Who Gets What and When

Dynamic Incentives in Repeated Matching Markets

Mitchell Watt | mwatt@stanford.edu | 34th Stony Brook International Conference on Game Theory | July 27, 2023

A Motivating Example

Ridesharing

- Jobs appear ~ randomly over time.
- Net of payments, Castro et al. (2021) show substantial heterogeneity in job value.
- Rides are typically allocated FCFS, little-to-no penalty for rejecting jobs.

A Motivating Example

Ridesharing

- Jobs appear ~ randomly over time.
- Net of payments, Castro et al. (2021) show substantial heterogeneity in job value.
- Rides are typically allocated FCFS, little-to-no penalty for rejecting jobs.

Drivers at the front of the queue become *selective* about job quality, leading to inefficiently long waiting times.

A Motivating Example
Ridesharing

- Jobs appear ~ randomly over time.
- Net of payments, Castro et al. (2021) show substantial heterogeneity in job value.
- Rides are typically allocated FCFS, little-to-no penalty for rejecting jobs.

Drivers at the front of the queue become *selective* about job quality, leading to inefficiently long waiting times.

Alternative: promise better *future* positions in the queue to drivers who accept *worse* jobs *today.*

This Paper

What is the optimal design of dynamic incentives in **matching markets** with **fixed transfers**?

I introduce a repeated matching model with a fixed population of agents and a period-by-period *ex post* participation constraint.

This Paper

What is the optimal design of dynamic incentives in **matching markets** with **fixed transfers**?

I introduce a repeated matching model with a fixed population of agents and a period-by-period *ex post* participation constraint.

Key assumption: values are common and observable.

This Paper

What is the optimal design of dynamic incentives in **matching markets** with **fixed transfers**?

I introduce a repeated matching model with a fixed population of agents and a period-by-period *ex post* participation constraint.

Key assumption: values are common and observable.

Key results:

- Principal incentivizes undesirable allocations using promises of improved future allocations.
- Principal is “loyal”: agents with worse historical allocations are prioritized for better current allocations.

Model

A fixed population of N agents and a single principal, who owns a stream of arriving items.

In each period, $t = 0,1,2,...$:

1. N items arrive with values v_t observed by principal and agents.
   Arrival process is common knowledge, values may be negative (“bads”).

2. Principal offers an item to agent i with probability x_{it}.

3. Agent i accepts with probability y_{it}, its outside option is normalized to 0.
   Unallocated/unaccepted items disappear.

Model

A fixed population of N agents and a single principal, who owns a stream of arriving items.

In each period, $t = 0,1,2,\ldots$:

1. N items arrive with values v_t observed by principal and agents.
   Arrival process is common knowledge, values may be negative ("bads").

2. Principal offers an item to agent i with probability x_{it}.

3. Agent i accepts with probability y_{it}, its outside option is normalized to 0.
   Unallocated/unaccepted items disappear.

Principal cares about total matches:

$$U^P = (1 - \delta_P) \sum_t \sum_i \delta_P^t x_{it} y_{it}$$

Model

A fixed population of N agents and a single principal, who owns a stream of arriving items.

In each period, $t = 0,1,2,...$:

1. N items arrive with values v_t observed by principal and agents.
   Arrival process is common knowledge, values may be negative ("bads").

2. Principal offers an item to agent i with probability x_{it}.

3. Agent i accepts with probability y_{it}, its outside option is normalized to 0.
   Unallocated/unaccepted items disappear.

Principal cares about total matches:

$$U^P = (1 - \delta_P) \sum_t \sum_i \delta_P^t x_{it} y_{it}$$

Agents care about quality of matches:

$$U^A = (1 - \delta_A) \sum_t \delta_A^T x_{it} y_{it} v_{it}$$

Recursive Formulation of Optimal Contract

Principal chooses history-dependent allocation rule to maximize payoffs subject to obedience constraint for agents $\Rightarrow$ reformulate recursively using *promised utility* as state vector

Recursive Formulation of Optimal Contract

Principal chooses history-dependent allocation rule to maximize payoffs subject to obedience constraint for agents $\Rightarrow$ reformulate recursively using *promised utility* as state vector

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),\, u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|\, + \delta_P \Phi\left(u'(v;u)\right)\right] \text{ subject to}$$

feasible match feasible promise match size value of promise

Recursive Formulation of Optimal Contract

Principal chooses history-dependent allocation rule to maximize payoffs subject to obedience constraint for agents ⇒ reformulate recursively using *promised utility* as state vector

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|+\delta_P\Phi\left(u'(v;u)\right)\right] \text{ subject to}$$

feasible match feasible promise match size value of promise

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\right] \geq u_i, \text{ for each } i,$$ *promise-keeping*

i's value today promised utility current promise

Recursive Formulation of Optimal Contract

Principal chooses history-dependent allocation rule to maximize payoffs subject to obedience constraint for agents ⇒ reformulate recursively using *promised utility* as state vector

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),\, u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|+\delta_P\Phi\left(u'(v;u)\right)\right]$$ subject to

feasible match feasible promise match size value of promise

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\right]\geq u_i,$$ for each i, *promise-keeping*

i's value today promised utility current promise

$$(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\geq 0,$$ for each i and v. *participation*

i's value today tomorrow's promise non-participation

Recursive Formulation of Optimal Contract

Principal chooses history-dependent allocation rule to maximize payoffs subject to obedience constraint for agents ⇒ reformulate recursively using *promised utility* as state vector

