Matching with Property Rights

Jinyong Jeong

Boston College

Tenth Conference on Economic Design, York

June 16, 2017

Motivating example

- Assign parking spaces to drivers via centralized system.
- Residents have own spaces, over which they have exclusive rights.

Motivating example

- Assign parking spaces to drivers via centralized system.
- Residents have own spaces, over which they have exclusive rights.

- Usage of resident's space may depend on what this resident gets from the system.
- For example, if a resident gets a 30-minutes parking, her space should be assigned to a (less than) 30-minutes parking too.

Introduction

- Temporal resource allocation problem with property rights

Introduction

- Temporal resource allocation problem with property rights

- temporal: goods are not assigned permanently.
- property right: some agents have their own goods and will reoccupy them after the matching period ends.

Model

$I = \{i_1, \cdots, i_n\}$: a set of individuals with unit demand,

$S = \{s_1, \cdots, s_m\}$: a set of spaces with unit capacity,

$T = \{t^+, t^-\}$: a set of contractual terms,

$\mathbb{X} = I \times S \times T$: a set of contracts,

$\succ_I = (\succ_{i_1}, \cdots, \succ_{i_n})$: a list of individuals' strict preferences over contracts,

$\succ_S = (\succ_{s_1}, \cdots, \succ_{s_m})$: a list of base priority over contracts at each space.

Examples

- Parking space assignment with residents :
  - t^+ is a long-term parking, and t^- is a short-term parking.
- Student exchange program :
  - t^+ includes housing support, and t^- does not.
- Sabbatical housing:
  - t^+ is a two semester contract, and t^- is one semester.

Model

- Matching is a set of contracts where each agent appears in at most one contract.
- $\mu(a)$ = contract that an agent a is matched to in matching μ.

Stability

Definition

A matching μ is stable if,

i) for all i, $\mu(i) \succ_i \emptyset$,
ii) there does not exist an individual-space pair (i, s), where $s \succ_i \mu(i)$ and $i \succ_s \mu(s)$.

Matching with Property Rights

- there are two kinds of individuals;
  - residents, $r \in I_R$,
  - visitors, $v \in I_V$,
- and two kinds of spaces;
  - resident space $s^r \in S^R$,
  - vacant spaces $s^v \in S^V$.

Matching with property rights

- Resident space s^r is owned by resident r,
- r's property right works as follows;
  - If r is assigned a t^+ contract, her space s^r can be assigned either t^+ or t^- contract.
  - If r is assigned a t^- contract, s^r cannot be assigned a t^+ contract.

Examples revisited

- Parking space assignment with residents
    - If a resident is assigned a short-term contract (t^-), her space cannot be matched to a long-term contract (t^+).

Examples revisited

- Parking space assignment with residents
  - If a resident is assigned a short-term contract (t^-), her space cannot be matched to a long-term contract (t^+).
- Student exchange program
  - If a student is assigned a contract without housing support, (t^-), then her seat can only be matched to one without housing support.

Examples revisited

- Parking space assignment with residents
  - If a resident is assigned a short-term contract (t^-), her space cannot be matched to a long-term contract (t^+).
- Student exchange program
  - If a student is assigned a contract without housing support, (t^-), then her seat can only be matched to one without housing support.
- Sabbatical housing
  - If a professor wants a one semester housing (t^-), his home will be only assigned a one semester contract.

Respecting property rights

Let $t(\mu(a))$ denote the term of the contract that an agent a is matched to in matching μ.

Definition

A matching μ *respects property rights* if, for any resident r and her space s^r,

$$t(\mu(r)) = t^- \implies t(\mu(s^r)) \neq t^+.$$

Respecting property rights

Let $t(\mu(a))$ denote the term of the contract that an agent a is matched to in matching μ.

Definition

A matching μ *respects property rights* if, for any resident r and her space s^r,

$$t(\mu(r)) = t^- \implies t(\mu(s^r)) \neq t^+.$$

When a resident r is matched to a t^- contract, her space s^r cannot be matched to a t^+ contract.

