Pairwise kidney exchange

Alvin E. Rotha,b,∗, Tayfun Sonmezc, M. Utku Unverd (2005)

Motivation and contribution

- **Kidney exchange is important:** 60,752 patients on waiting list in US (2005)
- **A first step**: is pairwise exchange (since surgeons has to operate simultaneously)
- **Authors show that**, although this constraint eliminates some potential exchanges, there is a wide class of constrained-efficient mechanisms that are strategy-proof when patient donor pairs and surgeons have 0–1 preferences
- **Provides:**
  - **Deterministic mechanisms:** that accommodate the priority setting that organ banks currently use to allocate cadaver organs, (main focus of this presentation)
  - **Stochastic mechanisms:** that allow distributive justice issues to be addressed.

About the Problem

Assumptions

- A1: Patients are either happy (matched), or unhappy (unmatched) in terms of utility (0-1) preferences.
- A2: Only pairwise exchanges takes place

The goal of the mechanism:

Efficiency:

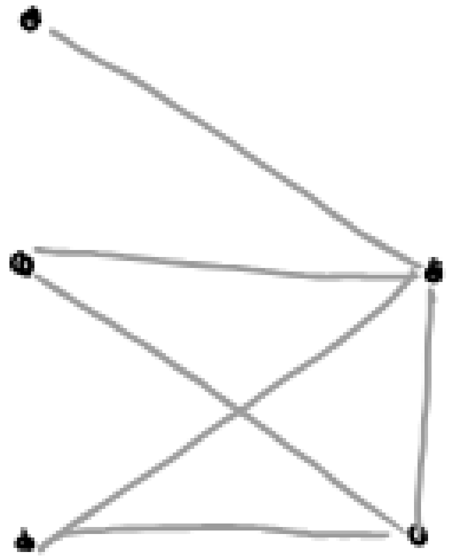
- Because of A1 pareto efficiency becomes a question of how many matched (happy) patients a given mechanism results in.
- An efficient mechanism hence results in max nr possible matches

Strategy proofness:

- We need patients not to have incentive to participate truthfully, not hide potential donors, or reject offers.

Representation of problem

- N is a set of n patients with incompatible donors
- R is a symmetric matrix n*n, where 1 symbolizes a match and 0 not a match.
- Consider a graph G based on R
- Each node is a patient with donors
- An edge between nodes shows a compatible match for kidney donor exchange.



Imposing priority

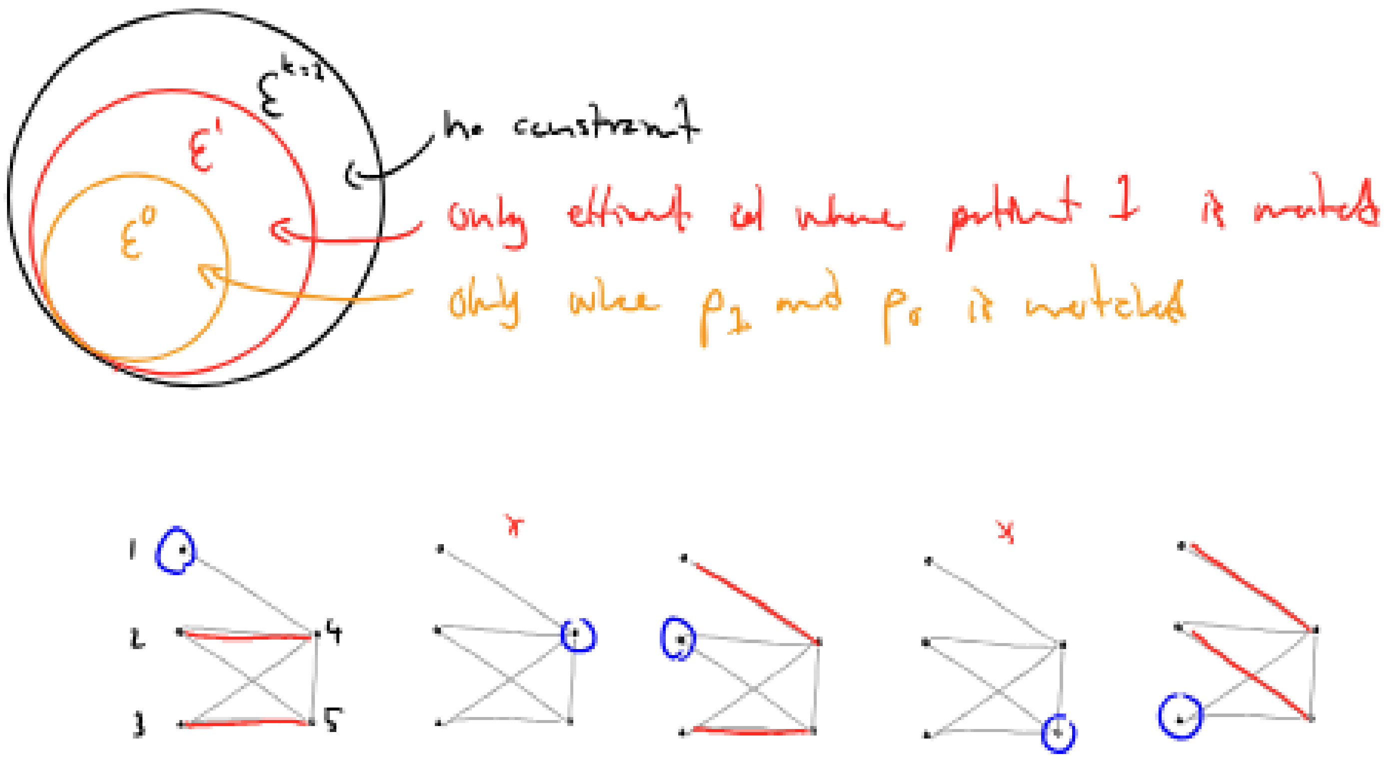
- In real life patients have different priority, based on how rare a match would be given their liver type. The authors therefore show how a priority can be imposed.
- Epsilon is a set of matches mu
- Authors propose greedily imposing preferences in the order of priority:

A priority mechanism produces a matching as follows, for any problem (N, R) and priority ordering $(1, 2, \ldots, n)$ among the patients:

- Let $\mathcal{E}^0 = \mathcal{M}$ (i.e. the set of all matchings).
- In general for $k \leqslant n$, let $\mathcal{E}^k \subseteq \mathcal{E}^{k-1}$ be such that

$$\mathcal{E}^k = \begin{cases} \{\mu \in \mathcal{E}^{k-1} : \mu(k) \neq k\} & \text{if } \exists \mu \in \mathcal{E}^{k-1} \text{ s.t. } \mu(k) \neq k, \\ \mathcal{E}^{k-1} & \text{otherwise.} \end{cases}$$

Imposing priority



Pareto Efficiency

- The collectable sets of matchable patients behave like the independent sets of a matroid. (Proposition 1)
- Matroids have the property that greedy algorithms (like priority mechanisms) find maximal cardinality, hence Pareto-efficient, independent sets [39] [20].
- Therefore, giving a higher priority patient a kidney never causes more than one lower-priority patient to lose out, this is a structural property of matroids.

