible donor, he will not participate in a kidney exchange program. However, as we also want to study the benefit of including already compatible pairs in the exchange program, the second option is to not discard this pair.

To make the two sets with and without directly compatible pairs comparable, we simulate the sets the way that the number of incompatible pairs remains the same. Suggest the number of incompatible pairs should be 30, then it may occur that the set including directly compatible pairs consists of 35 pairs because it contains 5 directly compatible pairs.

Thus for better comparison, we also determine the number of patients that can be matched by counting only the patients that come with an incompatible donor and get matched in the exchange pool. The patients with directly compatible donor will always be matched and would therefore distort the matching size upwards if included in the determination of the matching size.

4.2 Structure of Graphs and Analysis of Gallai-Edmonds Decomposition of Patient Data

Fig. 4.2 shows the resulting graph of a data set with 30 patient-donor pairs and a probability of tissue type compatibility of 25%. The dashed and solid lines (the edges of the graph) indicate the compatibility of two pairs. The computed maximum matching is marked by solid lines. Whereas the upper graph excludes directly compatible pairs, the lower graph shows the inclusion of directly compatible pairs. In this example, the inclusion of directly compatible pairs helps one patient with incompatible donor to find a match, in other words, one additional live can be saved.

Recipient	Donor							
	0-	0+	A-	A+	B-	B+	AB-	AB+
0-	√	x	x	x	x	x	x	x
0+	√	√	x	x	x	x	x	x
A-	√	x	√	x	x	x	x	x
A+	√	√	√	√	x	x	x	x
B-	√	x	x	x	√	x	x	x
B+	√	√	x	x	√	√	x	x
AB-	√	x	√	x	√	x	√	x
AB+	√	√	√	√	√	√	√	√

Table 4.1: Blood type compatibility matrix.

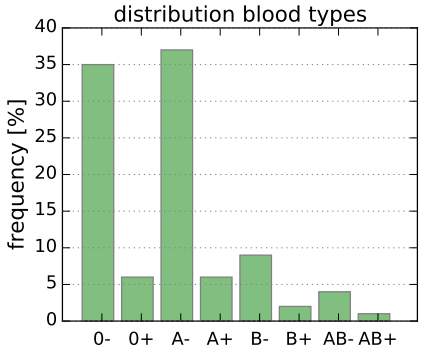


Figure 4.1: Blood type distribution in Germany.