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),\, u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|X(v;u)| + \delta_P \Phi\left(u'(v;u)\right)\right] \text{ subject to}$$

feasible match feasible promise match size value of promise

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u) + \delta_A u_i'(v;u)\right] \geq u_i, \text{ for each } i, \quad \textit{promise-keeping}$$

i's value today promised utility current promise

$$(1-\delta_A)v_i^X(v;u) + \delta_A u_i'(v;u) \geq 0, \text{ for each } i \text{ and } v\,. \quad \textit{participation}$$

i's value today tomorrow's promise non-participation

Theorem: $\Phi(\cdot)$ exists, is unique, monotone decreasing, Schur-concave, continuous, semi-differentiable

Properties of the Optimal Contract

Properties of the Optimal Contract

1. The principal promises better future allocations to agents who accept bad items in current period.

Properties of the Optimal Contract

1. The principal promises better future allocations to agents who accept bad items in current period.

2. Endogenous ‘loyalty’: agents with the worst historical allocations (= highest promised utility) have priority for better items.

Properties of the Optimal Contract

1. The principal promises better future allocations to agents who accept bad items in current period.

2. Endogenous 'loyalty': agents with the worst historical allocations (= highest promised utility) have priority for better items.

3. Optimal policy is a cutoff rule: worst-off agent allocated best item iff its value is at least $\overline{v}_1(u)$, second-worst-off agent allocated best remaining item iff its value is at least $\overline{v}_2(u)$.

Properties of the Optimal Contract

1. The principal promises better future allocations to agents who accept bad items in current period.

2. Endogenous 'loyalty': agents with the worst historical allocations (= highest promised utility) have priority for better items.

3. Optimal policy is a cutoff rule: worst-off agent allocated best item iff its value is at least $\overline{v}_1(u)$, second-worst-off agent allocated best remaining item iff its value is at least $\overline{v}_2(u)$.

4. Long-run performance depends on ratio of δ^P to δ^A.

Illustrating the Optimal Contract

Two agents, $\delta^A = 0.8$ and a patient principle, $\delta^P = 0.9$.
Two items arrive each period i.i.d. value -2 or +1 with equal probability.

Illustrating the Optimal Contract

Two agents, $\delta^A = 0.8$ and a patient principle, $\delta^P = 0.9$.
Two items arrive each period i.i.d. value -2 or +1 with equal probability.

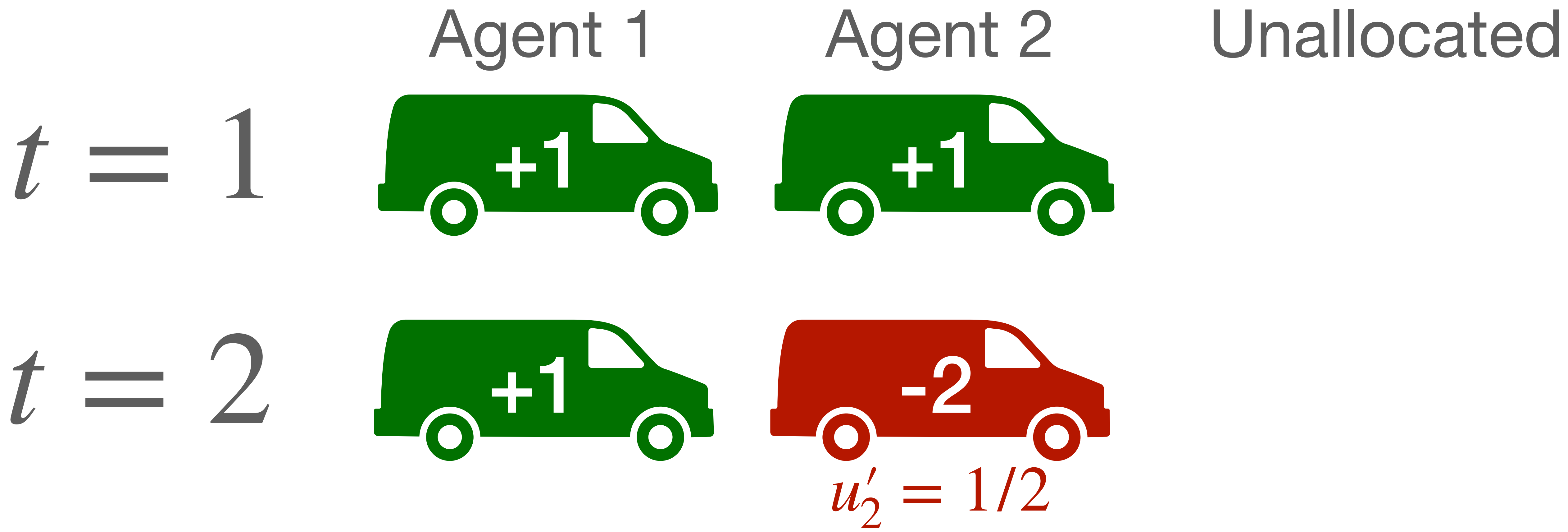
Agent 1 Agent 2 Unallocated

$t = 1$

Assign one to each agent: everyone is happy!

