ECO 426 (Market Design) - Lecture 4

Ettore Damiano

October 5, 2015

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)
  - A collection of individuals, A (agents)

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)
  - A collection of individuals, A (agents)
  - each agent $a \in A$:

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)
  - A collection of individuals, A (agents)
  - each agent $a \in A$:
    - owns a "house," h_a,

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)
    - A collection of individuals, A (agents)
    - each agent $a \in A$:
        - owns a "house," h_a, (H is the set of all houses);

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)
  - A collection of individuals, A (agents)
  - each agent $a \in A$:
    - owns a "house," h_a, (H is the set of all houses);
    - has (strict) preferences over the set of houses in the economy

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)
    - A collection of individuals, A (agents)
    - each agent $a \in A$:
        - owns a "house," h_a, (H is the set of all houses);
        - has (strict) preferences over the set of houses in the economy
    - the initial allocation might not be efficient (i.e. Pareto efficient)

Some allocation problems can be modelled as two-sided matching markets but with one side not having any preferences over the possible allocations

- Housing market (allocating houses to individuals)
    - A collection of individuals, A (agents)
    - each agent $a \in A$:
        - owns a "house," h_a, (H is the set of all houses);
        - has (strict) preferences over the set of houses in the economy
    - the initial allocation might not be efficient (i.e. Pareto efficient)
        - mutually beneficial trades might be possible

- Housing market vs. marriage market

- Housing market vs. marriage market
  - one side of the market (houses) has no preferences over matches;

- Housing market vs. marriage market
    - one side of the market (houses) has no preferences over matches;
    - agents have an initial endowment (i.e. each agent owns a house)

- Housing market vs. marriage market
  - one side of the market (houses) has no preferences over matches;
  - agents have an initial endowment (i.e. each agent owns a house)
    - the market starts from a default allocation where each agent is matched to his own house

- Housing market vs. marriage market
  - one side of the market (houses) has no preferences over matches;
  - agents have an initial endowment (i.e. each agent owns a house)
    - the market starts from a default allocation where each agent is matched to his own house
  - Goal: find a matching that cannot be improved

- Housing market vs. marriage market
  - one side of the market (houses) has no preferences over matches;
  - agents have an initial endowment (i.e. each agent owns a house)
    - the market starts from a default allocation where each agent is matched to his own house
  - Goal: find a matching that cannot be improved
    - it is not possible to reassign houses making some agent better off and making no agent worse off

- An allocation is an assignment (matching) of agents to houses such that

- An allocation is an assignment (matching) of agents to houses such that
  - each agent is assigned exactly one house; and

- An allocation is an assignment (matching) of agents to houses such that
  - each agent is assigned exactly one house; and
  - each house is assigned to exactly one agent.

- An allocation is an assignment (matching) of agents to houses such that
  - each agent is assigned exactly one house; and
  - each house is assigned to exactly one agent.
- An allocation in an housing market is described by a **"bijection"** $\mu : A \to H$.

- An allocation is an assignment (matching) of agents to houses such that
  - each agent is assigned exactly one house; and
  - each house is assigned to exactly one agent.
- An allocation in an housing market is described by a **"bijection"** $\mu : A \to H$.
- In a housing market, each agent is endowed (owns) one house (e.g. a owns h_a)

- An allocation is an assignment (matching) of agents to houses such that
  - each agent is assigned exactly one house; and
  - each house is assigned to exactly one agent.
- An allocation in an housing market is described by a **"bijection"** $\mu : A \to H$.
- In a housing market, each agent is endowed (owns) one house (e.g. a owns h_a)
- What allocations would we expect to arise if agents can freely dispose of their endowment?

- Agents in a group $S \subseteq A$ own together (in a "coalition") a subset of the houses in the market H_S

- Agents in a group $S \subseteq A$ own together (in a "coalition") a subset of the houses in the market H_S
- The agents in a coalition S can "independently" distribute the houses they own, H_S, among themselves.