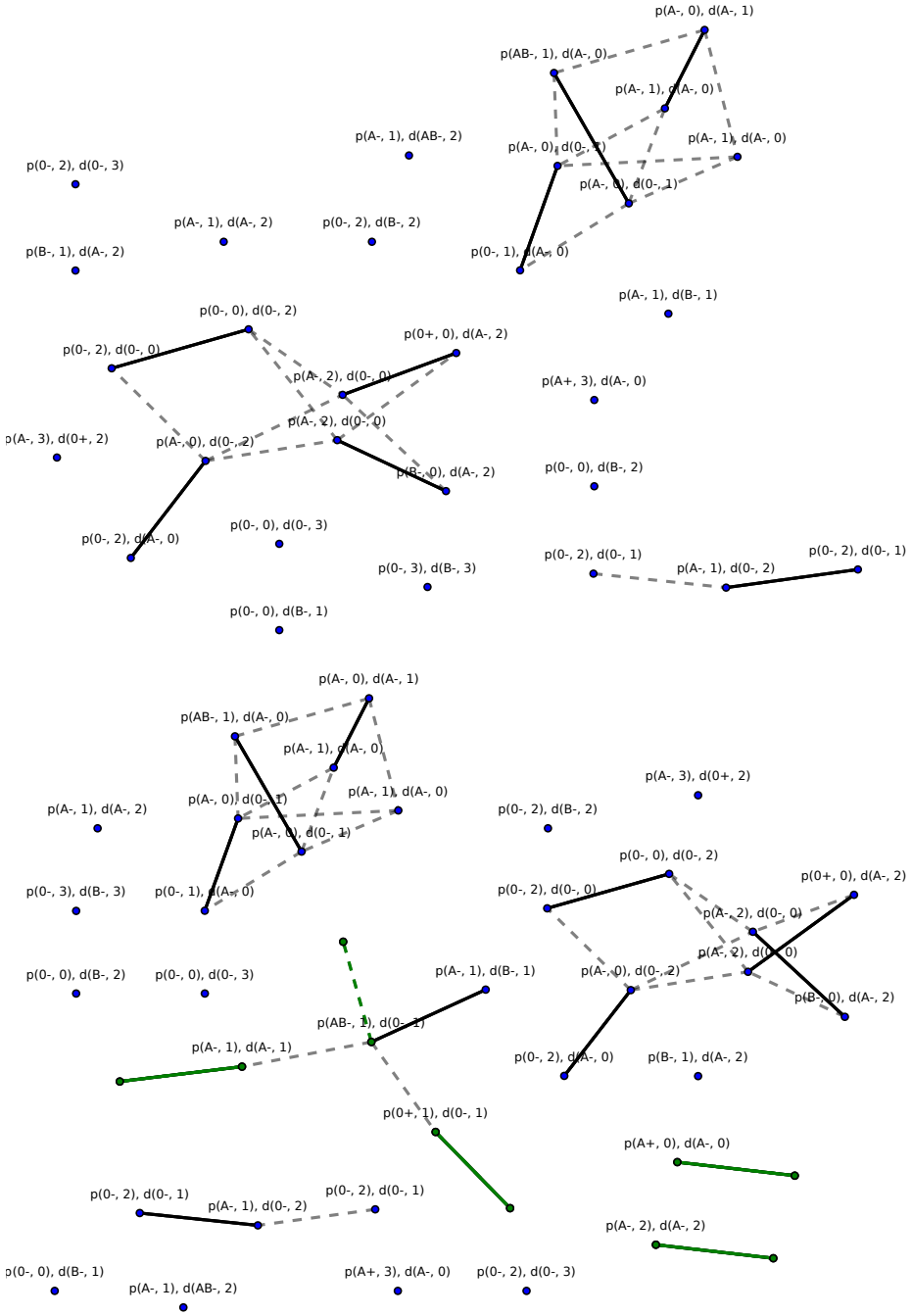


Figure 4.2: (top) Example of a simulated data set with 30 patient-donor pairs. The probability for a compatible tissue type was set to 25% (four different tissue types). The dashed lines represent the edges of the graph and the solid lines show the edges of the maximum matching. Next to each node the blood and tissue types of the patient and the donor are visible. E.g., "p(0-, 1)" means that the patient has blood type "0-" and tissue type "1" and "d(A-, 0)" means that the donor has blood type "A-" and tissue type "0". In this example 53% of the patients can be matched. (bottom) Shows the same example as above but includes directly compatible pairs, i.e., it includes patients that already have a compatible donor. These nodes and their dummy nodes are marked in green. In this case 63% of all patients and 57% of the patients with incompatible donor can be matched.

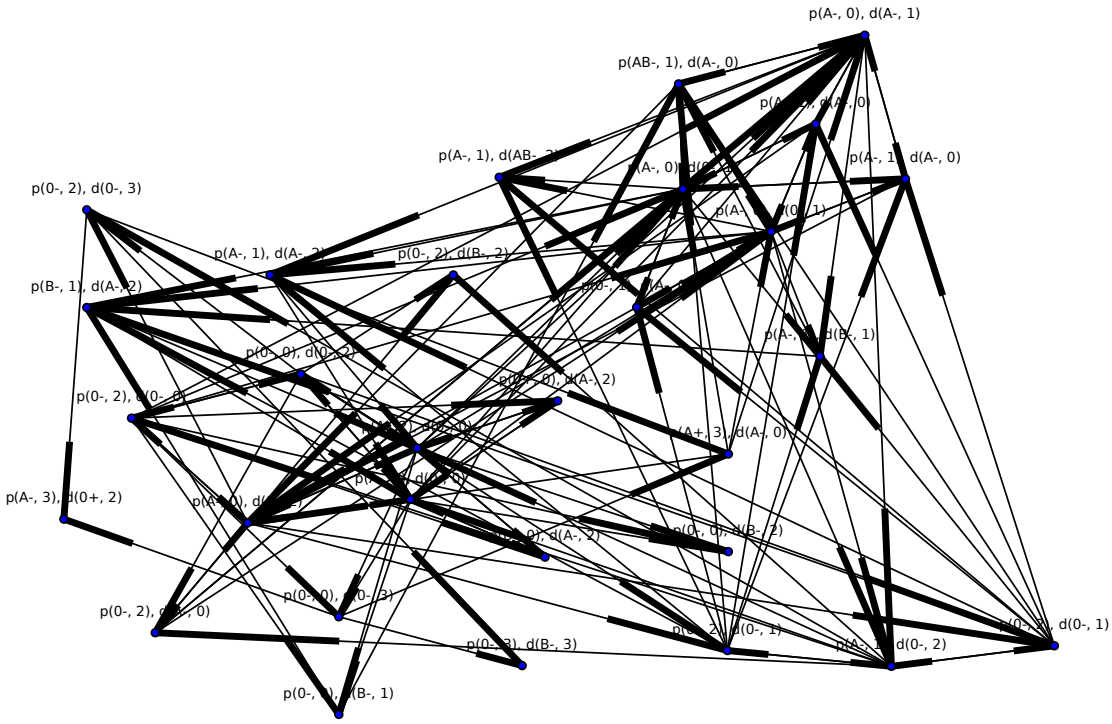


Figure 4.3: Same graph as in in Fig. 4.2 top but plotted as directed graph. An arrow from node *a* to node *b* means that the donor of patient *a* is compatible to patient *b*.

If two pairs are not compatible to each other, they are not completely independent. Often a patient is compatible to the others patient's donor but not the other way around. This is visible in the directed graph in Fig. 4.3.

The *Gallai-Edmonds decomposition* of the simulated patient-donor data sets yield interesting results. Other than expected from the example of Fig 2.2 the set of underdemanded patients N^U consists almost always of single nodes, i.e., each odd component of N^U consists of only one node (see Fig. 4.4 for an example). This is also true for larger data sets. Only for large data sets ($n \geq 500$), two patients per donor and a large probability of tissue type compatibility we found an odd component with several entries.

