towards stronger fairness guarantees

Rohit Vaish

The Model

Set of agents

Set of indivisible items

A B C D E

Allocation

Envy-Freeness

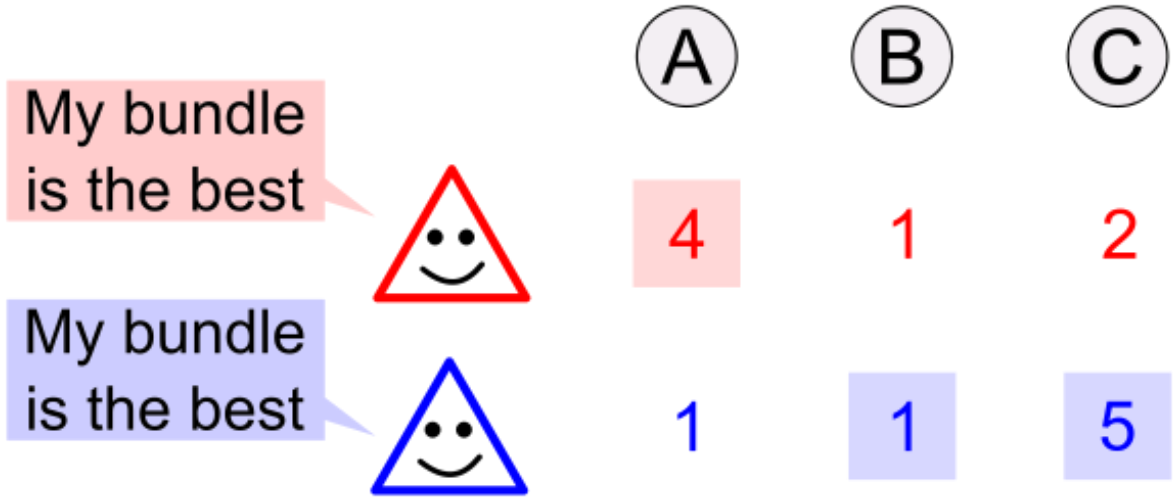
[Gamow and Stern, 1958; Foley, 1967]

Each agent prefers its own bundle over that of any other agent.

Envy-Freeness

[Gamow and Stern, 1958; Foley, 1967]

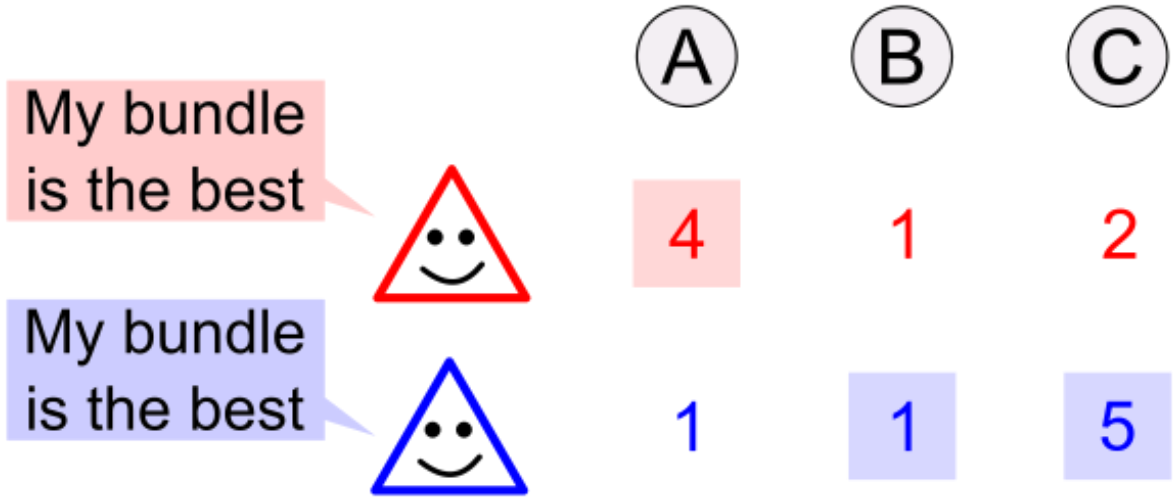
Each agent prefers its own bundle over that of any other agent.



Envy-Freeness

[Gamow and Stern, 1958; Foley, 1967]

Each agent prefers its own bundle over that of any other agent.