- Agents in a group $S \subseteq A$ own together (in a "coalition") a subset of the houses in the market H_S
- The agents in a coalition S can "independently" distribute the houses they own, H_S, among themselves.
    - An assignment of the houses in H_S to agents in S, is an allocation in the housing market where the set of agents is S and the set of houses is H_S

- Agents in a group $S \subseteq A$ own together (in a "coalition") a subset of the houses in the market H_S
- The agents in a coalition S can "independently" distribute the houses they own, H_S, among themselves.
  - An assignment of the houses in H_S to agents in S, is an allocation in the housing market where the set of agents is S and the set of houses is H_S

$$\mu_S : S \rightarrow H_S.$$

- Definition (Blocking) A coalition of agents S **blocks** an allocation μ, if there is an assignment μ_S of the houses owned by the coalition to the members of the coalition S, such that:

- Definition (Blocking) A coalition of agents S **blocks** an allocation μ, if there is an assignment μ_S of the houses owned by the coalition to the members of the coalition S, such that:
  i) some member of S prefers μ_S to μ;

- Definition (Blocking) A coalition of agents S **blocks** an allocation μ, if there is an assignment μ_S of the houses owned by the coalition to the members of the coalition S, such that: i) some member of S prefers μ_S to μ; ii) no member of S prefers μ to μ_S.

- Definition (Blocking) A coalition of agents S **blocks** an allocation μ, if there is an assignment μ_S of the houses owned by the coalition to the members of the coalition S, such that: i) some member of S prefers μ_S to μ; ii) no member of S prefers μ to μ_S.
  - A blocking coalition can find a mutually beneficial trade (i.e. an exchange of houses among members of the coalition that improves all members' welfare with respect to the allocation μ)

- Definition (Blocking) A coalition of agents S **blocks** an allocation μ, if there is an assignment μ_S of the houses owned by the coalition to the members of the coalition S, such that: i) some member of S prefers μ_S to μ; ii) no member of S prefers μ to μ_S.
  - A blocking coalition can find a mutually beneficial trade (i.e. an exchange of houses among members of the coalition that improves all members' welfare with respect to the allocation μ)
- Definition (Core) An allocation is in the core of the housing market if it is not blocked by any coalition.
  - At a core allocation benefits from trade are exhausted

- Definition (Blocking) A coalition of agents S **blocks** an allocation μ, if there is an assignment μ_S of the houses owned by the coalition to the members of the coalition S, such that: i) some member of S prefers μ_S to μ; ii) no member of S prefers μ to μ_S.
  - A blocking coalition can find a mutually beneficial trade (i.e. an exchange of houses among members of the coalition that improves all members' welfare with respect to the allocation μ)
- Definition (Core) An allocation is in the core of the housing market if it is not blocked by any coalition.
  - At a core allocation benefits from trade are exhausted
  - In a marriage market, core matchings and stable matchings coincide

Gale's Top trading cycle algorithm

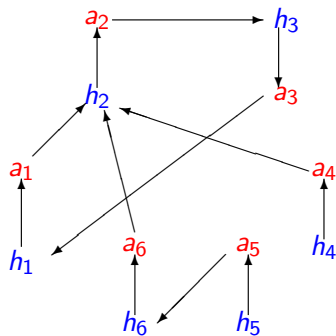Gale's Top trading cycle algorithm

- each agent points to
  his/her preferred house

Gale's Top trading cycle algorithm

- each agent points to his/her preferred house
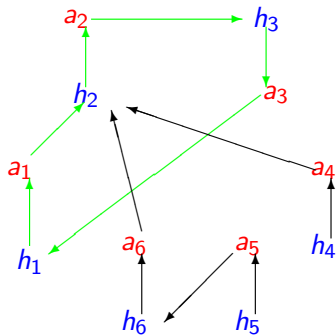- each house points to its owner

Gale's Top trading cycle algorithm

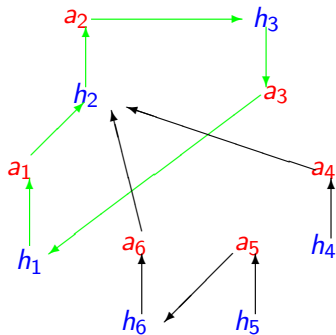- each agent points to his/her preferred house
- each house points to its owner

- there is at least one cycle
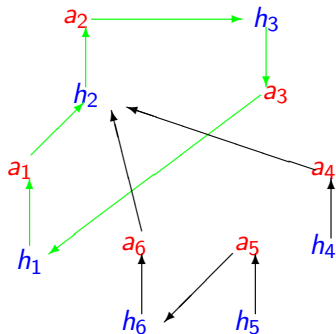
- there is at least one cycle

- there is at least one cycle

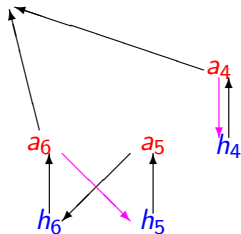- remove all cycles assigning houses to agents