The histograms in Fig. 4.5 show the percentage of patients in the three components of the *Gallai-Edmonds decomposition* for three different probabilities of tissue type compatibility. We discovered that this number hardly depends on the size of the data set. Consequently, the numbers of Fig. 4.5 are the average over data sets with different input size.

The number of patients in N^U increases with decreasing p_{tissue} mainly because less patients can be matched and all patients that can not be matched must belong to the set N^U by definition.

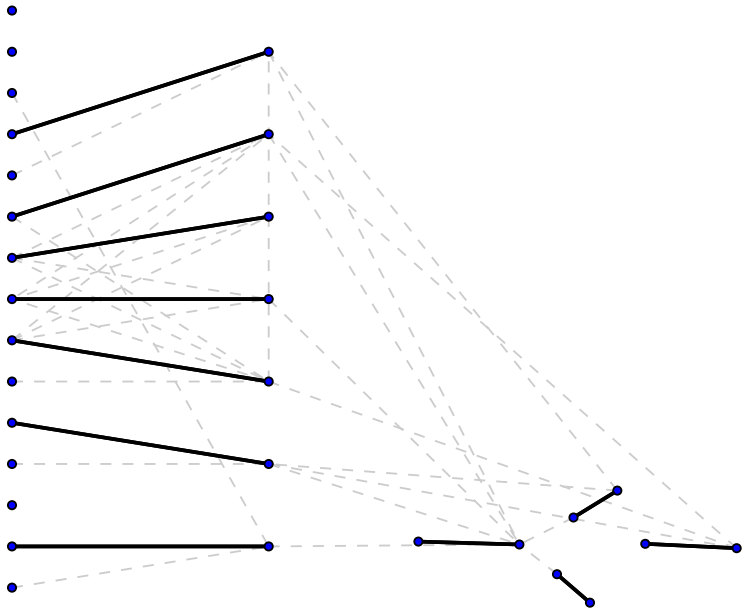


Figure 4.4: Gallai-Edmonds decomposition of a patient-donor data set with 30 pairs, two donors per patient, without directly compatible pairs and a probability of tissue type compatibility of 25%. The left part is the set N^U, the part in the middle is the set N^O and the right part is the set N^P.

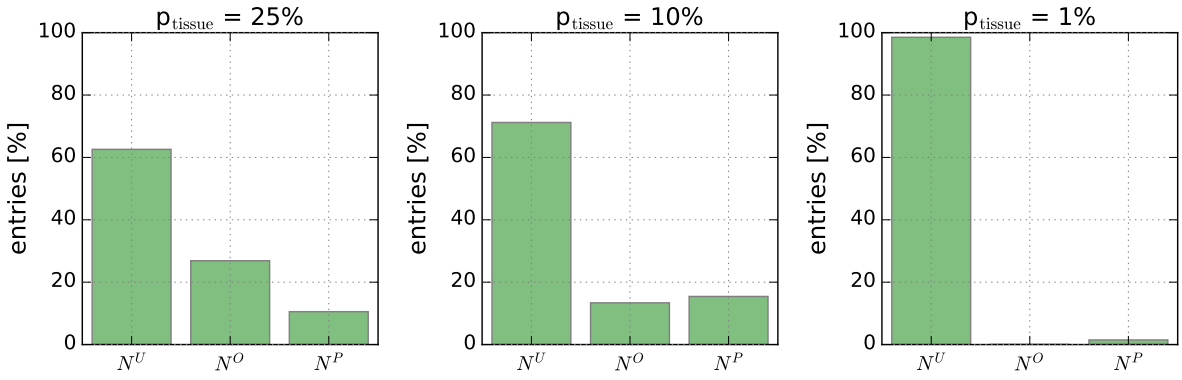


Figure 4.5: Number of patients in the different sets of the Gallai-Edmonds decomposition for three different probabilities of tissue type compatibility. The histograms show data sets with only one donor per patient and without directly compatible pairs. The numbers state the average of ten random realizations and three different sizes of the exchange pool (n = 100, 500, 1000).

4.3 Dependence of Matching Size on Input Size

Another important result of this case study is the dependence of the number of patients that can be matched on the size of the exchange pool. This is of practical relevance for the installation of a kidney exchange program as it answers the question of how large the program must be to make sense, e.g., would it be sufficient to have an exchange program in greater Aachen or should it be complete North Rhine-Westphalia? In Germany ∼8,000 patients are currently on the waiting list for a cadaver kidney. If all of them would participate in an exchange pool, 54 patients would live in greater Aachen and already 1736 of them would live in North Rhine-Westphalia[1].

The result is shown in Fig. 4.6 where the number of matched patients versus the total number of patient-donor pairs is presented for different probabilities of tissue type compatibility, number of donors per patient and with/without the inclusion of directly compatible pairs (denoted as "selfloop = 1/0"). As this number also depends on the concrete random realization of the data set, we average over several realization of the data set for the same parameters and show the spread as errorbar.