For a given problem (N, R) and priority ordering $(1, 2, \ldots, n)$, we refer to each matching in $\mathcal{E}^n$ as a *priority matching*, and a *priority mechanism* is a function which selects a priority

A *matroid* is a pair $(X, \mathcal{I})$ such that X is a set and $\mathcal{I}$ is a collection of subsets of X (called the *independent sets*) such that

M1. if I is in $\mathcal{I}$ and $J \subset I$ then J is in $\mathcal{I}$; and
M2. if I and J are in $\mathcal{I}$ and $|I| > |J|$ then there exists an $i \in I \backslash J$ such that $J \cup \{i\}$ is in $\mathcal{I}$.

Proposition 1. *Let $\mathcal{I}$ be the sets of simultaneously matchable patients, i.e. $\mathcal{I} = \{I \subseteq N : \exists \mu \in \mathcal{M}$ such that $I \subseteq M_\mu\}$. Then $(N, \mathcal{I})$ is a matroid.*

[39] R. Rado, Note on independence functions, Proc. London Math. Soc. 7 (1957) 300–320.

[20] J. Edmonds, Matroids and the greedy algorithm, Math. Programming 1 (1971) 127–136.

GED Decomposition

- Recognize that we don't care about priority if we don't run any risk of being unmatched NP,NO, we therefore decompose the graph to only focus on those patients with this risk NU and those connected to them NO.
- We run GED to decompose into:

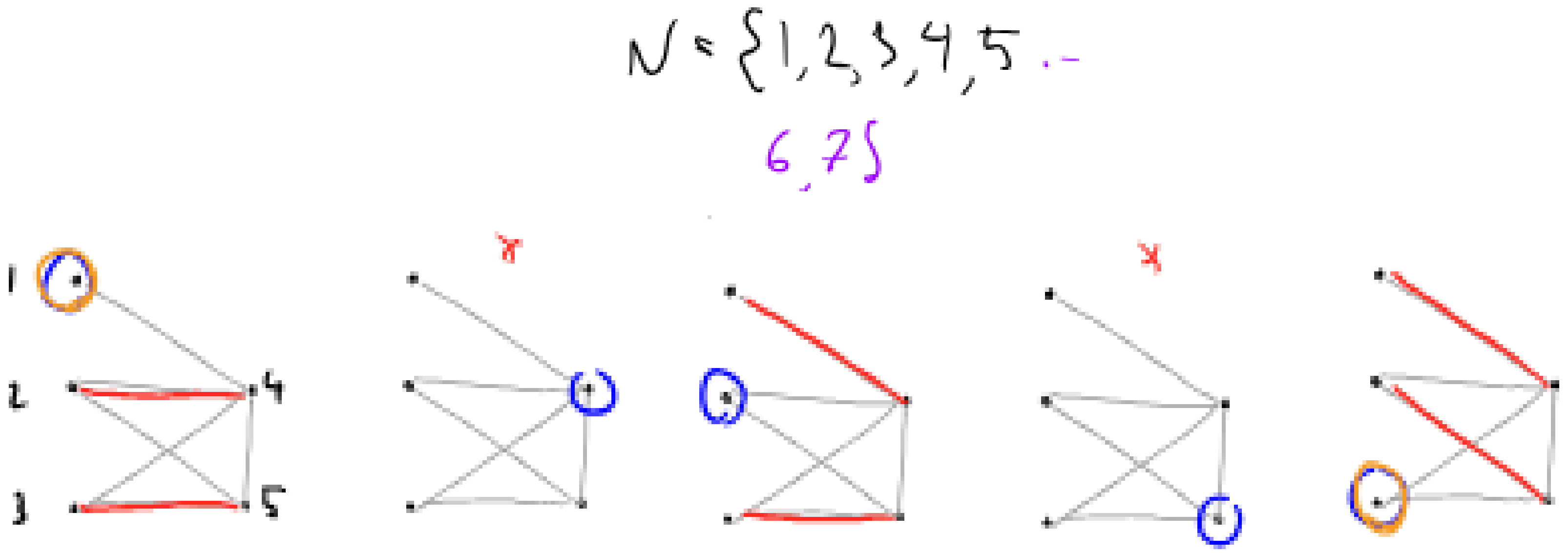
$$N^{\mathrm{U}} = \{i \in N : \exists \mu \in \mathcal{E} \text{ s.t. } \mu(i) = i\},$$

$$N^{\mathrm{O}} = \left\{i \in N \backslash N^{\mathrm{U}} : \exists j \in N^{\mathrm{U}} \text{ s.t. } r_{i,j} = 1\right\}, \text{ and}$$

$$N^{\mathrm{P}} = N \backslash \left(N^{\mathrm{U}} \cup N^{\mathrm{O}}\right).$$

GED Decomposition: Example

- Run GED on graph example:
- Find NU:



GED Decomposition: Example

- Run GED on graph example:
- Find NO and NP as well:

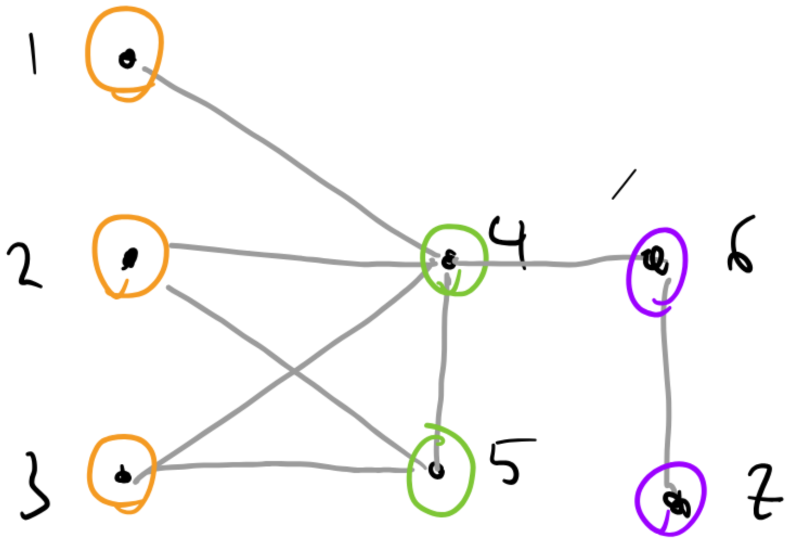
$N = \{1, 2, 3, 4, 5, \ldots 6, 7\}$

Summary:

$N^v = \{1, 2, 3\}$

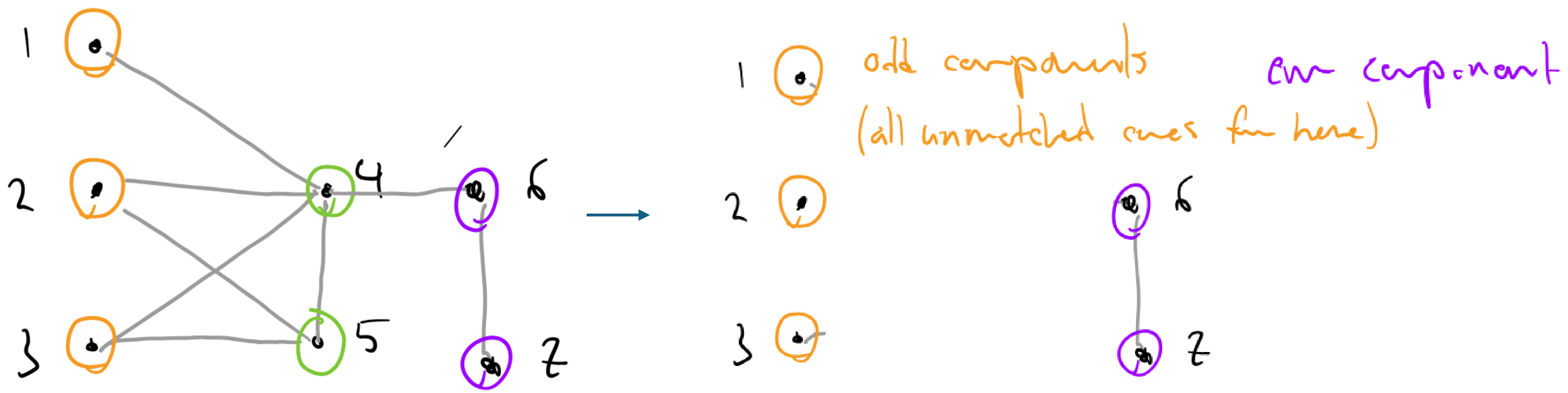
$N^a = \{4, 5\}$

$N^p = \{6, 7\}$



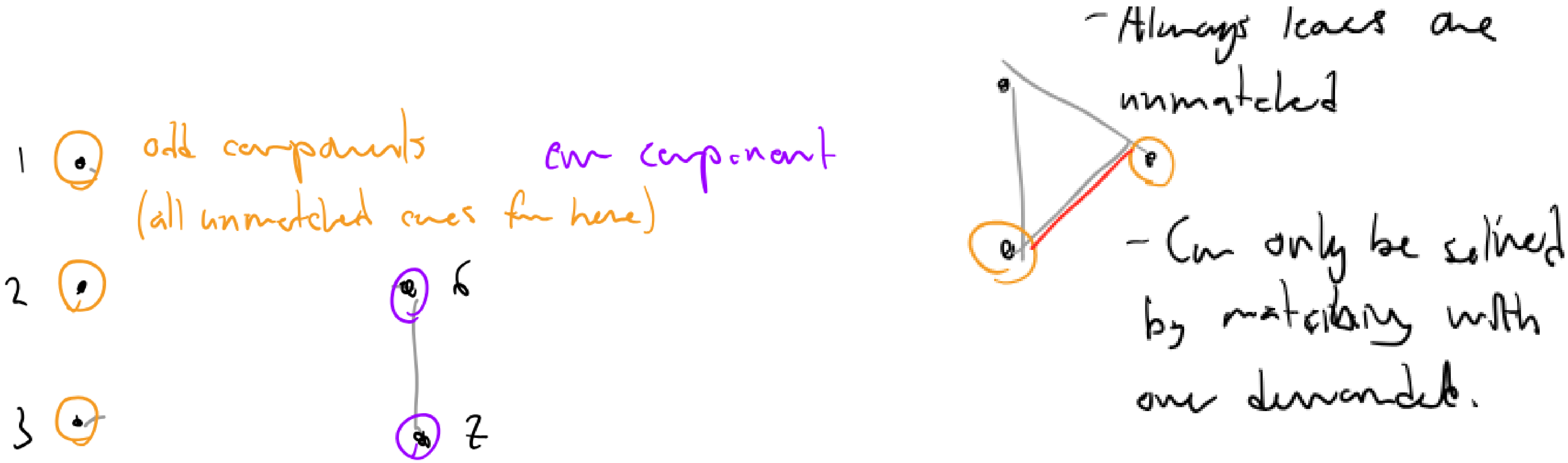
GED Decomposition: Example

- Removing No reveals graph components, we have even and odd components depending on nr vertices:



GED Decomposition: Example

- All our odd component in this example is 1
- Odd components can also be larger f.ex 3,



GED Decomposition: Observations

Constraining the order problem further by removing NP

- We observe Np will be matched no matter what, they don't care about the imposed order, and are only matches amongst themselves. We can remove Np from the ordering problem

Order and competition

- The order solves two types of competition:
  - For odd components to get matched with a No patient
  - For patients in unmatched odd components to not be the one who remains unmatched.

$$N^{\mathrm{U}} = \{i \in N : \exists \mu \in \mathcal{E} \text{ s.t. } \mu(i) = i\},$$
$$N^{\mathrm{O}} = \left\{i \in N \setminus N^{\mathrm{U}} : \exists j \in N^{\mathrm{U}} \text{ s.t. } r_{i,j} = 1\right\}, \text{ and}$$
$$N^{\mathrm{P}} = N \setminus \left(N^{\mathrm{U}} \cup N^{\mathrm{O}}\right).$$

GED Decomposition: Efficiency

- The GED Lemma allows us to see in detail how competition for compatible kidneys plays out in priority mechanisms.
- The outcome of a priority mechanism is Pareto-efficient
- and by the GED Lemma, each over demanded as well as each perfectly matched patient is matched at each Pareto-efficient matching

GED Lemma, (Lemma 2 closely follows)

Lemma 2. *(Gallai–Edmonds Decomposition Lemma) Let (I, R_I) be the reduced subproblem with $I = N \backslash N^O$ and let μ be a Pareto-efficient matching for the original problem (N, R).*

1. *For any patient $i \in N^O$, $\mu(i) \in N^U$.*
2. *For any even component (J, R_J) of (I, R_I), $J \subseteq N^P$ and for any patient $i \in J$, $\mu(i) \in J \backslash \{i\}$.*
3. *For any odd component (J, R_J) of (I, R_I), $J \subseteq N^U$ and for any patient $i \in J$ it is possible to match all remaining patients in J with each other (so that any patient $j \in J \backslash \{i\}$ can be matched with a patient in $J \backslash \{i, j\}$). Moreover for any odd component (J, R_J), either*
   (a) *one and only one patient $i \in J$ is matched with a patient in N^O under the Pareto-efficient matching μ whereas all remaining patients in J are matched with each other so that $\mu(j) \in J \backslash \{i, j\}$ for any patient $j \in J \backslash \{i\}$, or*
   (b) *one patient $i \in J$ remains unmatched under the Pareto-efficient matching μ whereas all remaining patients in J are matched with each other so that $\mu(j) \in J \backslash \{i, j\}$ for any patient $j \in J \backslash \{i\}$.*

See Lovasz and Plummer [31] for a proof of the GED Lemma.