- there is at least one cycle

- remove all cycles assigning houses to agents
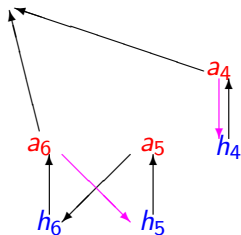 - agents within a cycle exchange houses among each others

- each remaining agent points to his/her preferred remaining house

- each remaining agent points to his/her preferred remaining house
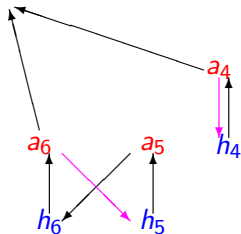
- each remaining agent
  points to his/her preferred
  remaining house

- remove all cycles assigning houses to agents

- each remaining agent points to his/her preferred remaining house

- remove all cycles assigning houses to agents
- continue until no agent/house is left

Theorem The outcome of the TTC mechanism is the unique core allocation of the housing market.

Theorem The outcome of the TTC mechanism is the unique core allocation of the housing market.

- The outcome of the TTC mechanism cannot be blocked

Theorem The outcome of the TTC mechanism is the unique core allocation of the housing market.

- The outcome of the TTC mechanism cannot be blocked
  - cannot make any agent matched in the first round better off

Theorem The outcome of the TTC mechanism is the unique core allocation of the housing market.

- The outcome of the TTC mechanism cannot be blocked
  - cannot make any agent matched in the first round better off (they are getting their favourite house)

Theorem The outcome of the TTC mechanism is the unique core allocation of the housing market.

- The outcome of the TTC mechanism cannot be blocked
  - cannot make any agent matched in the first round better off (they are getting their favourite house)
  - cannot make any agent matched in the second round better off without making some of the agents matched in the first round worse off

Theorem The outcome of the TTC mechanism is the unique core allocation of the housing market.

- The outcome of the TTC mechanism cannot be blocked
  - cannot make any agent matched in the first round better off (they are getting their favourite house)
  - cannot make any agent matched in the second round better off without making some of the agents matched in the first round worse off
  - cannot make any agent matched in round n better off without making some agents matched in earlier rounds worse off.

Theorem The TTC algorithm is a strategy proof mechanism.

Theorem The TTC algorithm is a strategy proof mechanism.

- an agent matched in round n cannot, by manipulating his/her preferences, break any of the cycles that form before round n

Theorem The TTC algorithm is a strategy proof mechanism.

- an agent matched in round n cannot, by manipulating his/her preferences, break any of the cycles that form before round n
  - preference manipulation cannot give the agent a house that was assigned earlier than round n.

Theorem The TTC algorithm is a strategy proof mechanism.

- an agent matched in round n cannot, by manipulating his/her preferences, break any of the cycles that form before round n
  - preference manipulation cannot give the agent a house that was assigned earlier than round n.

- getting an house that was assigned in a round later than n does not make the agent better off.

- An House allocation problem consists of

- An House allocation problem consists of
  - A collection of N agents, A;

- An House allocation problem consists of
  - A collection of N agents, A;
  - A collection of N houses, H;

- An House allocation problem consists of
  - A collection of N agents, A;
  - A collection of N houses, H;
  - For each agent (strict) a preference ordering over houses.
- House allocation vs. housing market

- An House allocation problem consists of
    - A collection of N agents, A;
    - A collection of N houses, H;
    - For each agent (strict) a preference ordering over houses.
- House allocation vs. housing market
    - Same preference structure (i.e. only one side has preferences)

- An House allocation problem consists of
    - A collection of N agents, A;
    - A collection of N houses, H;
    - For each agent (strict) a preference ordering over houses.
- House allocation vs. housing market
    - Same preference structure (i.e. only one side has preferences)
    - Same set of possible outcomes (i.e. assignment of a house to each agent)

- An House allocation problem consists of
  - A collection of N agents, A;
  - A collection of N houses, H;
  - For each agent (strict) a preference ordering over houses.
- House allocation vs. housing market
  - Same preference structure (i.e. only one side has preferences)
  - Same set of possible outcomes (i.e. assignment of a house to each agent)
  - Difference: agents do not "own" houses

House allocation

- An House allocation problem consists of
  - A collection of N agents, A;
  - A collection of N houses, H;
  - For each agent (strict) a preference ordering over houses.
- House allocation vs. housing market
  - Same preference structure (i.e. only one side has preferences)
  - Same set of possible outcomes (i.e. assignment of a house to each agent)
  - Difference: agents do not "own" houses (i.e. there is no initial allocation of houses to agents.)