Illustrating the Optimal Contract

Two agents, $\delta^A = 0.8$ and a patient principle, $\delta^P = 0.9$.
Two items arrive each period i.i.d. value -2 or +1 with equal probability.

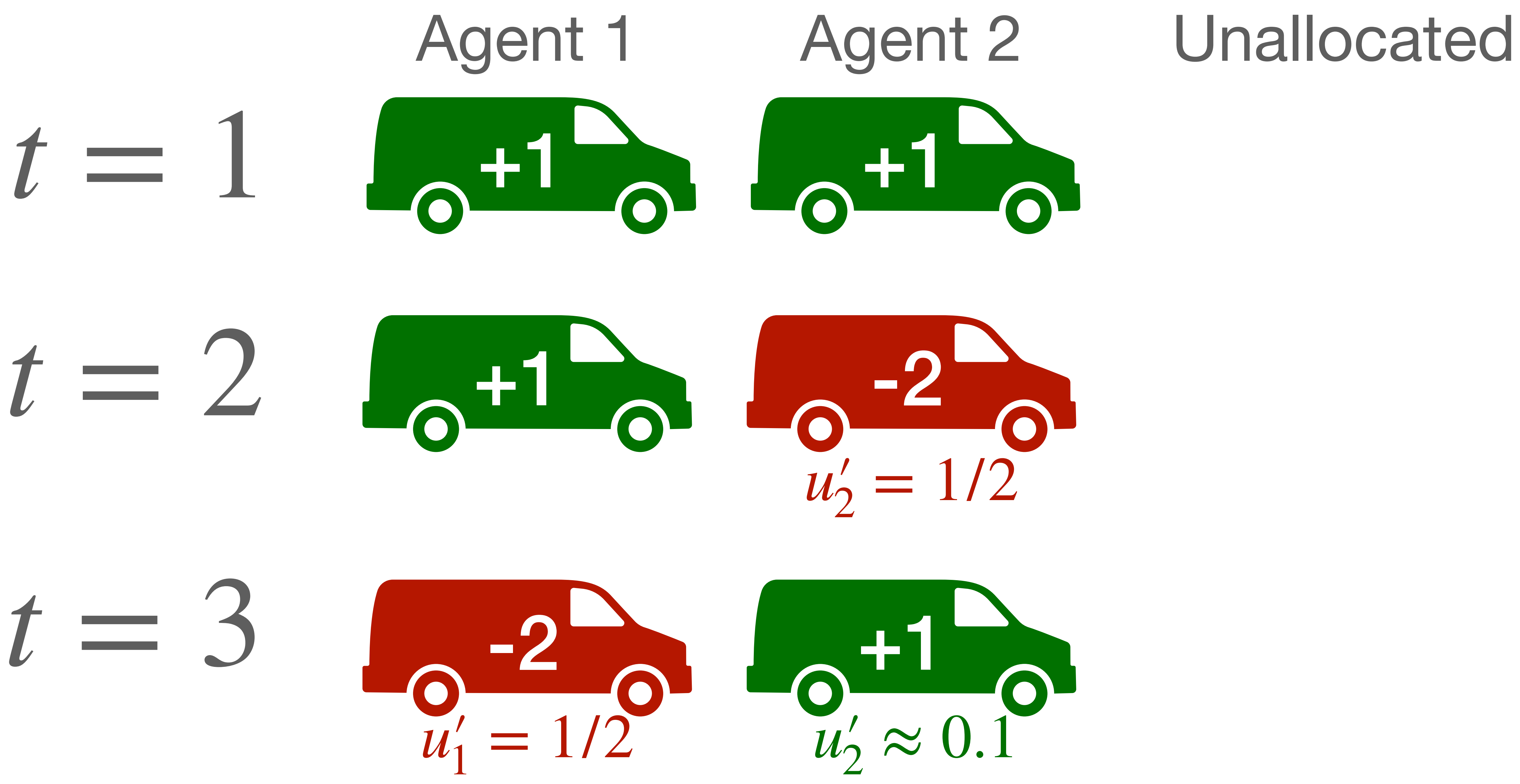


Assign one to each agent: everyone is happy!

Assign both items, but need to promise 1/2 for agent to accept bad item

Illustrating the Optimal Contract

Two agents, $\delta^A = 0.8$ and a patient principle, $\delta^P = 0.9$.
Two items arrive each period i.i.d. value -2 or +1 with equal probability.