☹ Not guaranteed to exist (two agents, one good)

☹ Checking whether an EF allocation exists is NP-complete

Envy-Freeness Up To One Good

[Budish, 2011]

Envy can be eliminated by removing some good in the envied bundle.

Envy-Freeness Up To One Good

[Budish, 2011]

Envy can be eliminated by removing some good in the envied bundle.

My bundle is better if Ⓐ is removed

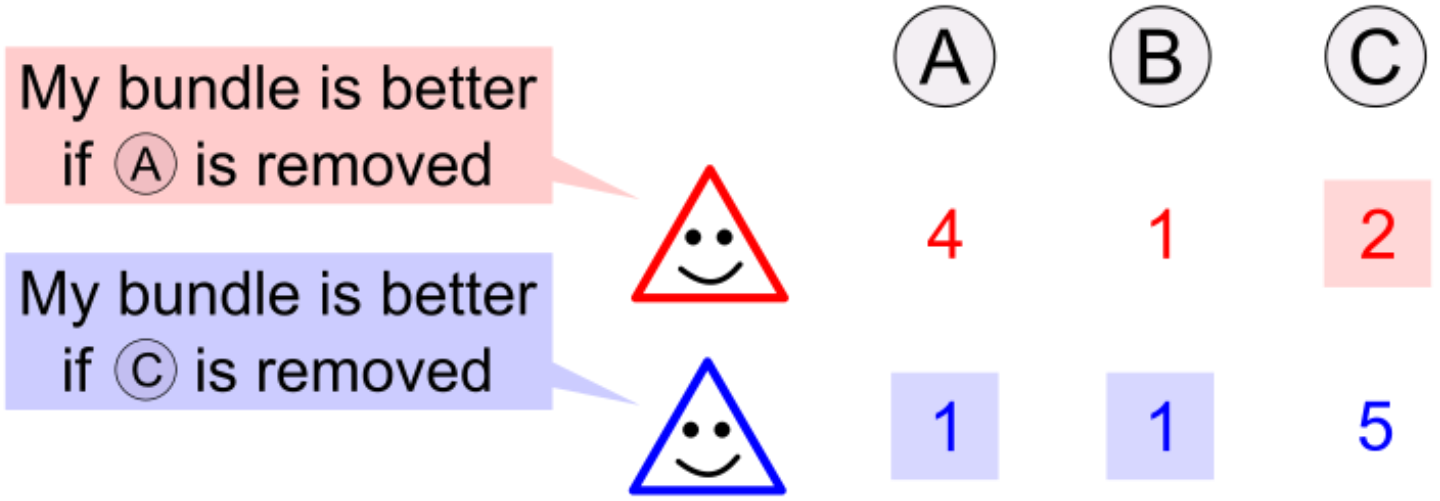
My bundle is better if Ⓒ is removed

Ⓐ	Ⓑ	Ⓒ
4	1	2
1	1	5

Envy-Freeness Up To One Good

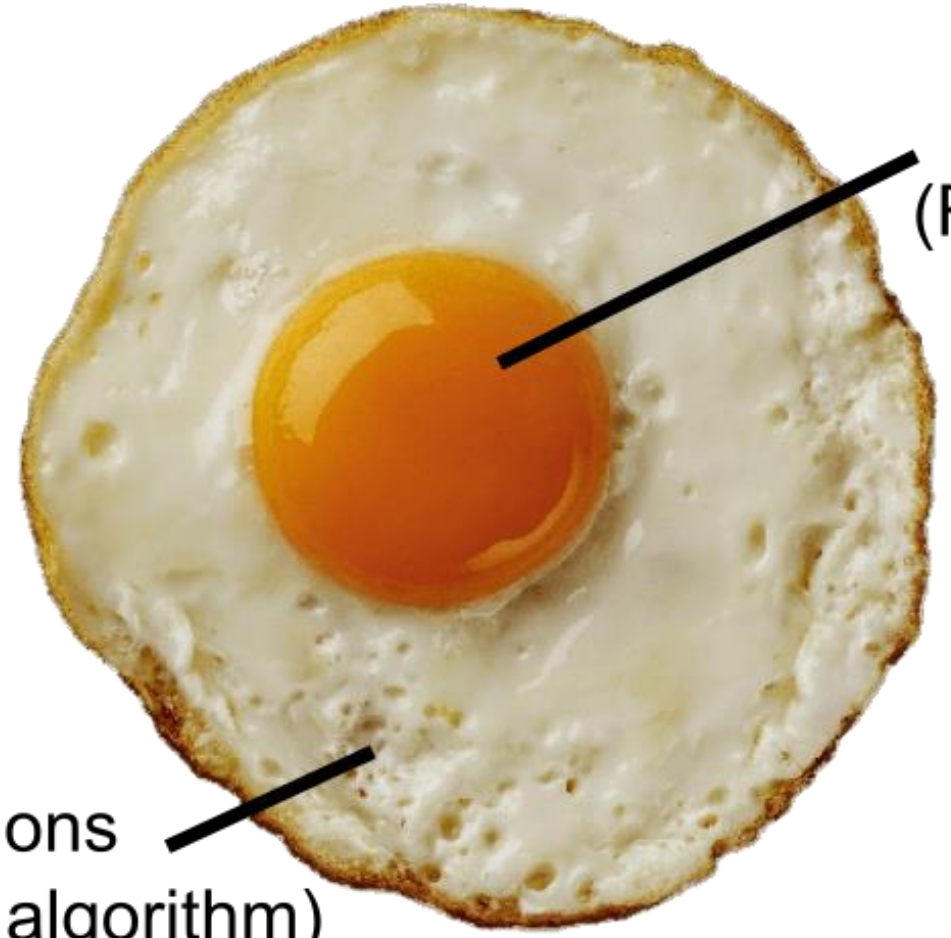
[Budish, 2011]