Incompatibility

Theorem

There is no mechanism that is both stable and respects property rights.

Example

Two individuals, a resident r and a visitor v. The resident has space s^r, and there is a vacant space s^v. Let $x = \{v, s^r, t^+\}$ and $y = \{r, s^v, t^-\}$. Both are acceptable in each of its involved space with highest priorities. The preferences of individuals are;

$$r : \{y\} \succ_r \emptyset$$
$$v : \{x\} \succ_v \emptyset$$

$\{x, y\}$ is the only stable matching in this economy. However, this does not respect r's property right because s^r is assigned a t^+ contract when r is assigned a t^- contract.

Overview

- Find a matching that respects property rights, with some desirable properties.

Overview

- Find a matching that respects property rights, with some desirable properties.

- TTC-type approach.
  - YRMH-IGYT (Abdulkadiroğlu and Sönmez (1999))
- DA-type approach.
  - Cumulative Offer (Hatfield and Milgrom 2005)

Overview

- Introduce a claim contract,
- Construct choice functions,
- Mechanism which always respects property rights.

Overview

	Benchmark	Cumulative	Repeated
property right	✓	✓	✓
stability $\succ$	X	X	X
stability Ch_a	✓	◯	◯
strategy proof	✓	✓	X
non-wastefulness	X	△	△ △

✓: condition satisfied X : condition violated

◯: almost satisfied △: violated but improved

Choice function design

- An agent a's choice function is a systematic procedure that selects a set of contracts from a set X.
- Preference profile $\succ$ can be converted to a choice function, for example, by letting each choice function of the agent select the highest ranked contract under his/her preference, i.e.,

$$Ch_a(X) = \max_{\succ_a}[\{x \in X_a\} \cup \emptyset]$$

.

Claim Contract

- To deal with property rights with the choice function, I introduce a claim contract.

Claim Contract

- To deal with property rights with the choice function, I introduce a claim contract.
- A claim contract $c_r = (r, s^r, t^-) \in \mathbb{X}$ indicates only t^- contract is acceptable.
  - Note that, this is not same as remaining unmatched.

Example

Let x be a contract with the term t^{+}, and y be a contract with the term t^{-}. A typical choice function of a resident space s^{r} is the following:

$$Ch_{s^r}(\{x, y\}) = \{x\}$$
$$Ch_{s^r}(\{x, y, c_r\}) = \{c_r, y\}$$

claim Contract

Example

Let x be a contract with the term t^+, and y be a contract with the term t^-. A typical choice function of a resident space s^r is the following:

$$Ch_{s^r}(\{x, y\}) = \{x\}$$
$$Ch_{s^r}(\{x, y, c_r\}) = \{c_r, y\}$$

- When c_r is available, t^+ contract cannot be chosen.
- Claim contract protects its owners property right.

claim Contract

Example

Let x be a contract with the term t^+, and y be a contract with the term t^-. A typical choice function of a resident space s^r is the following:

$$Ch_{s^r}(\{x, y\}) = \{x\}$$
$$Ch_{s^r}(\{x, y, c_r\}) = \{c_r, y\}$$

Note that,

- it violates substitutes condition (Hatfield and Kojima, 2010),
- it does not have path-independent modifications (Yenmez, 2017).

Choice function design

- For v and s^v, Ch_v and Ch_{s^v} chooses the top ranked contract in the preference list.
- For resident r, Ch_r chooses the claim contract if it's available, and chooses the top ranked contract.
  - $\rightarrow$ c_r is chosen for technical reason, and has no role in the choice of resident.

Choice function design

Definition

Given a set of contracts X and a base priority ordering, a space s^r's choice $Ch_{s^r}(X)$ is obtained as follows:

0 : Remove all the contracts for another space s' and add them to the rejected set $R_{s^r}(X)$ and proceed with phase 1. Contracts survived phase 0 involves only space s.