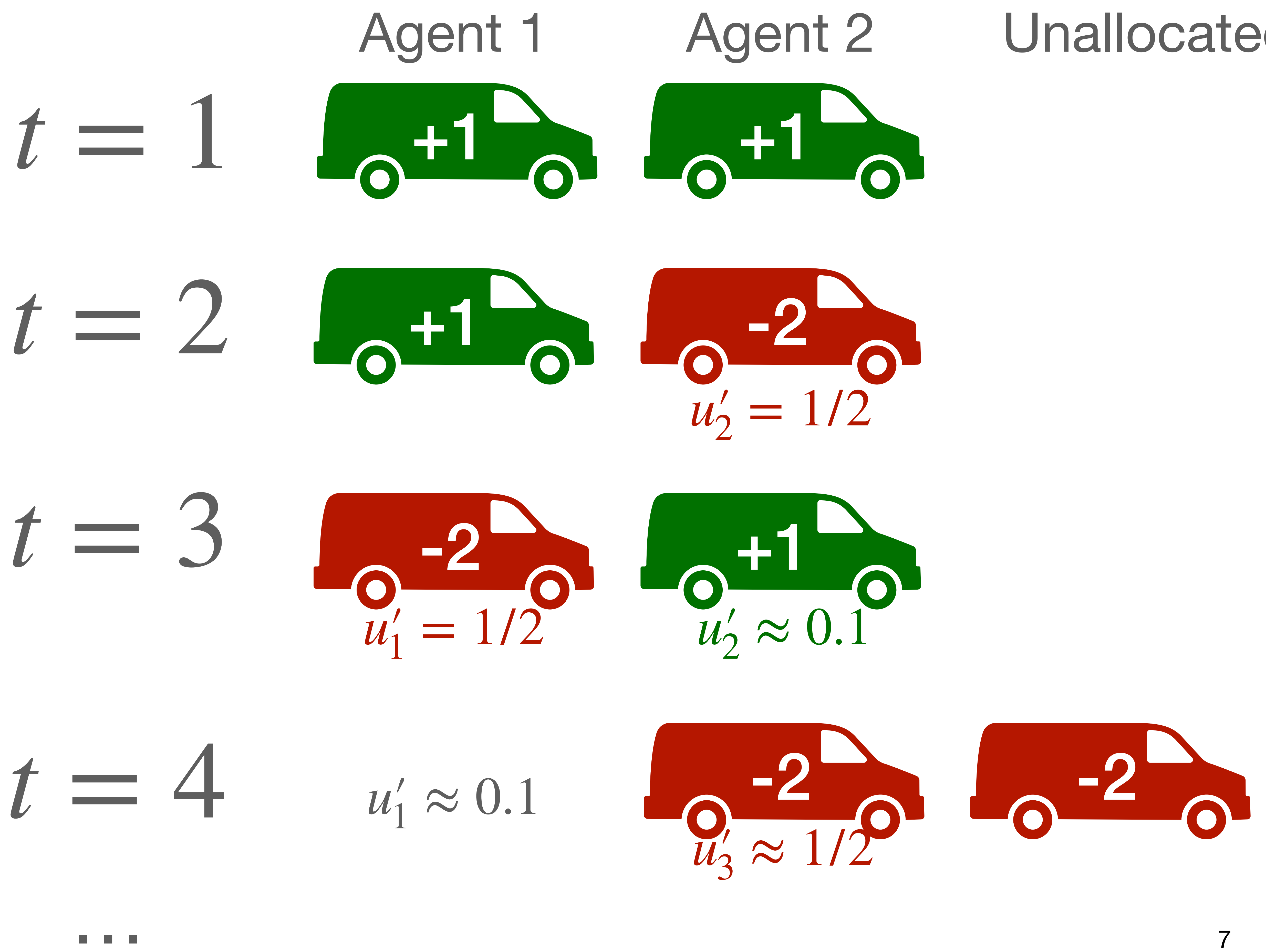
Assign one to each agent: everyone is happy!

Assign both items, but need to promise 1/2 for agent to accept bad item

Assign good item to agent 2 and reduce promise, assign bad item to agent 1 and increase promise

Illustrating the Optimal Contract

Two agents, $\delta^A = 0.8$ and a patient principle, $\delta^P = 0.9$.
Two items arrive each period i.i.d. value -2 or +1 with equal probability.



Assign one to each agent: everyone is happy!

Assign both items, but need to promise 1/2 for agent to accept bad item

Assign good item to agent 2 and reduce promise, assign bad item to agent 1 and increase promise

Assign only one bad item and reduce promise to agent 1

Schur-Concavity + Intuition for Loyalty

Majorization pre-order: $u \preceq u'$ if after re-ordering components of u and u' in increasing order, we have that for all k

$$\sum_{i=1}^{k} u_i \leq \sum_{i=1}^{k} u_i', \text{ and } \sum_{i=1}^{N} u_i = \sum_{i=1}^{N} u_i'.$$