GED Decomposition: Efficiency

For an efficient Pareto matching:

- Match internally in NP
- Each patient in NO is matched with patient in NU
- In each odd component
  - One is either matched with NO
  - Or remains unmatched
  - Remaining patients are matched amongst themselves

GED Lemma, (Lemma 2 closely follows)

Lemma 2. *(Gallai–Edmonds Decomposition Lemma) Let (I, R_I) be the reduced subproblem with $I = N \backslash N^{\mathrm{O}}$ and let μ be a Pareto-efficient matching for the original problem (N, R).*

1. *For any patient $i \in N^{\mathrm{O}}$, $\mu(i) \in N^{\mathrm{U}}$.*
2. *For any even component (J, R_J) of (I, R_I), $J \subseteq N^{\mathrm{P}}$ and for any patient $i \in J$, $\mu(i) \in J \backslash \{i\}$.*
3. *For any odd component (J, R_J) of (I, R_I), $J \subseteq N^{\mathrm{U}}$ and for any patient $i \in J$ it is possible to match all remaining patients in J with each other (so that any patient $j \in J \backslash \{i\}$ can be matched with a patient in $J \backslash \{i, j\}$). Moreover for any odd component (J, R_J), either*
   - (a) *one and only one patient $i \in J$ is matched with a patient in N^{O} under the Pareto-efficient matching μ whereas all remaining patients in J are matched with each other so that $\mu(j) \in J \backslash \{i, j\}$ for any patient $j \in J \backslash \{i\}$, or*
   - (b) *one patient $i \in J$ remains unmatched under the Pareto-efficient matching μ whereas all remaining patients in J are matched with each other so that $\mu(j) \in J \backslash \{i, j\}$ for any patient $j \in J \backslash \{i\}$.*

See Lovasz and Plummer [31] for a proof of the GED Lemma.

Deterministic (priority) mechanism on GED

Algorithm with GED : Imposing priority

- An iterative procedure was proposed under the priority mechanism to determine what odd components will have all its members matched.
- It is essentially an implementation of the GED Lemma 2

Algorithm

Step 1: If $\left|C\left(\{D_1\}, N^{\mathrm{O}}\right)\right| \geqslant |\{D_1\}| = 1$, then let $\mathcal{J}_1 = \{D_1\}$ and in this case all members of D_1 will be matched. If $\left|C\left(\{D_1\}, N^{\mathrm{O}}\right)\right| < |\{D_1\}| = 1$, then let $\mathcal{J}_1 = \emptyset$ and in this case all members of D_1 except its lowest priority patient ℓ_1 will be matched.

In general, at

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

Algorithm with GED : Imposing priority

Pseudo Code of Algorithm for Readability

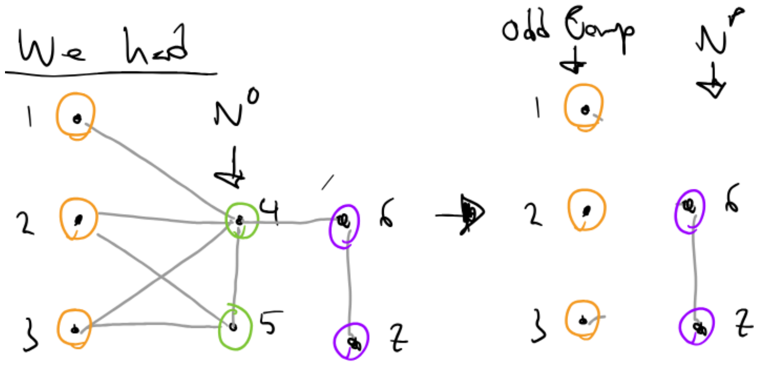
$\mathcal{J} = \{1, 2, 3\}$ (sorted by highest member)
$\mathcal{J}_k = \{\}$
$N^o = \{4, 5\}$ (Set of helpers/resources)
for count k, D_k in $\mathcal{J}$: (D_k is the current task being considered) **do**
  if $k = 1$ **then**
    if ($\exists N^o$ can help) **then**
      $\mathcal{J}_1 = \{D_1\}$
    else
      $\mathcal{J}_1 = \emptyset$
    end if
  else
    if we help D_k can we still help all in $\mathcal{J}_{k-1}$ based on N^o? **then**
      $\mathcal{J}_k = \mathcal{J}_{k-1} \cup \{D_k\}$ (All in D_k matched)
    else
      $\mathcal{J}_k = \mathcal{J}_{k-1}$ (D_k cannot be helped; D_k remains unmatched)
    end if
  end if
end for

Algorithm

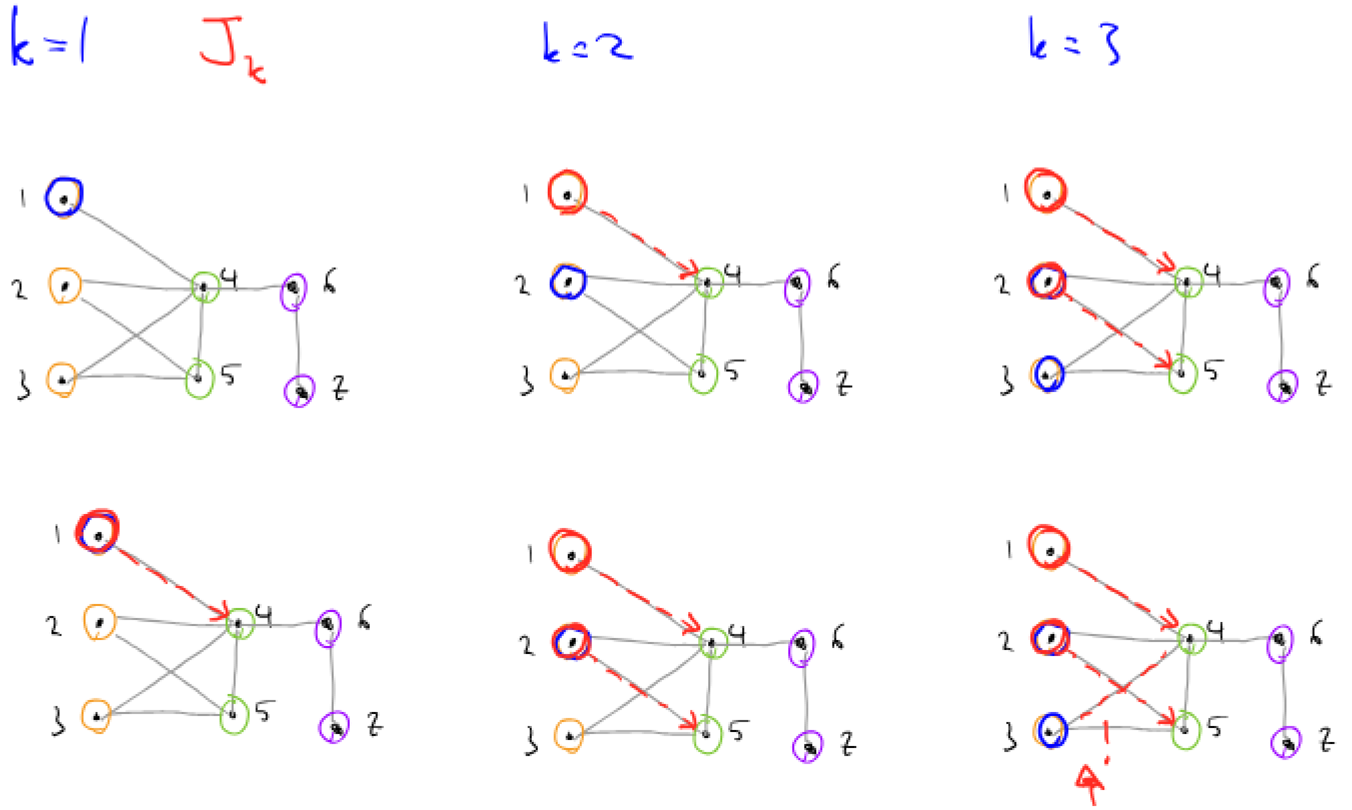
Step 1: If $|C(\{D_1\}, N^O)| \geqslant |\{D_1\}| = 1$, then let $\mathcal{J}_1 = \{D_1\}$ and in this case all members of D_1 will be matched. If $|C(\{D_1\}, N^O)| < |\{D_1\}| = 1$, then let $\mathcal{J}_1 = \emptyset$ and in this case all members of D_1 except its lowest priority patient ℓ_1 will be matched.