1 : If there is no claim contract, then choose the top priority contract and terminate the procedure. Otherwise, proceed with phase 2.

2 : When there is a claim contract $c_r = (r, s^r, t^-)$, choose the claim and the top priority contract among the contracts with the term t^-. If there is no contract with the term t^-, choose only the claim contract. Terminate the procedure.

Mechanism Design

- With the claim contract, design a mechanism that always produces a property-respecting allocation.
- The allocation will not be stable under preferences, but might be stable (or close to stable) with respect to the choice function.

Stability w.r.t. choice function

Let X_a be the set of contracts associated with agent $a \in I \cup S$.

Definition

A set of contracts X is stable (w.r.t. choice function Ch_a) if,

i) for all $a \in I \cup S$, $Ch_a(X) = X_a$,

ii) there does not exist a set of contracts Y such that $Y \cap X = \emptyset$, for every a, $Y_a \subseteq Ch_a(Y \cup X)$.

Mechanism 1: Benchmark

- One way to respect property rights is to assign resident space only t^- contracts.

Mechanism 1: Benchmark

- One way to respect property rights is to assign resident space only t^{-} contracts.
- This corresponds to put claim contracts in each resident space's choice function from the very beginning of the algorithm.

Cumulative offer

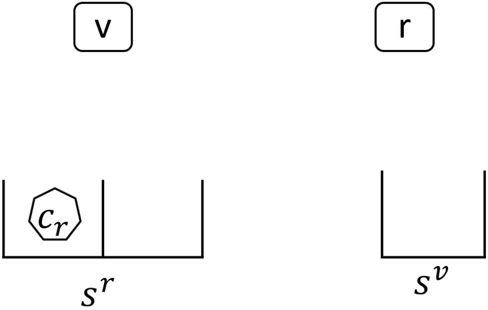
We will use the following cumulative offer algorithm to assign spaces to agents.

- Step 1: One (randomly chosen) agent offers her most preferred contract $x_1 = (i(1), s(1), t)$. The space that is offered the contract, $s(1)$, holds the contract if $x_1 \in C_{s(1)}(\{x_1\})$ and rejects it otherwise. Let $A_{s(1)}(1) = \{x_1\}$ and $A_s(1) = \emptyset$ for all $s \neq s(1)$.

In general

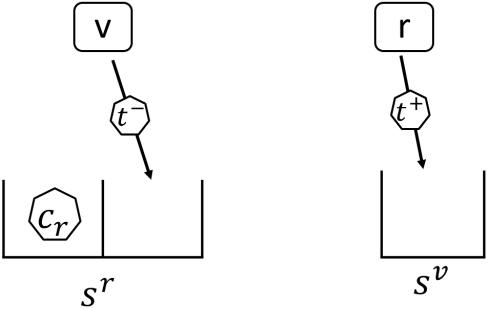
- Step k: One of the agents without contract held by any space offers her most preferred contract among the ones that are not previously rejected, $x_k = (i(k), s(k), t)$. Space $s(k)$ holds the contract if $x_k \in C_{s(k)}(A_{s(k)}(k-1) \cup \{x_k\})$ and rejects it otherwise. Let $A_{s(k)}(k) = A_{s(k)}(k-1) \cup \{x_k\}$ and $A_s(k) = A_s(k-1)$ for all $s \neq s(k)$.

Mechanism 1: Benchmark



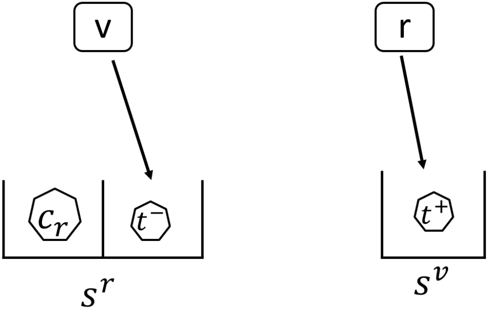
Each resident space has its own claim contract at the beginning of the algorithm.