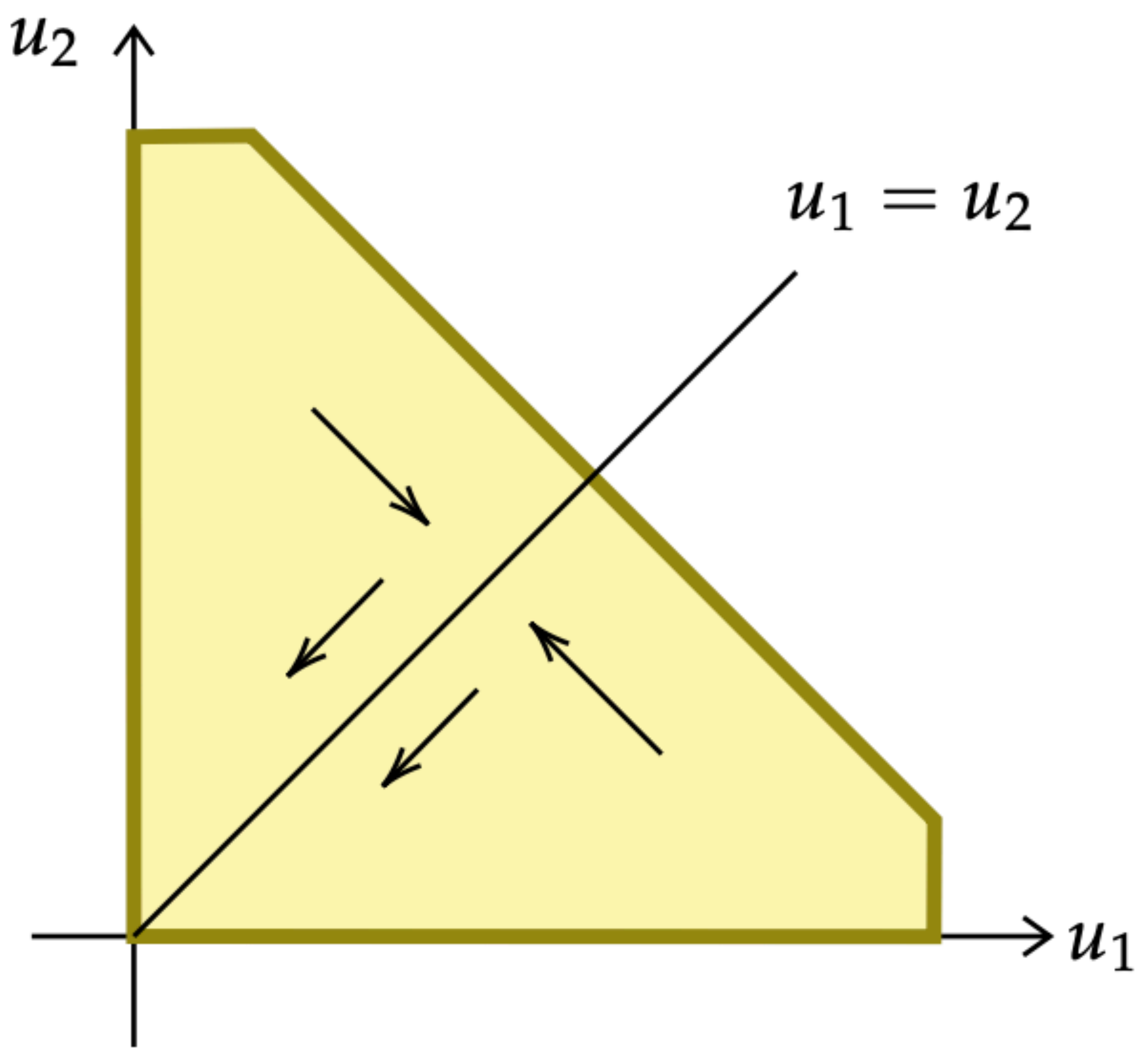
For example,

$$\left(\frac{1}{n},\ldots,\frac{1}{n}\right) \preceq \left(\frac{1}{n-1},\ldots,\frac{1}{n-1},0\right) \preceq \ldots \preceq \left(\frac{1}{2},\frac{1}{2},\ldots\right) \preceq (1,0,\ldots 0)$$

Symmetry + Concavity ⇒ Schur-concavity: Φ decreases in majorization pre-order

Schur-Ostrowski criterion:

Φ is Schur-concave iff $u_i < u_j \Rightarrow \dfrac{\partial \Phi}{\partial u_i} > \dfrac{\partial \Phi}{\partial u_j}$



Schur-Concavity of Φ implies principal prefers *equalization* of promised utilities among agents

Endogenous Loyalty

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|\,+\delta_P\Phi\left(u'(v;u)\right)\right] \text{ subject to}$$

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\right] \geq u_i, \text{ for each } i, \quad \lambda_i(u)$$

$$(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u) \geq 0, \text{ for each } i \text{ and } v\,.\ \mu_i(v;u)$$

Endogenous Loyalty

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v), u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|X(v;u)| + \delta_P \Phi\left(u'(v;u)\right)\right] \text{ subject to}$$

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u) + \delta_A u_i'(v;u)\right] \geq u_i, \text{ for each } i, \quad \lambda_i(u)$$

$$(1-\delta_A)v_i^X(v;u) + \delta_A u_i'(v;u) \geq 0, \text{ for each } i \text{ and } v\,. \ \mu_i(v;u)$$

Optimality for X and envelope theorem implies:

$$X(v;u) \text{ solves } \max_{X\in\mathcal{X}(v)} (1-\delta_P)\,|X| + (1-\delta_A)\lambda(u)\cdot v^X + (1-\delta_A)\mu(v;u)\cdot v^X$$

$$D_u\Phi(u) = -\lambda(u)$$

Endogenous Loyalty

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v), u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|\,+\delta_P\Phi\left(u'(v;u)\right)\right] \text{ subject to}$$

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\right] \geq u_i, \text{ for each } i, \quad \lambda_i(u)$$

$$(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u) \geq 0, \text{ for each } i \text{ and } v\,.\ \mu_i(v;u)$$

Optimality for X and envelope theorem implies:

$$X(v;u) \text{ solves } \max_{X\in\mathcal{X}(v)} (1-\delta_P)\,|\,X\,|\,+(1-\delta_A)\lambda(u)\cdot v^X+(1-\delta_A)\mu(v;u)\cdot v^X$$

$$D_u\Phi(u) = -\lambda(u)$$

For simplicity, consider $v \gg 0$. Then $X(v;u)$ solves $\max_{X\in\mathcal{X}(v)} (1-\delta_P)\,|\,X\,|\,-(1-\delta_A)D_u\Phi(u)\cdot v^X$

But Schur-concavity implies $u_i < u_j \Rightarrow \dfrac{\partial\Phi}{\partial u_i} > \dfrac{\partial\Phi}{\partial u_j}$, but then high v_i^X should be paired with high $-\dfrac{\partial\Phi}{\partial u_i} \Rightarrow$ assortativity.