Envy can be eliminated by removing some good in the envied bundle.



Guaranteed to exist and efficiently computable

Algorithms for finding an EF1 allocation



Additive valuations
(Round-robin algorithm)

Monotone valuations
(Envy-cycle elimination algorithm)

A Limitation of EF1

A Limitation of EF1

A Limitation of EF1

I donut think this is fair!

Why are you sad?
Aren't you envy-free up to a car?

A Limitation of EF1

Envy-Freeness Up To Any Good

[Caragiannis, Kurokawa, Moulin, Procaccia, Shah, and Wang, *EC* 2016, *TEAC* 2019]

Envy can be eliminated by removing any good in the envied bundle.

Envy-Freeness Up To Any Good

[Caragiannis, Kurokawa, Moulin, Procaccia, Shah, and Wang, *EC* 2016, *TEAC* 2019]

Envy can be eliminated by removing any good in the envied bundle.

	A	B	C	D
	3	3	4	1
	1	1	0	2

Allocation $A = (A_1, \ldots, A_n)$ is EFX if for every pair of agents i, k and for every good $j \in A_k$, we have $v_i(A_i) \geq v_i(A_k \setminus \{j\})$.

For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.

For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

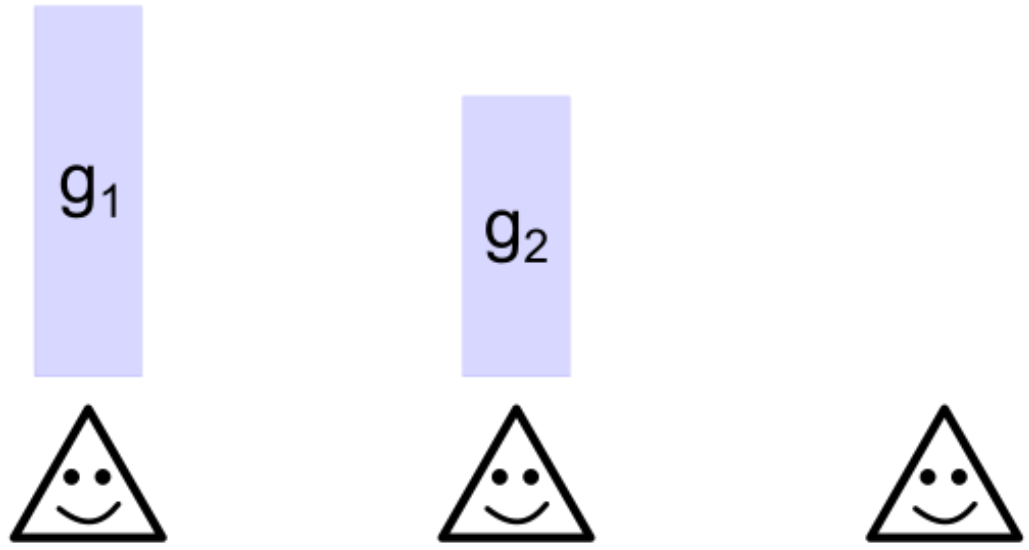
- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.