Mechanism 1: Benchmark



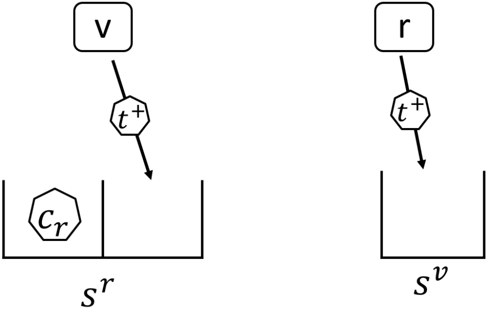
Suppose a visitor wants a t^- contract from the space s^r and the resident wants a t^+ contract from the space s^v.

Mechanism 1: Benchmark



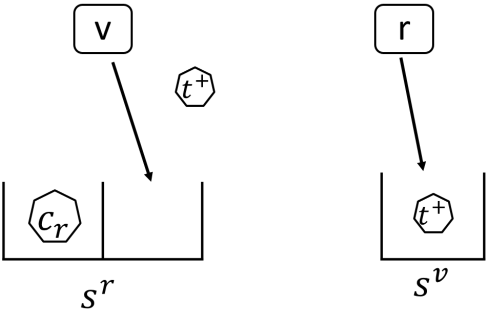
claim does not reject the t^- contract, v is assigned a t^- contract.

Mechanism 1: Benchmark



What happens if v wants a t^+ contract?

Mechanism 1: Benchmark



Because there is a claim in s^r's choice set, t^+ is rejected and v is remain unmatched.

Mechanism 1: Benchmark

Remark

Benchmark respects property rights, is stable w.r.t. the choice function, and is strategy proof.

Mechanism 1: Benchmark

Remark

Benchmark respects property rights, is stable w.r.t. the choice function, and is strategy proof.

Wasteful since some of the resident spaces may not be used even when the resident is assigned a t^+ contract.

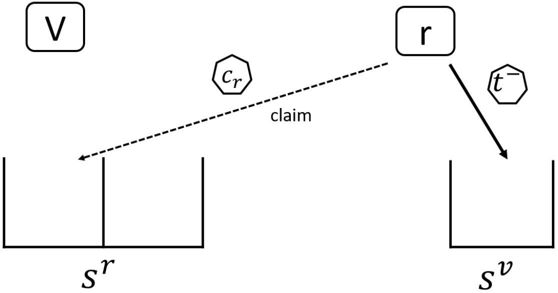
Mechanism 2: Cumulative offer with claim proposing

- Can we do better than this?

Mechanism 2: Cumulative offer with claim proposing

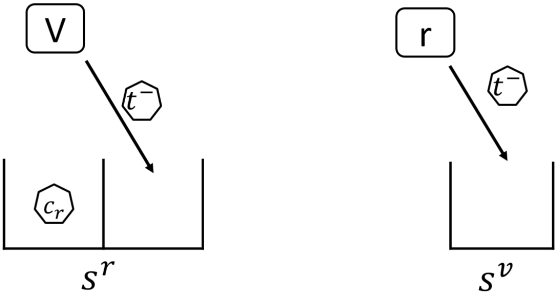
- Can we do better than this?
- Instead of having the claim from the beginning, let it come during the algorithm.
- When a resident is demanding a t^{-} contract, she sends a claim contract to her space as well.

Mechanism 2: Cumulative offer with claim proposing



When a resident r offers a t^- contract, she sends c_r to her space s^r, and the claim will remain there until the end of the algorithm.

Mechanism 2: Cumulative offer with claim proposing



With the claim in the choice set of s^r, only t^- contract is acceptable.