Long-Run Performance

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|+\delta_P\Phi\left(u'(v;u)\right)\right]$$ subject to

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\right]\geq u_i,$$ for each i, $\lambda_i(u)$

$$(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\geq 0,$$ for each i and v . $\mu_i(v;u)$

Long-Run Performance

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|\, + \delta_P\Phi\left(u'(v;u)\right)\right] \text{ subject to}$$

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u) + \delta_A u_i'(v;u)\right] \geq u_i, \text{ for each } i, \quad \lambda_i(u)$$

$$(1-\delta_A)v_i^X(v;u) + \delta_A u_i'(v;u) \geq 0, \text{ for each } i \text{ and } v\,. \ \mu_i(v;u)$$

First-order conditions for $u'(v;u)$ imply that

$$\delta_P D_u\Phi(u'(v;u)) + \delta_A\lambda(u) + \delta_A\mu(v;u) = 0 \Rightarrow D_u\Phi(u'(v;u)) = -\frac{\delta_A}{\delta_P}(\lambda(u) + \mu(v;u))$$

Envelope Theorem implies that

$$D_u\Phi(u) = -\lambda(u)$$

Long-Run Performance

$$\Phi(u) = \max_{X(v;u)\in\mathcal{X}(v),u'(v;u)\in\mathcal{U}} \mathbb{E}_{v\sim F}\left[(1-\delta_P)\,|\,X(v;u)\,|+\delta_P\Phi\left(u'(v;u)\right)\right]$$ subject to

$$\mathbb{E}_{v\sim F}\left[(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u)\right] \geq u_i,$$ for each i, $\lambda_i(u)$

$$(1-\delta_A)v_i^X(v;u)+\delta_A u_i'(v;u) \geq 0,$$ for each i and v . $\mu_i(v;u)$

First-order conditions for $u'(v;u)$ imply that

$$\delta_P D_u\Phi(u'(v;u)) + \delta_A\lambda(u) + \delta_A\mu(v;u) = 0 \Rightarrow D_u\Phi(u'(v;u)) = -\frac{\delta_A}{\delta_P}(\lambda(u)+\mu(v;u))$$