House allocation

- An House allocation problem consists of
  - A collection of N agents, A;
  - A collection of N houses, H;
  - For each agent (strict) a preference ordering over houses.
- House allocation vs. housing market
  - Same preference structure (i.e. only one side has preferences)
  - Same set of possible outcomes (i.e. assignment of a house to each agent)
  - Difference: agents do not "own" houses (i.e. there is no initial allocation of houses to agents.)
- Goal: find an efficient assignment

- An House allocation problem consists of
  - A collection of N agents, A;
  - A collection of N houses, H;
  - For each agent (strict) a preference ordering over houses.
- House allocation vs. housing market
  - Same preference structure (i.e. only one side has preferences)
  - Same set of possible outcomes (i.e. assignment of a house to each agent)
  - Difference: agents do not "own" houses (i.e. there is no initial allocation of houses to agents.)
- Goal: find an efficient assignment (i.e. one that cannot be improved for all agents)

Serial dictatorship

Serial dictatorship

- Agents are given a priority ordering from 1 to N;

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

House allocation - mechanisms

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

Random serial dictatorship

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

Random serial dictatorship

- Same rules as in serial dictatorship;

House allocation - mechanisms

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

Random serial dictatorship

- Same rules as in serial dictatorship;
- Priority ordering is chosen randomly from the set of all priority orderings of agents.

House allocation - mechanisms

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

Random serial dictatorship

- Same rules as in serial dictatorship;
- Priority ordering is chosen randomly from the set of all priority orderings of agents.

CORE from random assignment

House allocation - mechanisms

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

Random serial dictatorship

- Same rules as in serial dictatorship;
- Priority ordering is chosen randomly from the set of all priority orderings of agents.

CORE from random assignment

- Randomly draw an initial assignment.

House allocation - mechanisms

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

Random serial dictatorship

- Same rules as in serial dictatorship;
- Priority ordering is chosen randomly from the set of all priority orderings of agents.

CORE from random assignment

- Randomly draw an initial assignment.
- Treat the initial assignment as an endowment.

House allocation - mechanisms

Serial dictatorship

- Agents are given a priority ordering from 1 to N;
- Agents choose houses in order of their priority.

Random serial dictatorship

- Same rules as in serial dictatorship;
- Priority ordering is chosen randomly from the set of all priority orderings of agents.

CORE from random assignment

- Randomly draw an initial assignment.
- Treat the initial assignment as an endowment.
- Use the TTC algorithm to find the unique core allocation given the initial assignment.

- Serial dictatorship, random serial dictatorship, and core from random endowment are

- Serial dictatorship, random serial dictatorship, and core from random endowment are
  - Pareto efficient;

- Serial dictatorship, random serial dictatorship, and core from random endowment are
  - Pareto efficient; and
  - Strategy proof.

- Housing market: each agent owns a house;

- Housing market: each agent owns a house;
- Housing allocation: no agents owns a house;

- Housing market: each agent owns a house;
- Housing allocation: no agents owns a house;
- Housing allocation with existing tenants: some agents "own" houses others do not

House allocation with existing tenants

- Housing market: each agent owns a house;
- Housing allocation: no agents owns a house;
- Housing allocation with existing tenants: some agents "own" houses others do not
  - allocating student housing - with upper year students having the right to keep their current residence (i.e. "own" their house)

- Natural variation of the random serial dictatorship mechanism

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;
  - Students already in residence can:

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;
  - Students already in residence can:
    - keep their residence;

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;
  - Students already in residence can:
    - keep their residence; or
    - give up their residence and participate in the lottery.

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
    - Students without residence participate in a lottery determining the priority ordering;
    - Students already in residence can:
        - keep their residence; or
        - give up their residence and participate in the lottery.
    - Stock of available houses to choose from is formed by

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;
  - Students already in residence can:
    - keep their residence; or
    - give up their residence and participate in the lottery.
  - Stock of available houses to choose from is formed by
    - empty houses;

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
    - Students without residence participate in a lottery determining the priority ordering;
    - Students already in residence can:
        - keep their residence; or
        - give up their residence and participate in the lottery.
    - Stock of available houses to choose from is formed by
        - empty houses; and
        - houses of students who choose to participate in the lottery.