For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.

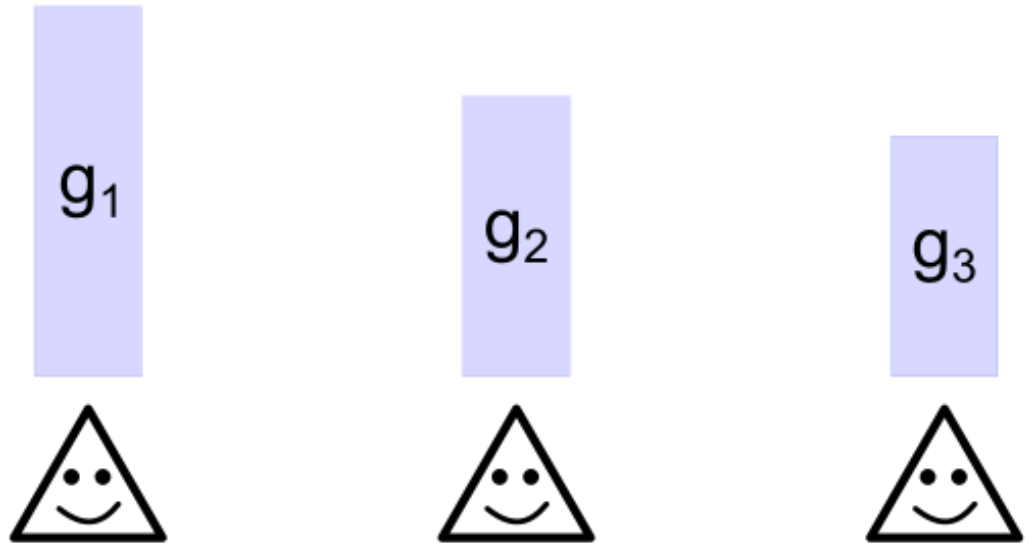
For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.



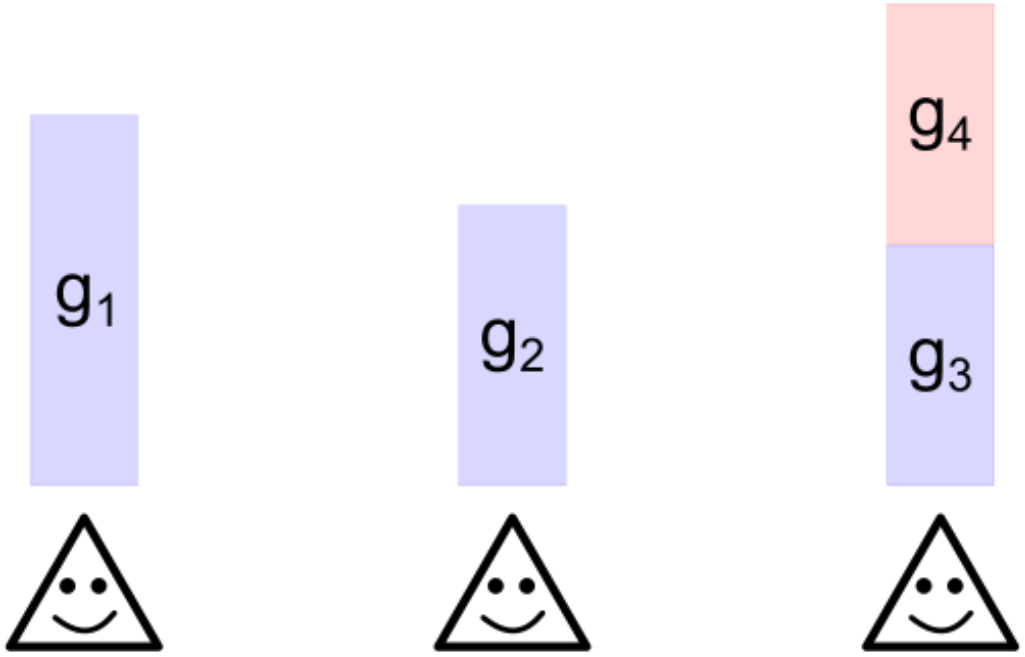
For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.