In general, at

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



Algorithm with GED : Imposing priority



We can not match without breaking previous matching!,
Component remains unmatched

Strategy proofness:

Can I benefit from hiding possible donors?

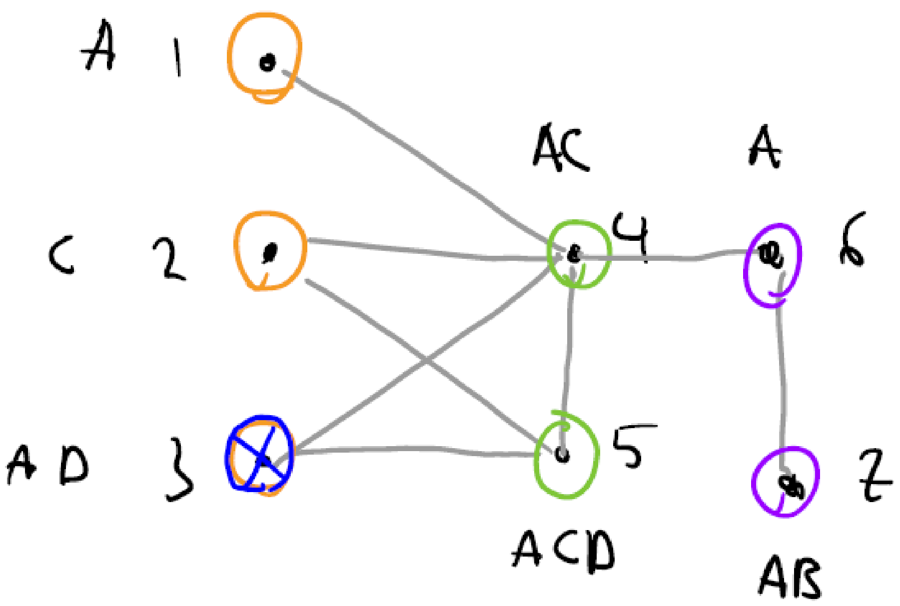
- No, more donors = more or equal nr edges leading to your node, sine we can match with more livers.
- More possible matches = higher probability of being selected by higher priority patient for swap.
- (we have donor monotonicity)

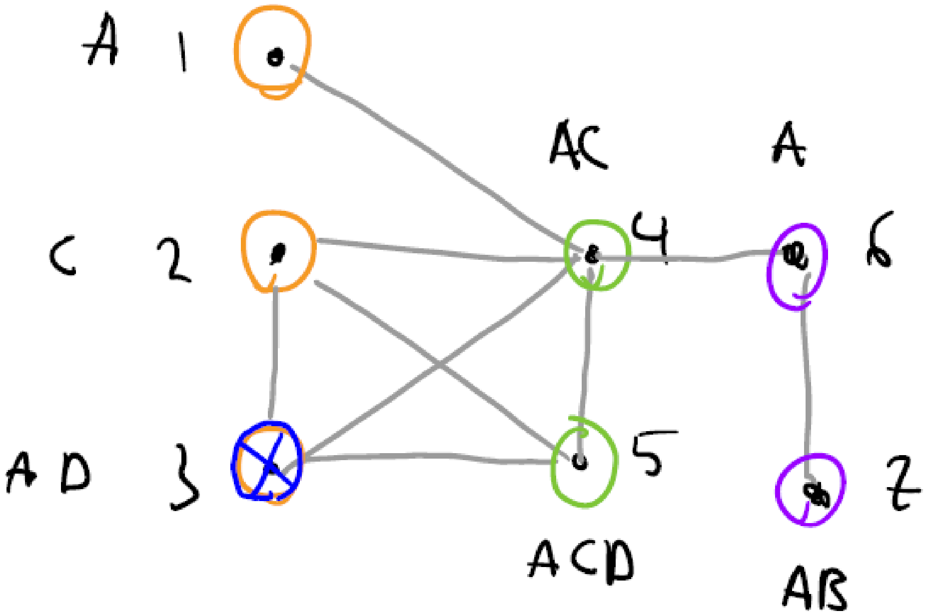
Or rejecting a liver type?

- No again this will reduce the number of edges leading to your node. Also this violates our 0-1 preference assumption A1

Strategy proofness: Example

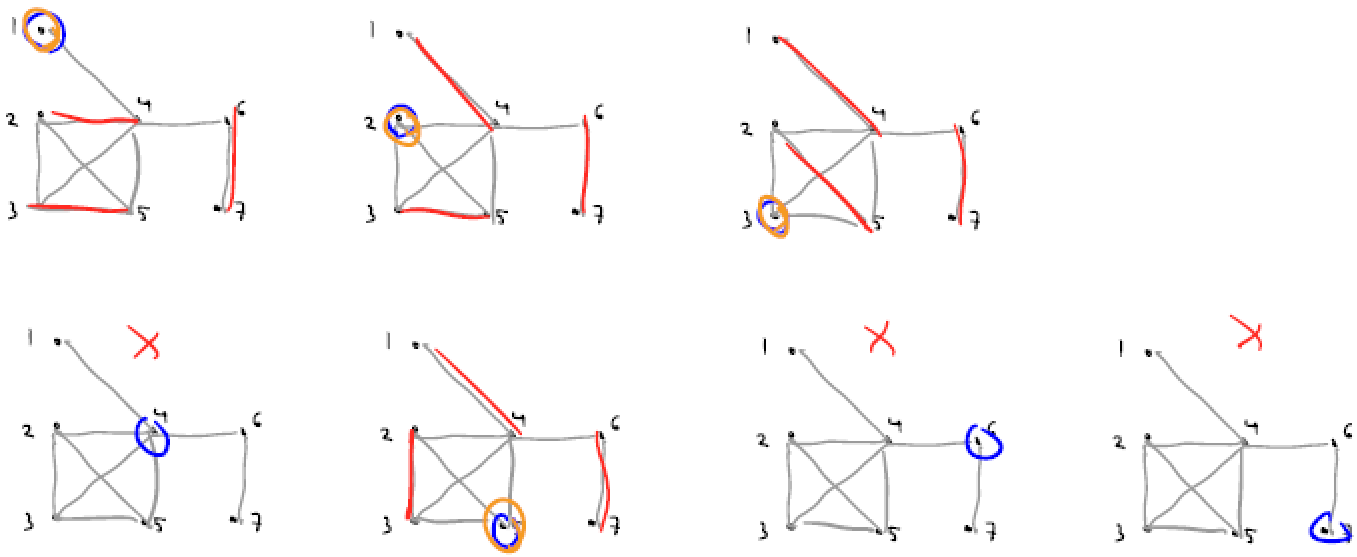
- Consider nr 3 adds a donor type C, creating new connection to C