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;
  - Students already in residence can:
    - keep their residence; or
    - give up their residence and participate in the lottery.
  - Stock of available houses to choose from is formed by
    - empty houses; and
    - houses of students who choose to participate in the lottery.
- Many US colleges use exactly this mechanism for assigning student residences

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;
  - Students already in residence can:
    - keep their residence; or
    - give up their residence and participate in the lottery.
  - Stock of available houses to choose from is formed by
    - empty houses; and
    - houses of students who choose to participate in the lottery.
- Many US colleges use exactly this mechanism for assigning student residences
- By participating in the lottery, a student in residence might end up with a worse house

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
  - Students without residence participate in a lottery determining the priority ordering;
  - Students already in residence can:
    - keep their residence; or
    - give up their residence and participate in the lottery.
  - Stock of available houses to choose from is formed by
    - empty houses; and
    - houses of students who choose to participate in the lottery.
- Many US colleges use exactly this mechanism for assigning student residences
- By participating in the lottery, a student in residence might end up with a worse house
  - this might induce some students in residence **not to participate** in the lottery

House allocation with existing tenants

- Natural variation of the random serial dictatorship mechanism
    - Students without residence participate in a lottery determining the priority ordering;
    - Students already in residence can:
        - keep their residence; or
        - give up their residence and participate in the lottery.
    - Stock of available houses to choose from is formed by
        - empty houses; and
        - houses of students who choose to participate in the lottery.
- Many US colleges use exactly this mechanism for assigning student residences
- By participating in the lottery, a student in residence might end up with a worse house
    - this might induce some students in residence **not to participate** in the lottery
    - the outcome can be inefficient

Example: Existing tenants a_1, a_2, a_3

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3;

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6;
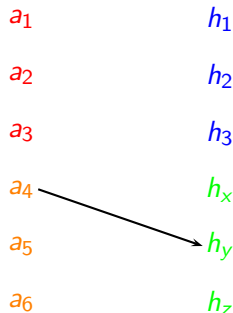
Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z;

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
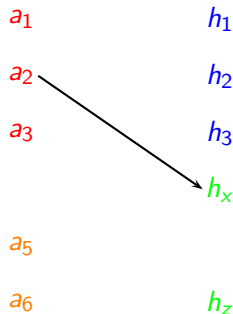
Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$

- assign agents their top choice in priority order

a_1 h_1

a_2 h_2

a_3 h_3

a_4 h_x

a_5 h_y

a_6 h_z

YRMH-IGYT mechanism

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
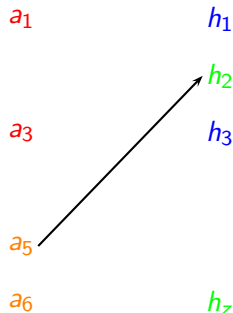
- assign agents their top choice in priority order

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
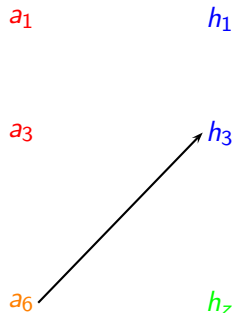
- assign agents their top choice in priority order

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
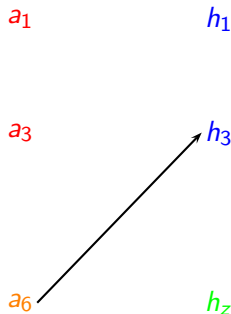
- assign agents their top choice in priority order
- until an agent requests an occupied house

a_1 h_1

a_3 h_3

a_6 h_z

YRMH-IGYT mechanism

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
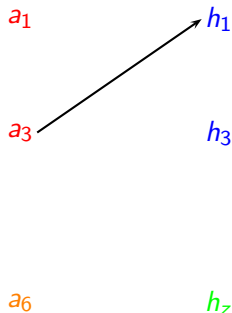
- assign agents their top choice in priority order
- until an agent requests an occupied house
- change the priority ordering placing the existing tenant ahead of requestor $\{4,2,5,\mathbf{3},\mathbf{6},1\}$

a_1 $\qquad\qquad$ h_1

a_3 $\qquad\qquad$ h_3

a_6 $\qquad\qquad$ h_z

YRMH-IGYT mechanism

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
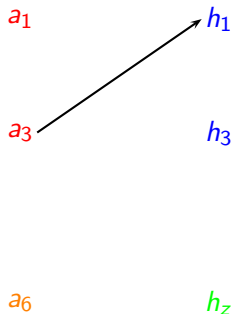
- assign agents their top choice in priority order
- until an agent requests an occupied house
- change the priority ordering placing the existing tenant ahead of requestor $\{4,2,5,\mathbf{3,6},1\}$