Going from one donor per patient to two donors per patient roughly doubles the number of matched patients unless the number of matched patients is already saturated, i.e., its number is already close to 100%. In the case of $p_{\text{tissue}} = 1\%$ the improvement is even larger than a factor of two. This is because the number of edges of the graph increases by more than a factor of two which increases the possibilities for a maximum matching.

The improvement of the inclusion of directly compatible pairs (selfloop = 1) is most prominent for large p_{tissue}. For $p_{\text{tissue}} = 100\%$ the number of matched patients increases from ∼30% to almost 100%. Please keep in mind that for the number of matched patients we only count the number of patients that come with an incompatible donor. So for a scenario where a good match of the tissue type is not important (as in the United States), the benefit of the inclusion of directly compatible pairs is tremendous. However, its relevance decreases with decreasing probability of tissue type compatibility. For $p_{\text{tissue}} = 1\%$ the effect of the inclusion of directly compatible pairs is negligible.

The black squares in Fig. 4.6 top left, that denote a data set without the inclusion of directly compatible pairs and irrelevant tissue type, saturate at ∼35%.. This is among other things because a patient with blood type "0" will never find a match. If he would have a compatible donor he would not participate in this data set. Furthermore, he can only receive a kidney from a donor with blood type "0" but a donor with blood type zero is the universal donor and is therefore always compatible to its own patient and hence also not in this data set.

[1] We used the following numbers of inhabitants for this calculation: Greater Aachen: 547,661 inhabitants, North Rhine-Westphalia: 17,638,098 inhabitants, and Germany: 81,292,428 inhabitants.

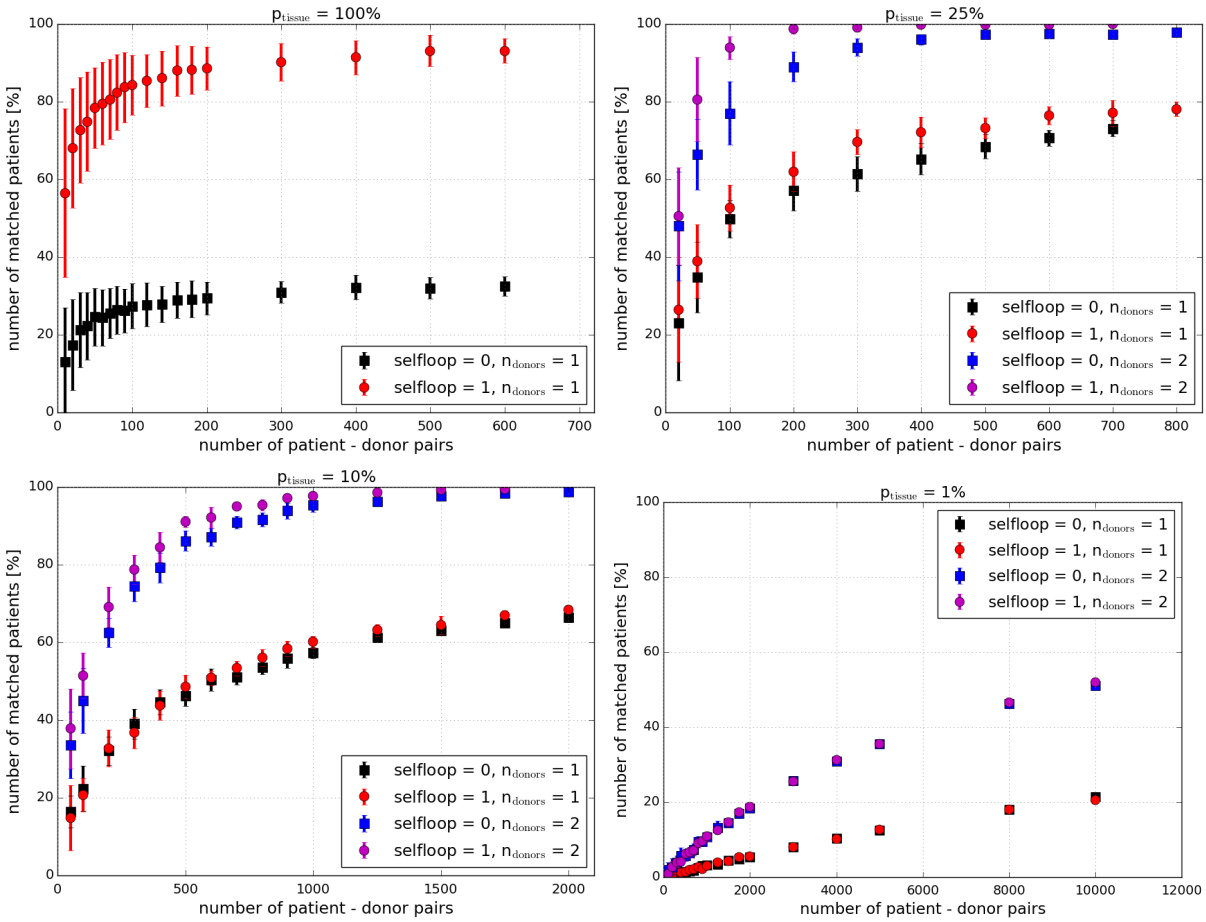


Figure 4.6: Percentage of patients that can be matched vs. the size of the exchange pool for four different probabilities of tissue type compatibility. The markers show the mean and the error bars the standard deviations of at least ten realizations of the simulated data set for the specific parameters. For data points with large scatter significantly more realizations have been simulated to reduce the uncertainty on the mean. The squares denote data sets where only incompatible patient-donor pairs are considered whereas the circles denote data sets where also directly compatible pairs participate in the kidney exchange pool. The numbers on the x- and y-axis refer only to the incompatible pairs.