Mechanism 2: Cumulative offer with claim proposing

Definition

A set of contracts X is *practically stable* (w.r.t. Ch_a) if,

i) for all $a \in I \cup S$, $Ch_a(X) = X_a$,

ii) if there exists a set of blocking contracts Y such that $Y \cap X = \emptyset$,

$x \notin Y$ if $x \neq c_r$ for some r,

Mechanism 2: Cumulative offer with claim proposing

Definition

A set of contracts X is *practically stable* (w.r.t. Ch_a) if,

i) for all $a \in I \cup S$, $Ch_a(X) = X_a$,

ii) if there exists a set of blocking contracts Y such that $Y \cap X = \emptyset$,

$x \notin Y$ if $x \neq c_r$ for some r,

- Only possible blocking contracts are claim contracts, which cannot affect the resident's physical assignment, and which will not appear in the cumulative offer process.

Mechanism 2: Cumulative offer with claim proposing

Proposition

Cumulative offer with claim proposing respects property rights, is practically stable w.r.t. the choice function, is strategy proof, and is less wasteful than the Benchmark.

Mechanism 2: Cumulative offer with claim proposing

Proposition

Cumulative offer with claim proposing respects property rights, is practically stable w.r.t. the choice function, is strategy proof, and is less wasteful than the Benchmark.

- less wasteful because,
  some of the resident spaces can be assigned a t^+ contract.

Mechanism 3: Repeated claim proposing

- Can we do even better?

Mechanism 3: Repeated claim proposing

- Can we do even better?

- Postponing the claim contract to the end of the cumulative algorithm.
- Less wasteful, but not strategy proof.

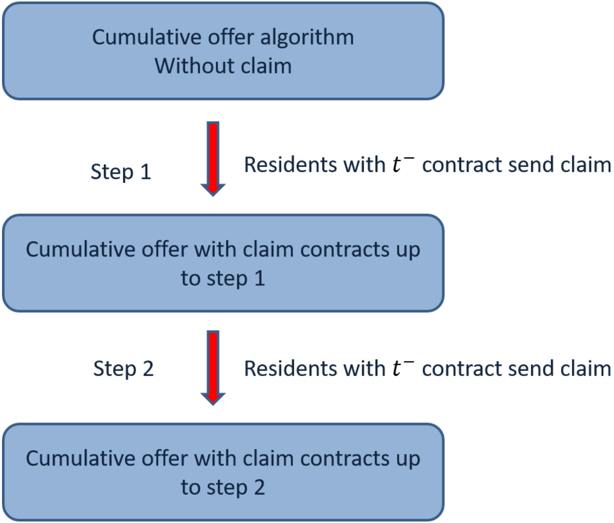
Mechanism 3: Repeated claim proposing

- Can we do even better?

- Postponing the claim contract to the end of the cumulative algorithm.
- Less wasteful, but not strategy proof.

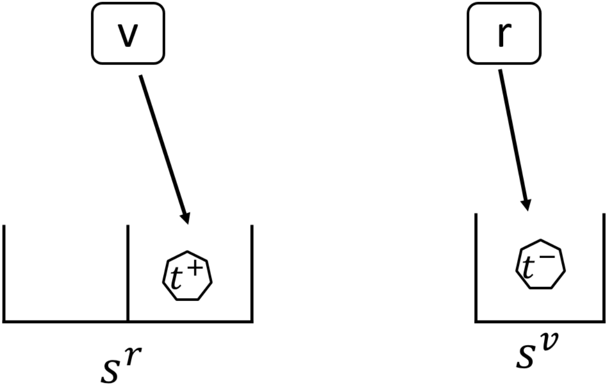
- Not yet known if the previous one is least wasteful if we were to keep strategy proofness.