a_1 h_1

a_3 h_3

a_6 h_z

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
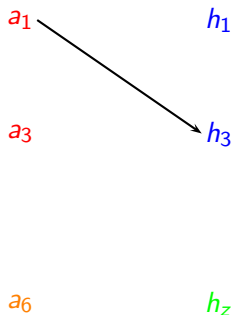
- assign agents their top choice in priority order
- until an agent requests an occupied house
- change the priority ordering placing the existing tenant ahead of requestor $\{4,2,5,\mathbf{3},\mathbf{6},1\}$
- ...repeat each time this happens $\{4,2,5,\mathbf{1},\mathbf{3},\mathbf{6}\}$

a_1 h_1

a_3 h_3

a_6 h_z

YRMH-IGYT mechanism

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
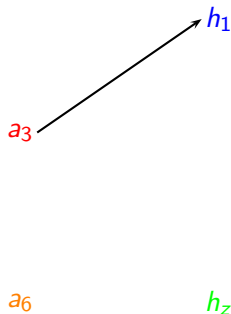
- assign agents their top choice in priority order
- until an agent requests an occupied house
- change the priority ordering placing the existing tenant ahead of requestor $\{4,2,5,\mathbf{3},\mathbf{6},1\}$
- ...repeat each time this happens $\{4,2,5,\mathbf{1},\mathbf{3},\mathbf{6}\}$

a_1 ⟶ h_1

a_3 h_3

a_6 h_z

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$
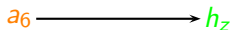
- assign agents their top choice in priority order
- until an agent requests an occupied house
- change the priority ordering placing the existing tenant ahead of requestor $\{4,2,5,\mathbf{3},\mathbf{6},1\}$
- ...repeat each time this happens $\{4,2,5,\mathbf{1},\mathbf{3},\mathbf{6}\}$

h_1

a_3

a_6 h_z

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$

- assign agents their top choice in priority order
- until an agent requests an occupied house
- change the priority ordering placing the existing tenant ahead of requestor $\{4,2,5,\mathbf{3},\mathbf{6},1\}$
- ...repeat each time this happens $\{4,2,5,\mathbf{1},\mathbf{3},\mathbf{6}\}$

$a_6 \longrightarrow h_z$

YRMH-IGYT mechanism

Example: Existing tenants a_1, a_2, a_3 with houses h_1, h_2, h_3; newcomers agents a_4, a_5, a_6; empty houses h_x, h_y, h_z; randomly chosen priority ordering of agents $\{4,2,5,6,3,1\}$

- assign agents their top choice in priority order
- until an agent requests an occupied house
- change the priority ordering placing the existing tenant ahead of requestor $\{4,2,5,\mathbf{3},\mathbf{6},1\}$
- ...repeat each time this happens $\{4,2,5,\mathbf{1},\mathbf{3},\mathbf{6}\}$
  - YRMH-IGYT stands for: "You request my house - I get your turn"

Theorem The YRMH-IGYT mechanism is Pareto efficient, strategy proof, and makes no existing tenant worse off.

Theorem The YRMH-IGYT mechanism is Pareto efficient, strategy proof, and makes no existing tenant worse off.

- Solve the participation problem

Theorem The YRMH-IGYT mechanism is Pareto efficient, strategy proof, and makes no existing tenant worse off.

- Solve the participation problem
- Relation to other mechanisms

Theorem The YRMH-IGYT mechanism is Pareto efficient, strategy proof, and makes no existing tenant worse off.

- Solve the participation problem
- Relation to other mechanisms
  - Coincide with Serial dictatorship when no agent has a house

Theorem The YRMH-IGYT mechanism is Pareto efficient, strategy proof, and makes no existing tenant worse off.

- Solve the participation problem
- Relation to other mechanisms
  - Coincide with Serial dictatorship when no agent has a house
  - Coincide with TTC when all agents are "tenants"

Theorem The YRMH-IGYT mechanism is Pareto efficient, strategy proof, and makes no existing tenant worse off.

- Solve the participation problem
- Relation to other mechanisms
    - Coincide with Serial dictatorship when no agent has a house
    - Coincide with TTC when all agents are "tenants"
    - IRMH-IGYT is a version of TTC with all unoccupied houses pointing to the agent with the highest priority (among those remaining in the market)

- Shortage of kidneys available for transplantation

- Shortage of kidneys available for transplantation
  - Yet, the "potential" supply (i.e. the number of not strictly needed functioning kidneys) vastly exceeds the demand

- Shortage of kidneys available for transplantation
    - Yet, the "potential" supply (i.e. the number of not strictly needed functioning kidneys) vastly exceeds the demand
- How do we reduce the shortage?