4.4 Distribution of Blood Types in the Gallai-Edmonds Decomposition

Further insights can be gained if the distribution of blood types of patients and donors in the different sets of the *Gallai-Edmonds decomposition* is analyzed. This is of interest because the blood type compatibility is not symmetric, e.g., blood type "0" can be donated all other blood types but a patient with blood type "0" can only receive organs from donors with the same blood type. See Tab. 4.1 for a complete overview.

Furthermore, the occurrence of blood types is not uniform as shown in Fig. 4.1. To make the distribution of blood types in the different components of the *Gallai-Edmonds decomposition* comparable, the number of patients or donors with blood type "X" is divided by the occurrence of blood type "X" in the population according to Fig. 4.1. Consider the following example: We have 20 patients with blood type "0-" but only 10 patients with blood type "AB-" in the set of overdemanded patients N^O. Naively one would think that a patient with blood type "0-" is more likely to end up in N^O than a patient with blood type "AB-". However, we must take into account the probability that a patient has a certain blood type. In this example, the probability of having blood type "0-" is $\sim$35% whereas the probability of having blood type "AB-" is only $\sim$4%. If we weight the occurrence of blood type "X" in N^O with its occurrence in the population, we get $20/35\% = 57$ for blood type "0-" and $10/4\% = 250$ for blood type "AB-". So actually a patient with blood type "AB" is more likely to end up in the set N^O.

This normalization is done in the histograms of Fig. 4.7 where the distributions for a set of 100 patient-donor pairs, one donor per patient, p_{tissue} = 25% and without the inclusion of directly compatible pairs are shown. The histograms for different parameters of the data set are qualitatively similar and therefore only shown in the appendix (Fig. A.1 - A.3).

The most prominent feature is that most donors in the set of overdemanded patients have blood type "0-" which is not surprising as a person with blood type "0-" is the universal donor. In contrast, only a small number of patients with blood type "0-" are in the set of overdemanded patients and for the set of underdemanded patients it's the other way around. A large number of patients with blood type "0-" are in N^U, which means that a lot of them can not be matched, and only a small number of donors with blood type "0-" are in N^U.

Patients with blood type "AB-" are the universal recipients. Therefore, their occurrence in N^O is frequent. In contrast, donors with blood type "AB-" can only donate their kidney to patients with the same blood type. Hence, their occurrence in N^O is very small.

Concluding, a patient with blood type "AB" or a patient that comes with a donor with blood type "0" has good chances to get matched whereas a patient with bood type "0" or a patient that brings a donor with blood type "AB" has the lowest chance to get matched.