Envelope Theorem implies that

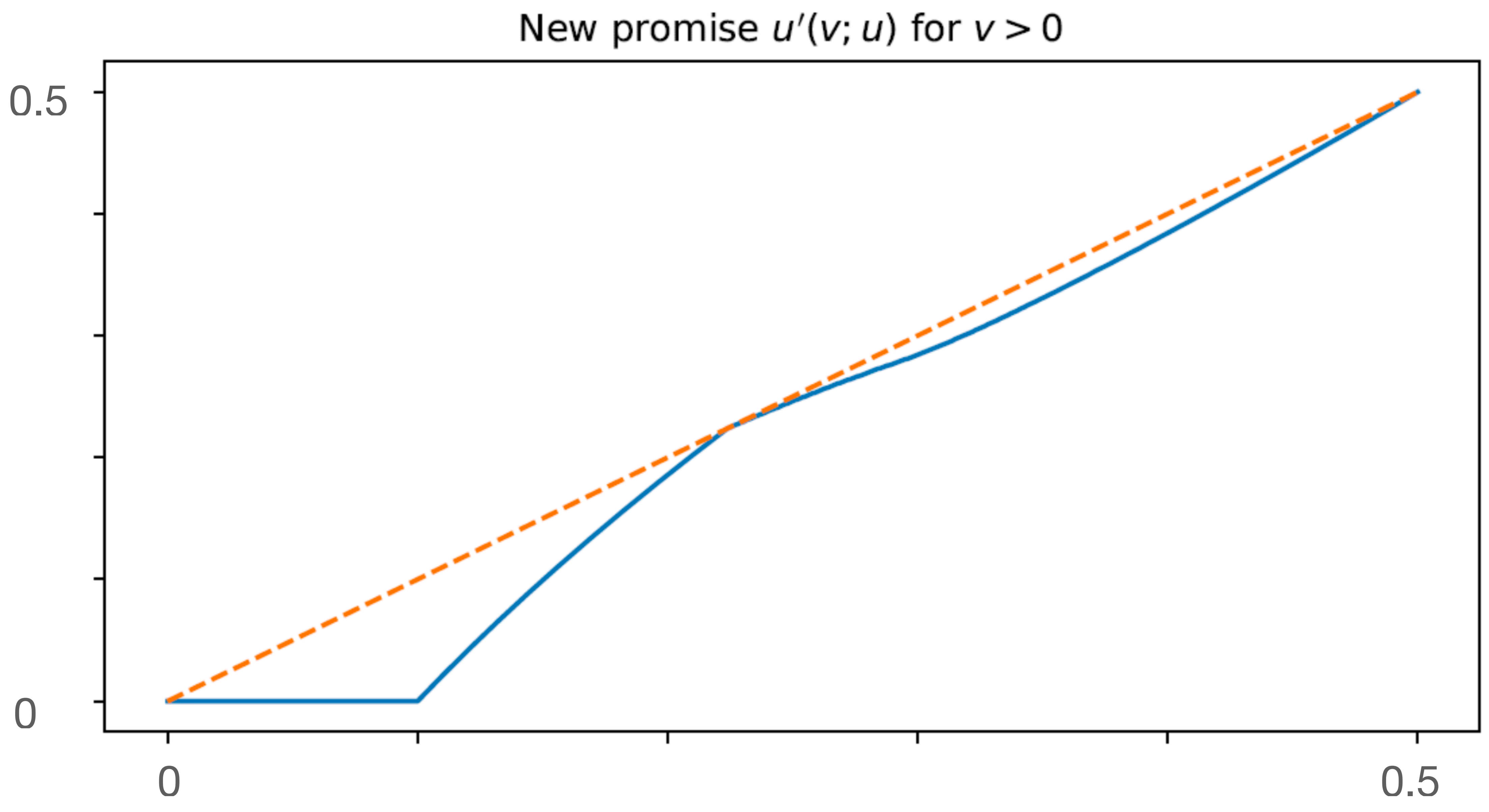
$$D_u\Phi(u) = -\lambda(u)$$

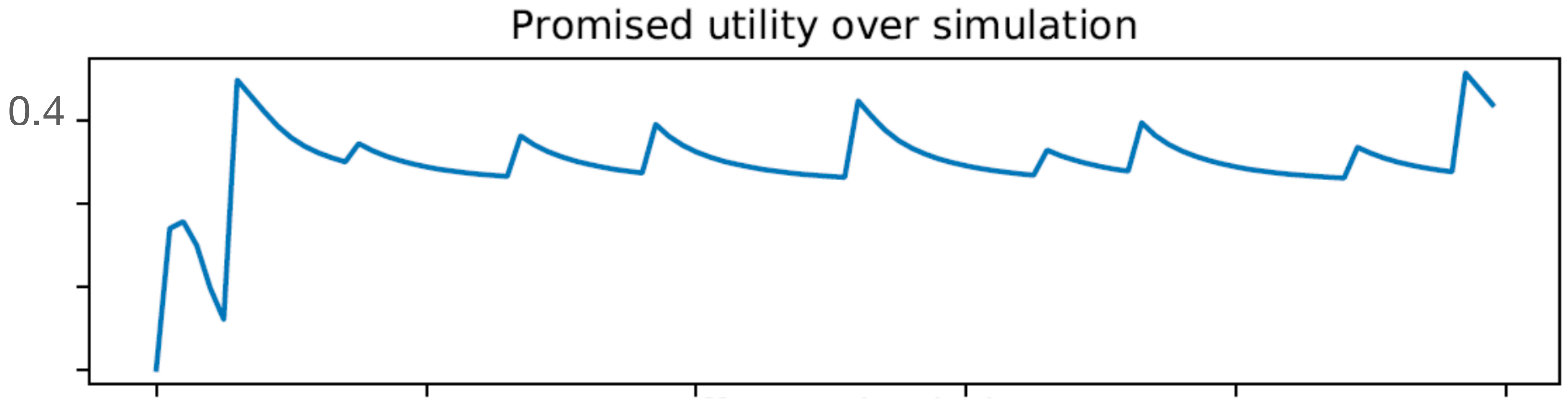
Together, implies for $v \gg 0$ that $D_u\Phi(u'(v;u)) = \frac{\delta_A}{\delta_P}D_u\Phi(u'(v;u))$

Illustrating Long-Run Performance

Consider single-agent version of previous example:

If principle is *more patient* than agent, it ‘works’ off promises over time and more bad items are allocated in long-run, e.g. $\delta_P = 0.9 > \delta_A = 0.8$