- Shortage of kidneys available for transplantation
  - Yet, the "potential" supply (i.e. the number of not strictly needed functioning kidneys) vastly exceeds the demand
- How do we reduce the shortage?
  - Increase the availability of cadaveric kidneys

- Shortage of kidneys available for transplantation
  - Yet, the "potential" supply (i.e. the number of not strictly needed functioning kidneys) vastly exceeds the demand
- How do we reduce the shortage?
  - Increase the availability of cadaveric kidneys
  - Increase the supply of live donor kidneys

- Shortage of kidneys available for transplantation
  - Yet, the "potential" supply (i.e. the number of not strictly needed functioning kidneys) vastly exceeds the demand
- How do we reduce the shortage?
  - Increase the availability of cadaveric kidneys
  - Increase the supply of live donor kidneys
    - cannot use any "price" mechanism because of legal (and moral) constraints

- Live kidney donor must pass two compatibility tests before a transplantation is carried out

- Live kidney donor must pass two compatibility tests before a transplantation is carried out
  - Blood type match

Kidney donation

- Live kidney donor must pass two compatibility tests before a transplantation is carried out
  - Blood type match
    - O-type patient can only accept O-type donor
    - A-type patient can accept A and O-type donor
    - B-type patient can accept B and O-type donor
    - AB-type patient can accept all donors

Kidney donation

- Live kidney donor must pass two compatibility tests before a transplantation is carried out
  - Blood type match
    - O-type patient can only accept O-type donor
    - A-type patient can accept A and O-type donor
    - B-type patient can accept B and O-type donor
    - AB-type patient can accept all donors
  - Tissue (HLA) compatibility

Kidney donation

- Live kidney donor must pass two compatibility tests before a transplantation is carried out
  - Blood type match
    - O-type patient can only accept O-type donor
    - A-type patient can accept A and O-type donor
    - B-type patient can accept B and O-type donor
    - AB-type patient can accept all donors
  - Tissue (HLA) compatibility
- Potential inefficiency: when a willing donor does not meet the compatibility tests the kidney is "wasted"

Kidney donation

- Live kidney donor must pass two compatibility tests before a transplantation is carried out
  - Blood type match
    - O-type patient can only accept O-type donor
    - A-type patient can accept A and O-type donor
    - B-type patient can accept B and O-type donor
    - AB-type patient can accept all donors
  - Tissue (HLA) compatibility
- Potential inefficiency: when a willing donor does not meet the compatibility tests the kidney is "wasted"
  - most donors are close friends/relatives, unwilling to donate to a stranger

Kidney donation

- Live kidney donor must pass two compatibility tests before a transplantation is carried out
  - Blood type match
    - O-type patient can only accept O-type donor
    - A-type patient can accept A and O-type donor
    - B-type patient can accept B and O-type donor
    - AB-type patient can accept all donors
  - Tissue (HLA) compatibility
- Potential inefficiency: when a willing donor does not meet the compatibility tests the kidney is "wasted"
  - most donors are close friends/relatives, unwilling to donate to a stranger
- Possible improvements: donor could be willing to donate to stranger if that improves the chances of their close friend/relative receiving a kidney **i.e a kidney exchange**

- Two types of kidney exchanges

- Two types of kidney exchanges
  - Pairwise kidney exchange:

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

- Two types of kidney exchanges
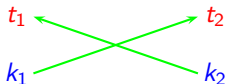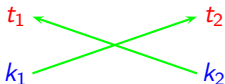 - Pairwise kidney exchange: exchange kidney with another patient-donor pair

$$t_1 \qquad\qquad t_2$$

$$k_1 \qquad\qquad k_2$$

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

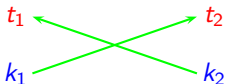$$t_1 \qquad\qquad\longrightarrow\quad t_2$$
$$k_1 \qquad\qquad\qquad\quad k_2$$

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

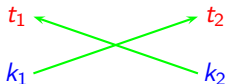$$t_1 \longleftarrow \quad \longrightarrow t_2$$
$$k_1 \longrightarrow \quad \longleftarrow k_2$$

  - Exchange to list: donate kidney to patient at top of waiting list in exchange of top spot on waiting list