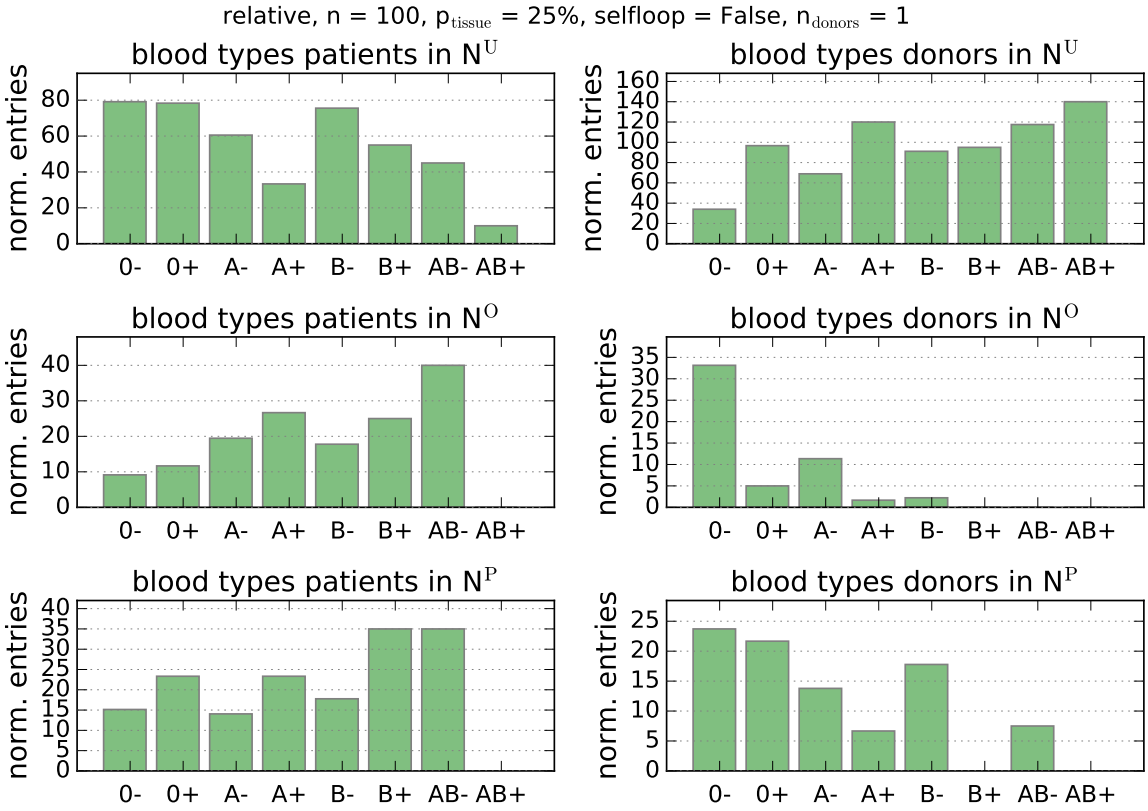


Figure 4.7: Distribution of blood types of patients (left) and donors (right) grouped in the three sets of the Gallai-Edmonds decomposition: underdemanded patients (top), overdemanded patients (center) and perfectly matched patients (bottom). The histograms show the average of 10 realizations of a data set with 100 incompatible patient-donor pairs, a 25% probability of tissue type compatibility and only one donor per patient.

5 Summary

The usage of living donors has a promising potential to satisfy the large demand of kidney transplants. Advances in medicine nowadays allow a healthy person to donate one of his kidneys without the risk of death or any limitation in his future life. However, finding a compatible donor among family and friends of a patient remains difficult. One solution for this problem is a kidney exchange pool in which many incompatible patient-donor pairs are registered. Then, an incompatible patient-donor pair can be matched with another incompatible pair such that they are crosswise compatible, i.e., the donor of the first patient is compatible to the second patient and vice versa.

The maximum number of compatible pairs in such a kidney exchange pool can be computed efficiently with a matching algorithm where the problem is modeled as a graph in which the patient-donor pairs are represented by the nodes and the edges denote a crosswise compatibility between two pairs. Normally, this combination is not unique but several maximum matchings exist. We discussed a prescription, the so-called priority mechanism, to select the maximum matching that maximizes the priority of the patients. This algorithm is already discussed in literature and makes use of the Gallai-Edmonds decomposition. We give an alternative implementation that runs in $\mathcal{O}(n^3)$ time and uses a maximum weight matching algorithm where the priorities of the patients are transformed into edge weights.

This new formulation of the priority mechanism allows for an extension of the algorithm that takes additional preferences of the patients into account. We discussed the case where a patient has two willing but incompatible donors. With our extension of the algorithm this patient can now declare a preferred donor.

Furthermore, we studied the benefit of the implementation of an exchange pool on simulated patient-donor data sets. The compatibility between a patient and a donor is determined by the blood and tissue type, whereby the importance of a good match of tissue type is evaluated differently in different countries. Hence, we performed this case study with four different assumptions of the importance of a good tissue type match.

We determined how many patients will find a match in an exchange pool of a certain size on average. For a probability of 25% that patient and donor have a compatible tissue type, and considering only patients that come with an incompatible donor, we found that $\sim$60% of the patients will be matched if the exchange pool has a size of 300 patient-donor pairs. If we increase the number of donors that each patient brings to the exchange pool from one to two, already more than 90% of the patients will find a match.

We also studied the impact of patients participating in an exchange pool who have already

found a compatible donor. These patients must always be matched but not necessarily with their own donor. In the scenario where a good match of the tissue type is considered unimportant, the improvement is largest. The number of patients with incompatible donor that can be matched increases from 30% to 90% if patients who already come with a compatible donor are included in the exchange pool.

Concluding, living donor kidney exchanges in combination with the establishment of kidney exchange pools can solve the large demand of kidney transplants.