Strategy proofness: Example

RUN GED:



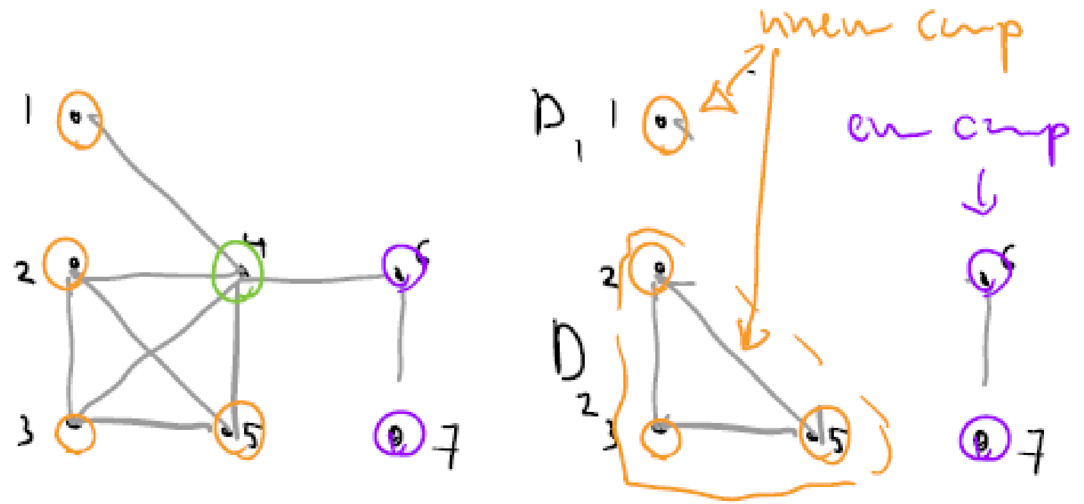
Strategy proofness: Example

RUN GED:

$N^V = \{1, 2, 3, 5\}$

$N^O = \{4\}$

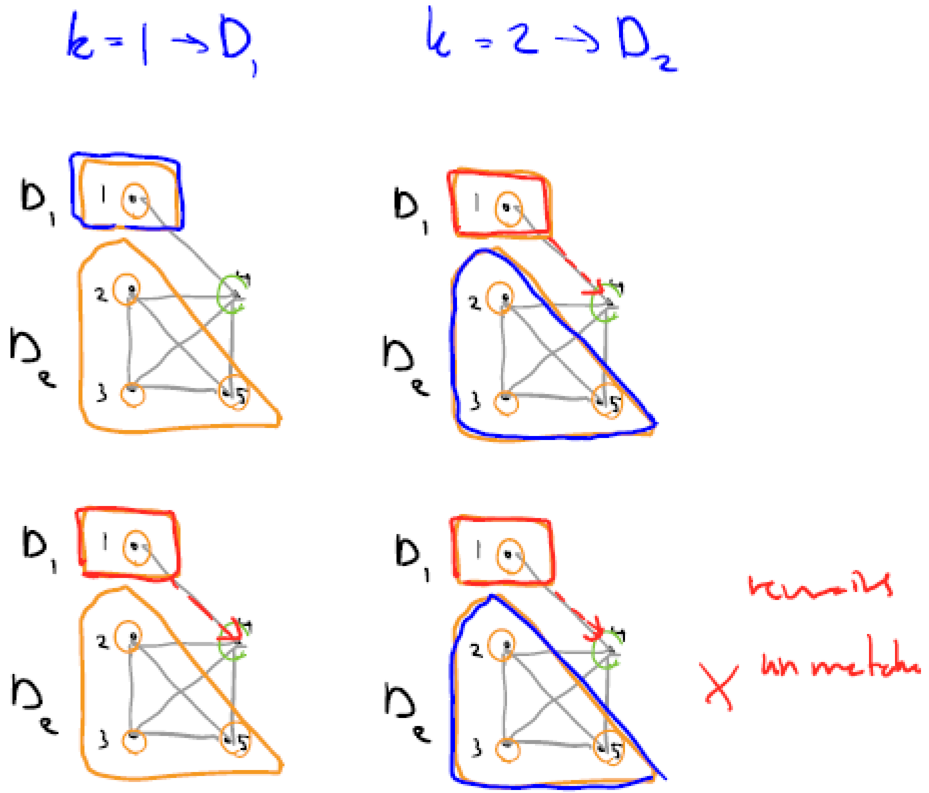
$N^A = \{6, 7\}$



Strategy proofness: Example

RUN PRIORITY ALGO:

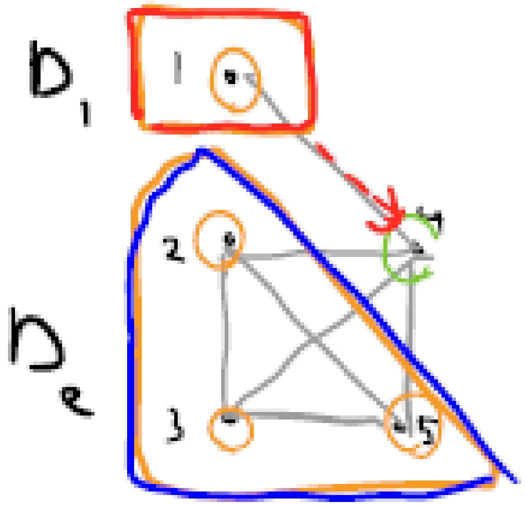
$$J = \{D_1, D_2\}$$



Strategy proofness: Example

RUN PRIORITY ALGO:

- In D2 = {2,3,5}, only 5 will be unmatched
- Adding connection made 3 go from unmatched to now matched!