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

$$t_1 \quad\longleftarrow\quad t_2$$
$$k_1 \quad\longrightarrow\quad k_2$$

  - Exchange to list: donate kidney to patient at top of waiting list in exchange of top spot on waiting list

$$t_1 \qquad\qquad w_1$$
$$w_2$$
$$k_1 \qquad\qquad w_3$$
$$w_4$$
$$\vdots$$

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

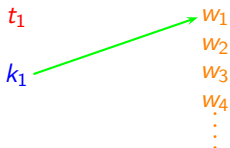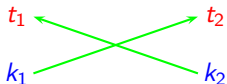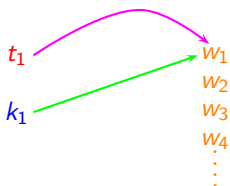 - Exchange to list: donate kidney to patient at top of waiting list in exchange of top spot on waiting list

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

$$t_1 \quad\quad\quad t_2$$
$$k_1 \quad\quad\quad k_2$$

  - Exchange to list: donate kidney to patient at top of waiting list in exchange of top spot on waiting list

$$t_1 \quad\quad\quad w_1$$
$$\quad\quad\quad\quad w_2$$
$$k_1 \quad\quad\quad w_3$$
$$\quad\quad\quad\quad w_4$$
$$\quad\quad\quad\quad \vdots$$

- Two types of kidney exchanges
  - Pairwise kidney exchange: exchange kidney with another patient-donor pair

$$t_1 \quad\quad\quad\quad t_2$$
$$k_1 \quad\quad\quad\quad k_2$$

  - Exchange to list: donate kidney to patient at top of waiting list in exchange of top spot on waiting list

$$
\begin{aligned}
& t_1 \quad\quad\quad\quad w_1 \\
& \quad\quad\quad\quad\quad w_2 \\
& k_1 \quad\quad\quad\quad w_3 \\
& \quad\quad\quad\quad\quad w_4 \\
& \quad\quad\quad\quad\quad \vdots
\end{aligned}
$$
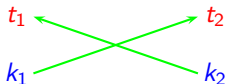
- Looks similar to YRMH-IGYT

- A Kidney exchange problem consists of:

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
  - For each patient, t_i, a set of compatible kidneys
    $K_i \subseteq K = \{k_1, \ldots, k_n\}$

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
  - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
  - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i

Kidney exchange problem

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
  - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
  - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i **and** the option of exchanging own kidney, k_i for priority w on the waiting list

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
  - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
  - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i **and** the option of exchanging own kidney, k_i for priority w on the waiting list
- Question: How do we organize a kidney exchange program such that
  - The outcome is Pareto efficient,

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
  - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
  - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i **and** the option of exchanging own kidney, k_i for priority w on the waiting list
- Question: How do we organize a kidney exchange program such that
  - The outcome is Pareto efficient, it is not possible to improve further the welfare of **all**

Kidney exchange problem

- A Kidney exchange problem consists of:
    - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
    - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
    - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i **and** the option of exchanging own kidney, k_i for priority w on the waiting list

- Question: How do we organize a kidney exchange program such that
    - The outcome is Pareto efficient, it is not possible to improve further the welfare of **all**
    - For each patient, the outcome is never worse than not participating in the mechanism,

- A Kidney exchange problem consists of:
    - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
    - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
    - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i **and** the option of exchanging own kidney, k_i for priority w on the waiting list
- Question: How do we organize a kidney exchange program such that
    - The outcome is Pareto efficient, it is not possible to improve further the welfare of **all**
    - For each patient, the outcome is never worse than not participating in the mechanism, ensures broad participation, no donor kidney is un-necessarily "wasted"

Kidney exchange problem

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
  - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
  - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i **and** the option of exchanging own kidney, k_i for priority w on the waiting list
- Question: How do we organize a kidney exchange program such that
  - The outcome is Pareto efficient, it is not possible to improve further the welfare of **all**
  - For each patient, the outcome is never worse than not participating in the mechanism, ensures broad participation, no donor kidney is un-necessarily "wasted"
  - The mechanism is strategy proof,

Kidney exchange problem

- A Kidney exchange problem consists of:
  - A set of donor-patient pairs $\{(t_1, k_1), \ldots, (t_n, k_n)\}$
  - For each patient, t_i, a set of compatible kidneys $K_i \subseteq K = \{k_1, \ldots, k_n\}$
  - For each patient, t_i, a (strict) preference ordering over the set of compatible kidneys K_i **and** the option of exchanging own kidney, k_i for priority w on the waiting list
- Question: How do we organize a kidney exchange program such that
  - The outcome is Pareto efficient, it is not possible to improve further the welfare of **all**
  - For each patient, the outcome is never worse than not participating in the mechanism, ensures broad participation, no donor kidney is un-necessarily "wasted"
  - The mechanism is strategy proof, patients have incetive to disclose their preferences honestly