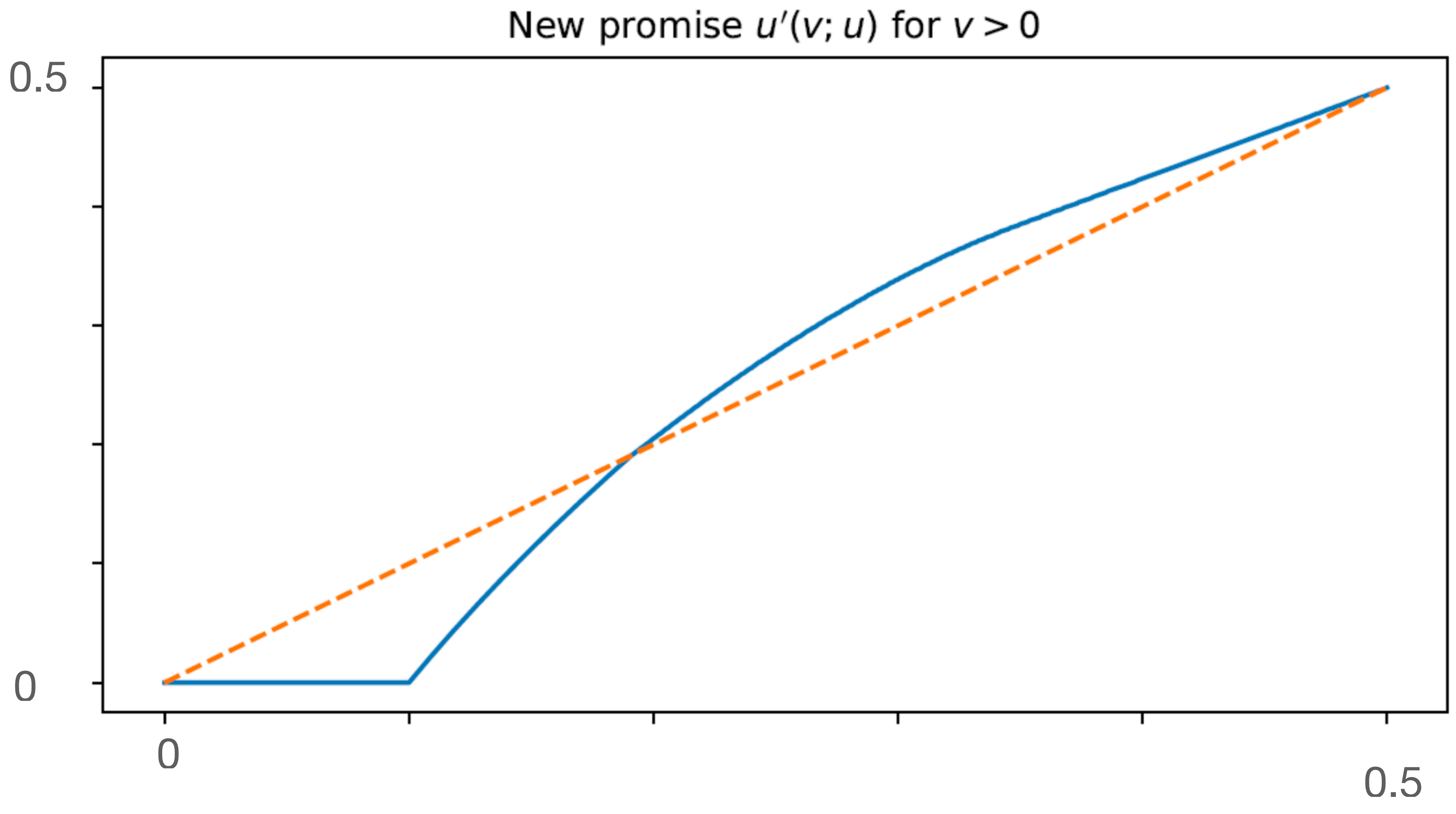


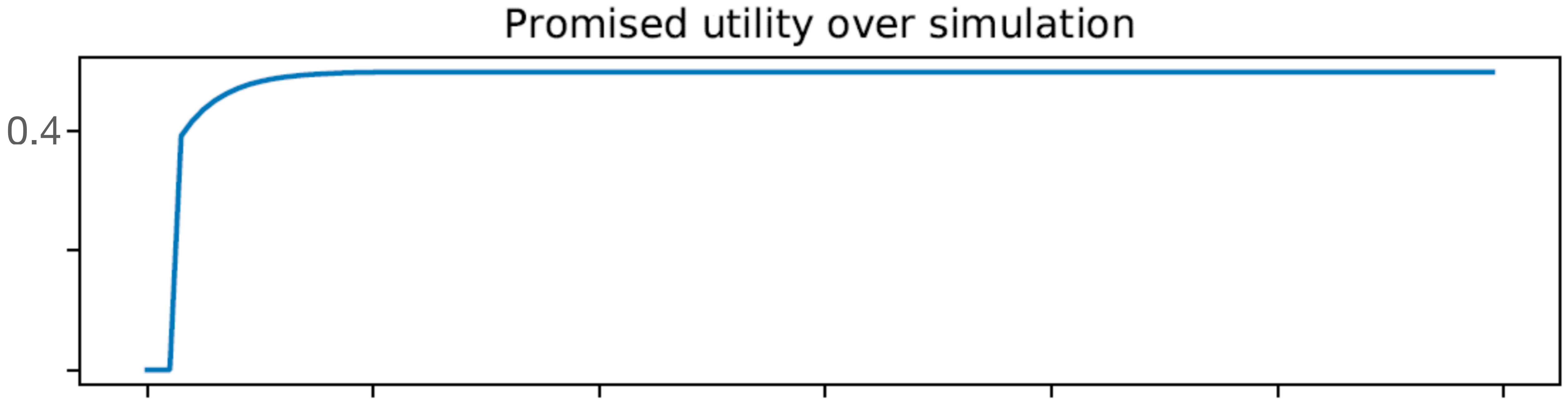


Illustrating Long-Run Performance

Consider single-agent version of previous example:

If principle is *less patient* than agents, it always promises more to incentivize bad item allocation today: eventually only good items allocated (c.f. ‘immiseration’).





Conclusion

I introduce a repeated matching model with a fixed population of agents and a period-by-period *ex post* participation constraint.

1. The principal promises better future allocations to agents who accept bad items in current period.
2. Endogenous 'loyalty': agents with the worst historical allocations (= highest promised utility) have priority for better items.
3. Optimal policy is a cutoff rule.
4. Long-run performance depends on ratio of δ^P to δ^A.

Implication: My results suggest that ride-sharing apps could benefit from using dynamic incentives, transforming first-come-first-served to worst-off first-served. Practical implementation could include 'fast pass' in airport queues.