A Appendix

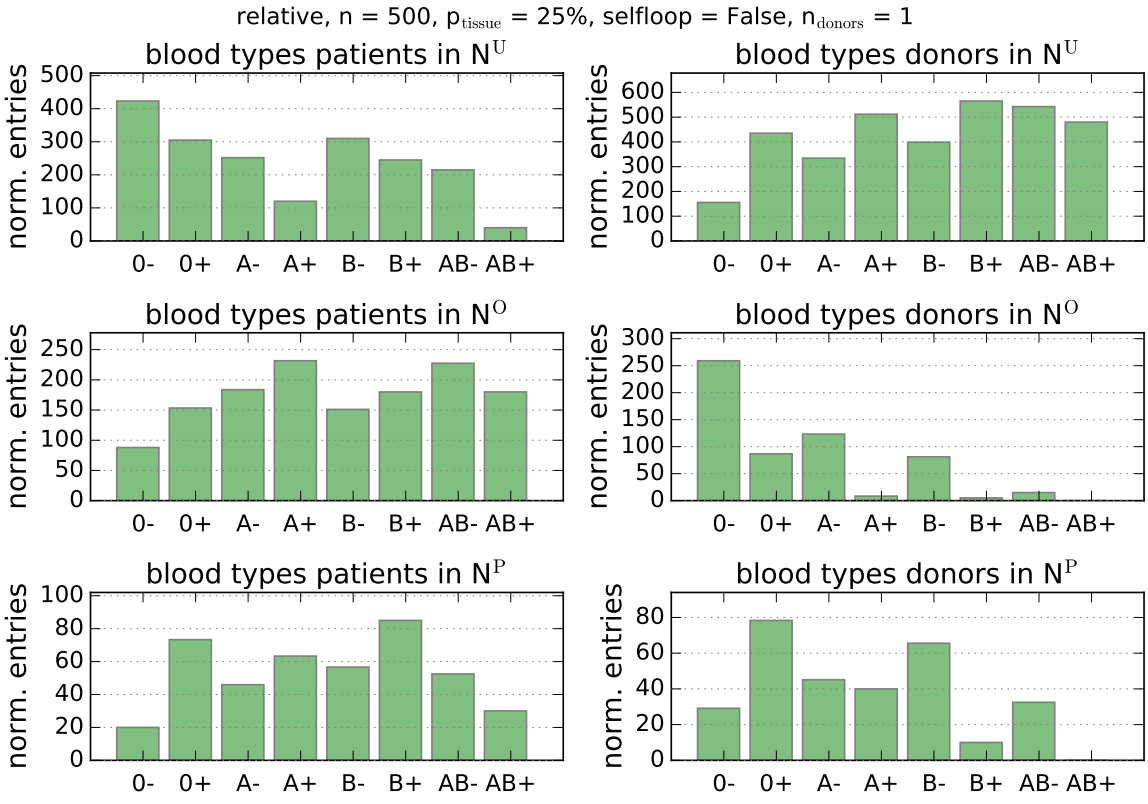


Figure A.1: Same as Fig. 4.7 but for a data sets with 500 incompatible patient-donor pairs, a 25% probability of tissue type compatibility and only one donor per patient.

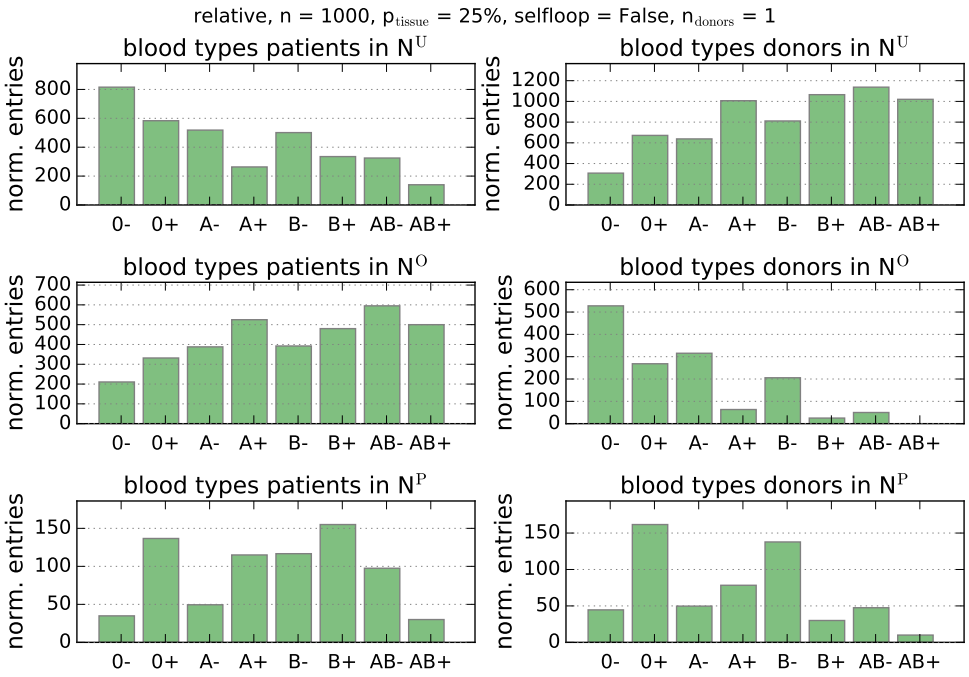


Figure A.2: Same as Fig. 4.7 but for a data sets with 1000 incompatible patient-donor pairs, a 25% probability of tissue type compatibility and only one donor per patient.