Corollary 2. *A priority mechanism is donor-monotonic.*

Strategy proofness : Proof

- Hiding donors, can only remove possible matches.
- This set Q (with hidden donors) of matches is a subset of the previous R
- and hence if no match was found in previous R (otherwise you would have chosen to match given your preference), no match will be found in subset Q.
- **You have no incentive to hide donors**

Theorem 1. *A priority mechanism makes it a dominant strategy for a patient to reveal both* (a) *her full set of acceptable kidneys; and* (b) *her full set of available donors.*

Proof of Theorem 1(a). W.l.o.g. we will prove the theorem for the priority mechanism ϕ induced by the natural ordering. Let R be a reduced problem and k be a patient. If patient k is matched with another patient under $\phi(R)$, then she has nothing to gain by revealing only a subset of her full set of compatible kidneys. Suppose patient k remains unmatched under $\mu = \phi(R)$ and let Q be a reduced problem obtained from R by patient k declaring some of her compatible kidneys to be incompatible. Observe that this implies $\mathcal{E}^{k-1}(Q) \subseteq \mathcal{E}^{k-1}(R)$. Let $\phi(Q) = \nu$. Since $\mu(k) = k$, $\mu'(k) = k$ for all $\mu' \in \mathcal{E}^{k-1}(R)$. But then $\mu'(k) = k$ for all $\mu' \in \mathcal{E}^{k-1}(Q)$ as well and hence $\nu(k) = k$ completing the proof. □

Concluding Remarks

Deterministic (Priority) Mechanism:

- A priority-based deterministic mechanism can be constructed under realistic constraints (pairwise-only exchange, 0–1 preferences)
- It always selects a Pareto-efficient (maximum-cardinality) matching
- It is strategy proof: no patient benefits from hiding donors or rejecting compatible kidneys, under this specific constraints (key contribution)
- It can impose any priority structure (similar to current cadaveric organ allocation) without sacrificing efficiency or strategy-proofness.

Stochastic (egalitarian) mechanism on GED
Brief remarks summary

Egalitarian stochastic case:

- **If we think that imposing priority is inefficient and unfair, we could take an alternative path:**
- **New Goal:** Maximize the utility (probability of matching) for the least fortunate patients (distributive justice/fairness)
- **Egalitarian Utility Profile**: This profile defines the maximum achievable probability of being matched for every patient, under the fairest possible distribution.
- **The Mechanism:** The mechanism itself is an internal lottery or a probabilistic matching scheme. Instead of generating one final set of matches, it generates a set of different maximum-size matchings and selects one of them according to a specific probability distribution (the lottery).

Egalitarian stochastic case:

- The probability of being selected depends on the graphs GED structure, who is under demanded and over demanded

Example Rule of thumb for min utility:

- If odd component D is not saved by No, one in D goes unmatched , **min utility is:**

$$\text{Utility } (u^E) = \frac{\text{Number of Matched Positions}}{\text{Total Number of Patients in } D} = \frac{|D|-1}{|D|}$$

- Hence min egalitarian utility value for an odd component with 3 members is 2/3, and 5 members 4/5 etc. A more advanced algorithm is used to calculate the final egalitarian profile. That Lorentz dominates ie most evenly distributes probabilities

A.E. Roth et al. / Journal of Economic Theory 125 (2005) 151–188

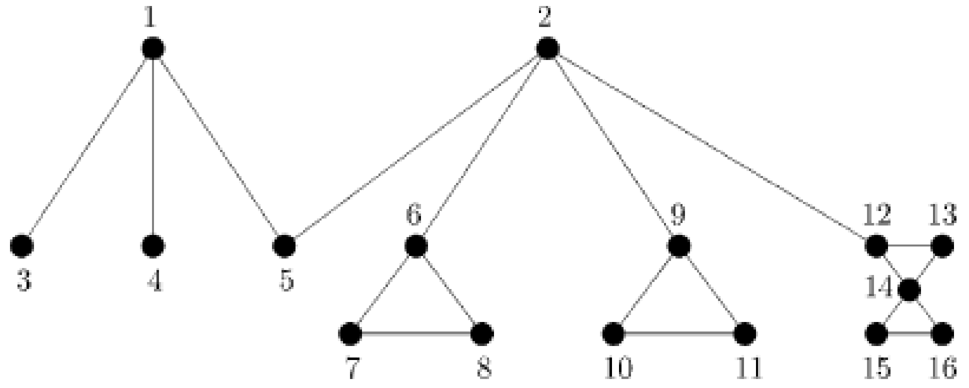


Fig. 1. Graphical Representation for Example 2.

$$u^{\mathrm{E}} = (1, 1, \tfrac{1}{2}, \tfrac{1}{2}, \tfrac{5}{7}, \tfrac{5}{7}, \tfrac{5}{7}, \tfrac{5}{7}, \tfrac{5}{7}, \tfrac{5}{7}, \tfrac{5}{7}, \tfrac{4}{5}, \tfrac{4}{5}, \tfrac{4}{5}, \tfrac{4}{5}, \tfrac{4}{5})$$

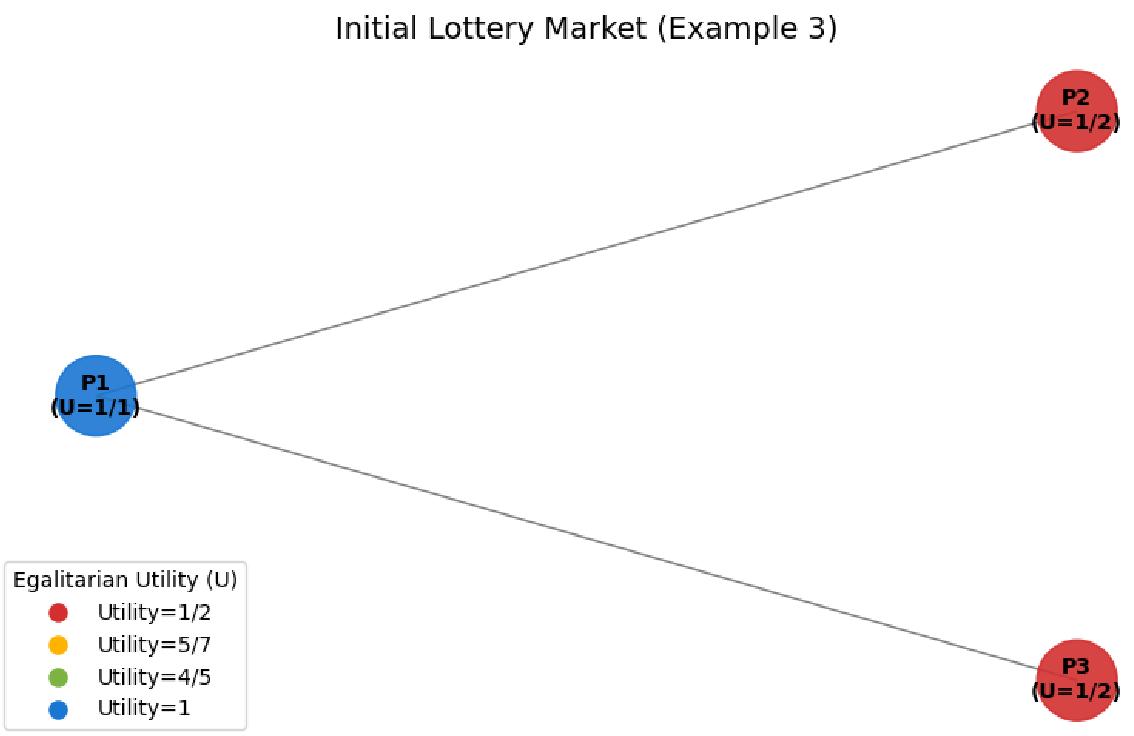
Egalitarian stochastic case:

Example 3. Let $N = \{1, 2, 3, 4\}$ and suppose patient 1 is mutually compatible with patient 2 as well as with patient 3 but patients 2 and 3 are not mutually compatible. The two Pareto efficient matchings are

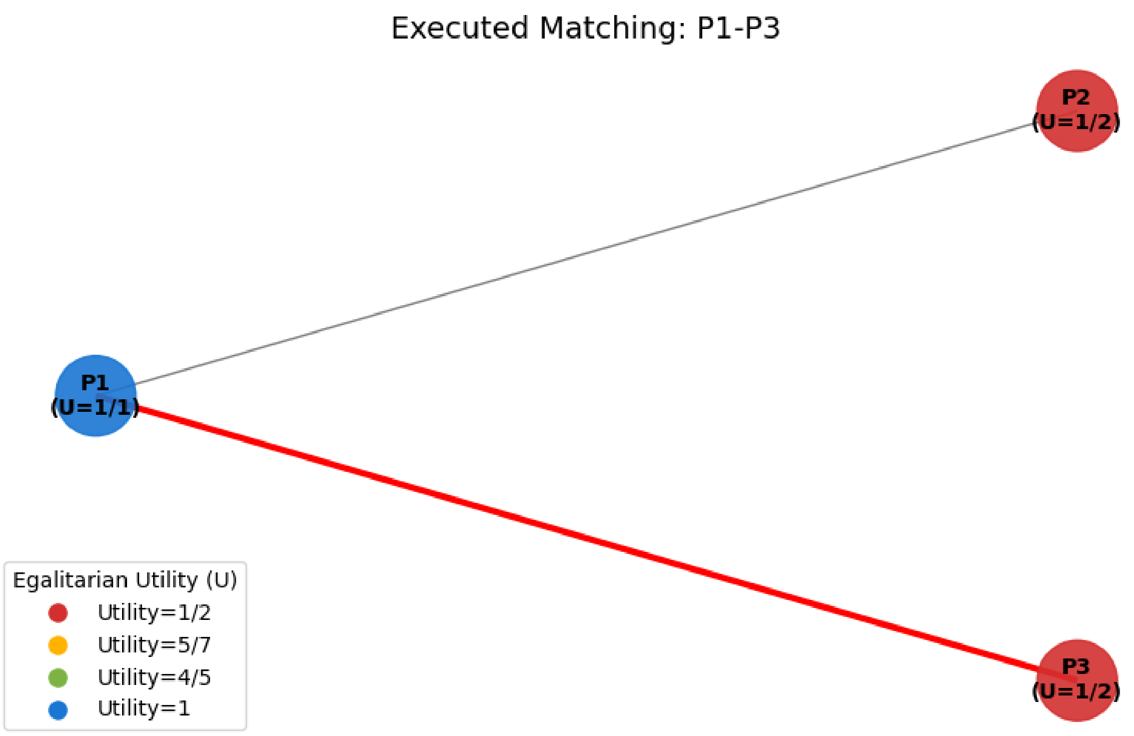
$$\mu = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix} \quad \text{and} \quad \nu = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix}$$

	Patient 1	Patient 2	Patient 3
Egalitarian Utility (u^E):	1	$\frac{1}{2}$	$\frac{1}{2}$
Efficient Matching μ:	$\{(1,2)\}$	Matched	Unmatched
Efficient Matching μ':	$\{(1,3)\}$	Unmatched	Matched

Egalitarian stochastic case:



Egalitarian stochastic case:



Concluding Remarks

Deterministic (Priority) Mechanism:

- A priority-based deterministic mechanism can be constructed under realistic constraints (pairwise-only exchange, 0–1 preferences)
- It always selects a Pareto-efficient (maximum-cardinality) matching
- It is strategy-proof: no patient benefits from hiding donors or rejecting compatible kidneys
- It can impose any priority structure (similar to current cadaveric organ allocation) without sacrificing efficiency or strategy-proofness.

Extension to egalitarian mechanism

- The authors extend the framework to a stochastic mechanism
- Instead of using priorities, it maximizes the egalitarian (Lorenz-dominant) utility profile, the fairest possible distribution of transplant probabilities.
- It remains efficient (lotteries only over maximum matchings) and strategy-proof.

Pareto Efficiency

- By construction, a priority matching is maximal and hence Pareto Efficient
- Proposition 1 implies, through the second property of matroids, that the "opportunity cost" of matching a higher priority patient will never be more than one lower priority patient who could otherwise have been matched.

For a given problem (N, R) and priority ordering $(1, 2, \ldots, n)$, we refer to each matching in $\mathcal{E}^n$ as a *priority matching*, and a *priority mechanism* is a function which selects a priority

A *matroid* is a pair $(X, \mathcal{I})$ such that X is a set and $\mathcal{I}$ is a collection of subsets of X (called the *independent sets*) such that

M1. if I is in $\mathcal{I}$ and $J \subset I$ then J is in $\mathcal{I}$; and
M2. if I and J are in $\mathcal{I}$ and $|I| > |J|$ then there exists an $i \in I \backslash J$ such that $J \cup \{i\}$ is in $\mathcal{I}$.

Proposition 1. *Let $\mathcal{I}$ be the sets of simultaneously matchable patients, i.e. $\mathcal{I} = \{I \subseteq N : \exists \mu \in \mathcal{M}$ such that $I \subseteq M_\mu\}$. Then $(N, \mathcal{I})$ is a matroid.*

Independent set = any set of patients that can be matched *together* in some feasible matching.

Formally:

$$\mathcal{I} = \{I \subseteq N : \exists \text{ a matching } \mu \text{ such that every patient in } I \text{ is matched in } \mu\}.$$

In plain English:

> **An independent set is any collection of patients for which it is possible to find pairwise compatible exchanges so that all of them receive a kidney simultaneously.**

✅ Why this corresponds to “independent sets” in a matroid

In a matroid, independent sets are the subsets that are “feasible” to take together.

Here:

- The ground set is the set of all patients N.
- “Feasible to take together” = “can all be matched simultaneously.”
- So independence = **joint matchability**.

✅ Example to make it intuitive

Suppose you have patients $\{1, 2, 3, 4\}$ and compatibilities:

- 1 compatible with 2
- 3 compatible with 4

Then:

- $\{1, 2\}$ is independent — they can be swapped.
- $\{3, 4\}$ is independent.
- $\{1, 2, 3, 4\}$ is independent — you can run two disjoint exchanges.
- $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$ are independent.

But:

- $\{1, 3\}$ might or might not be independent depending on compatibilities.
- A set like $\{1, 2, 3\}$ is independent if some feasible matching matches all three — which in pairwise exchange is impossible, so it would *not* be independent.

✅ Why this matters

Defining independence this way makes the set system $(N, \mathcal{I})$ satisfy:

1. **Hereditary property**
   A subset of matchable patients is also matchable.
2. **Exchange property**
   If one matchable set is larger than another, you can extend the smaller one by adding someone from the larger one.

And that’s exactly why it forms a **matroid**, which is why **the greedy priority algorithm returns a maximum-cardinality matching**.