Mechanism 3: Repeated claim proposing



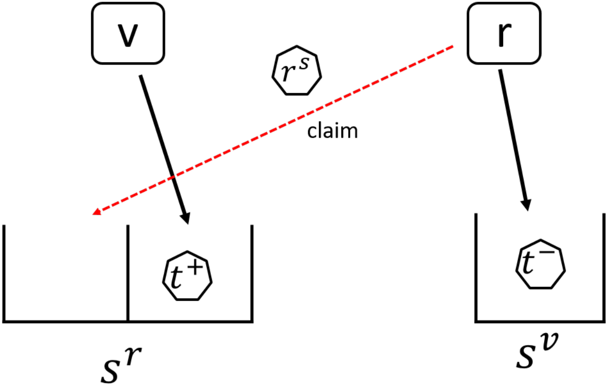
Mechanism 3: Repeated claim proposing

Example 1



Resident r holds a t^- contract after the first cumulative offer algorithm.

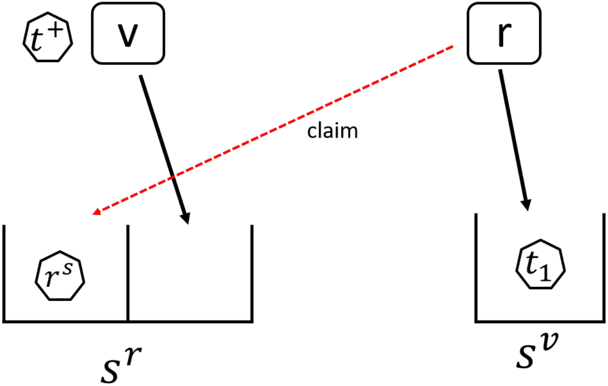
Example 1



r sends a claim to her space.

Mechanism 3: Repeated claim proposing

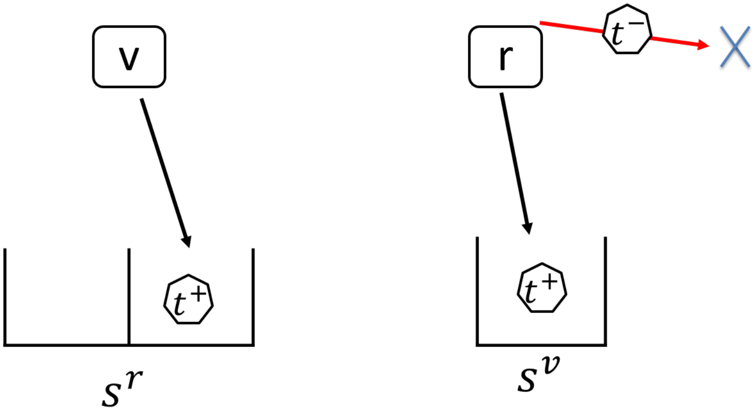
Example 1



As a result, t^+ is rejected from s^r.

Mechanism 3: Repeated claim proposing

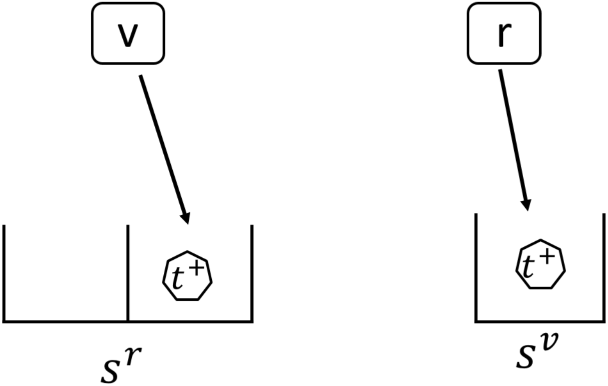
Example 2



If r is rejected a t^- contract in the previous phase, but gets a t^+ at the end of the algorithm, r does not send a claim contract.

Mechanism 3: Repeated claim proposing

Example 2



As a result, v keeps his t^+ contract.

Conclusion

- Introduces a new idea, the claim contract, to solve matching with property rights model.
- Proposes a algorithm that produces a property respecting assignment, while reducing wastefulness from the property rights.
- Can be extended to a many-to-one matching model, and/or incorporate general contractual terms.
- Real worlds application includes parking space allocation, student exchange problem.