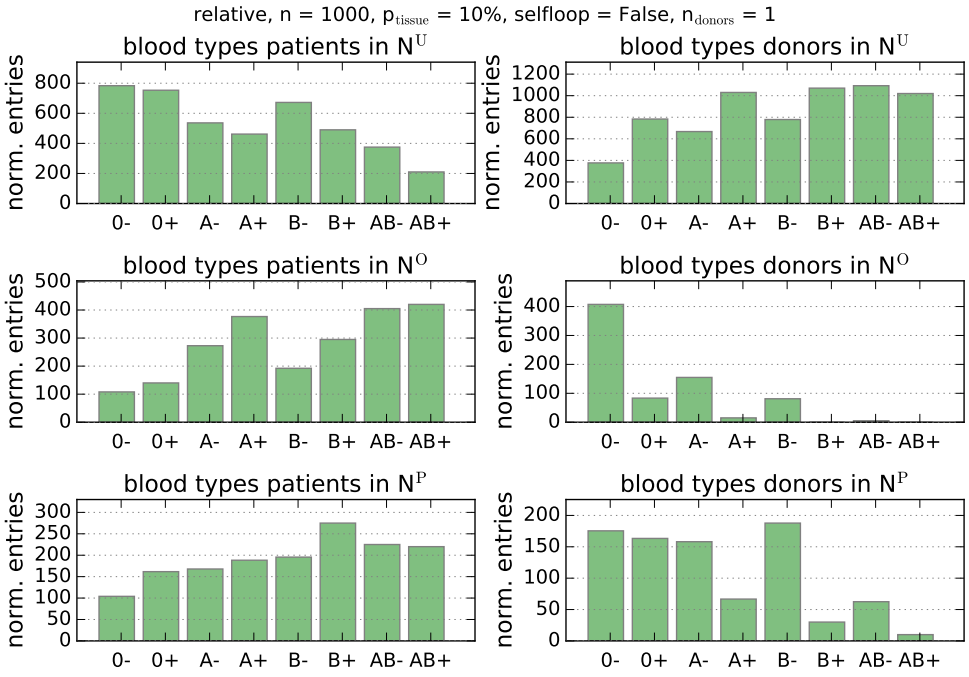


Figure A.3: Same as Fig. 4.7 but for a data sets with 1000 incompatible patient-donor pairs, a 10% probability of tissue type compatibility and only one donor per patient.

References

[1] Eurotransplant, *Statistics from the eurotransplant registration and allocation system*, 2015, http://statistics.eurotransplant.org.

[2] Deutsche Stiftung Organtransplantation, *Niere – Warteliste und Vermittlung*, online article (last visit Feb 12, 2016), http://www.dso.de/organspende-und-transplantation/warteliste-und-vermittlung/niere.html.

[3] Deutsche Stiftung Organtransplantation (DSO), *Organspende und Transplantation in Deutschland*, Jahresbericht, 2014.

[4] A. E. Roth, T. Sönmez, and M. Utku Ünver, „Pairwise kidney exchange“, Journal of Economic Theory **125**, 151–188 (2005), http://dx.doi.org/10.1016/j.jet.2005.04.004.

[5] K. Sack, *60 lives, 30 kidneys, all linked*, New York Times, Feb. 2012, http://www.nytimes.com/2012/02/19/health/lives-forever-linked-through-kidney-transplant-chain-124.html.

[6] J. Med, *Neues Verfahren: Nierentransplantation trotz Blutgruppenunverträglichkeit und positiver Kreuzprobe*, Nov. 2006, http://www.journalmed.de/newsview.php?id=15635.

[7] Transplantationszentrum Freiburg, online article (last visit Dec 20, 2015), http://www.transplantationszentrum-freiburg.de.

[8] G. Opelz, „Impact of hla compatibility on survival of kidney transplants from unrelated live donors“, Transplantation **64**, 1473–1475 (1997), http://dx.doi.org/10.1097/00007890-199711270-00017.

[9] D. W. Gjertson and J. M. Cecka, „Living unrelated donor kidney transplantation“, Kidney International **58**, 491–499 (2000), http://dx.doi.org/10.1046/j.1523-1755.2000.00195.x.

[10] Deutsches Referenzzentrum für Ethik in den Biowissenschaften, *Judgment of the federal social court on the admission of a cross-over living donation*, online article (last visit Feb 14, 2016), http://www.drze.de/in-focus/organ-transplantation/modules/urteil-des-bundessozialgerichts-ueber-die-zulassung-einer-ueberkreuz-lebendspende?set_language=en.

[11] Bundessozialgericht (BSG), „Urteil vom 10.12.2003. Az. B9 VS 1/01 R“, Juristenzeitung, 464–469 (2004).

[12] J. Edmonds, „Paths, trees, and flowers“, Canad. J. Math. **17**, 449–467 (1965), http://dx.doi.org/10.4153/CJM-1965-045-4.

[13] W. J. Cook et al., *Combinatorial Optimizaiton* (Wiley, 1998).

[14] C. Berge, „Two theorems in graph theory“, Proceedings of the National Academy of Sciences of the United States of America **43**, 842–844 (1957).

[15] G. van Rossum, *Python language*, www.python.org.

[16] *Networkx*, http://networkx.github.io/.

[17] *Matplotlib*, http://matplotlib.org/.

[18] J. Edmonds, „Maximum matching and a polyhedron with 0,1-vertices“, Journal of Research National Bureau of Standards Section B **69**, 125–130 (1965).

[19] Z. Galil, „Efficient algorithms for finding maximum matching in graphs“, ACM Comput. Surv. **18**, 23–38 (1986), http://doi.acm.org/10.1145/6462.6502.

Erklärung

Hiermit versichere ich, dass ich diese Arbeit einschließlich beigefügter Zeichnungen, Darstellungen und Tabellen selbstständig angefertigt und keine anderen als die angegebenen Hilfsmittel und Quellen verwendet habe. Alle Stellen, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, habe ich in jedem einzelnen Fall unter genauer Angabe der Quelle deutlich als Entlehnung kenntlich gemacht.

Aachen, den 14. Februar 2016

Christian Glaser