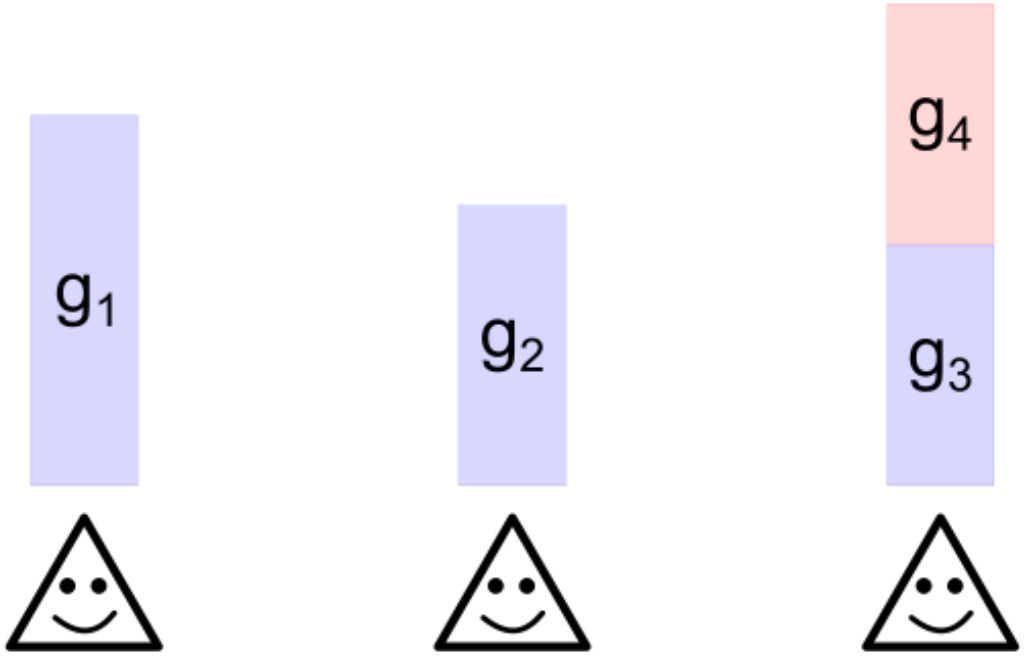
For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.



For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

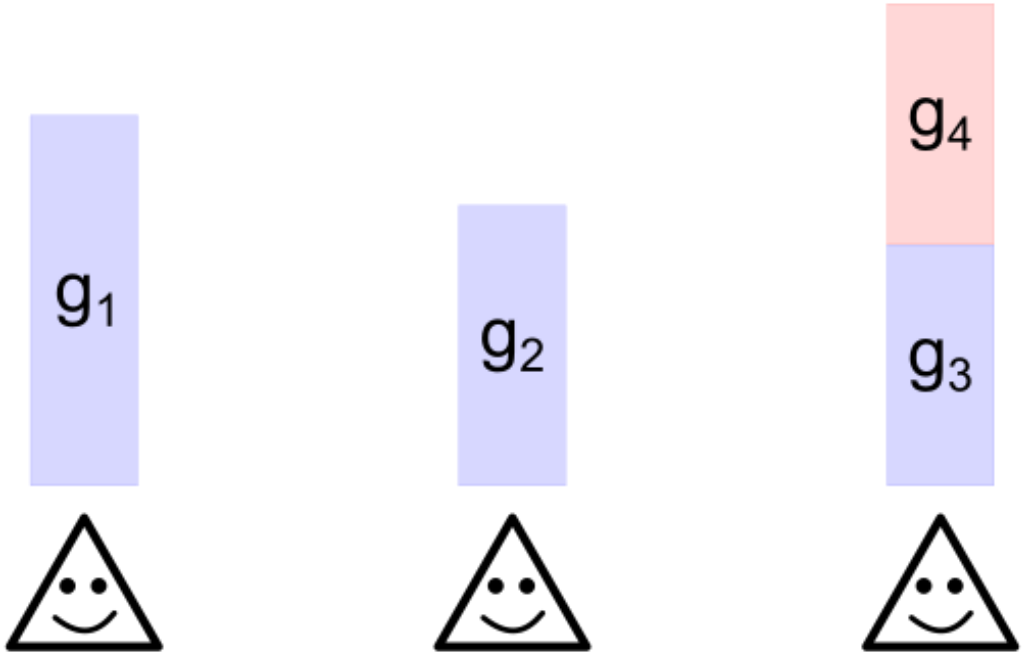
- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.



Why EFX?

For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.

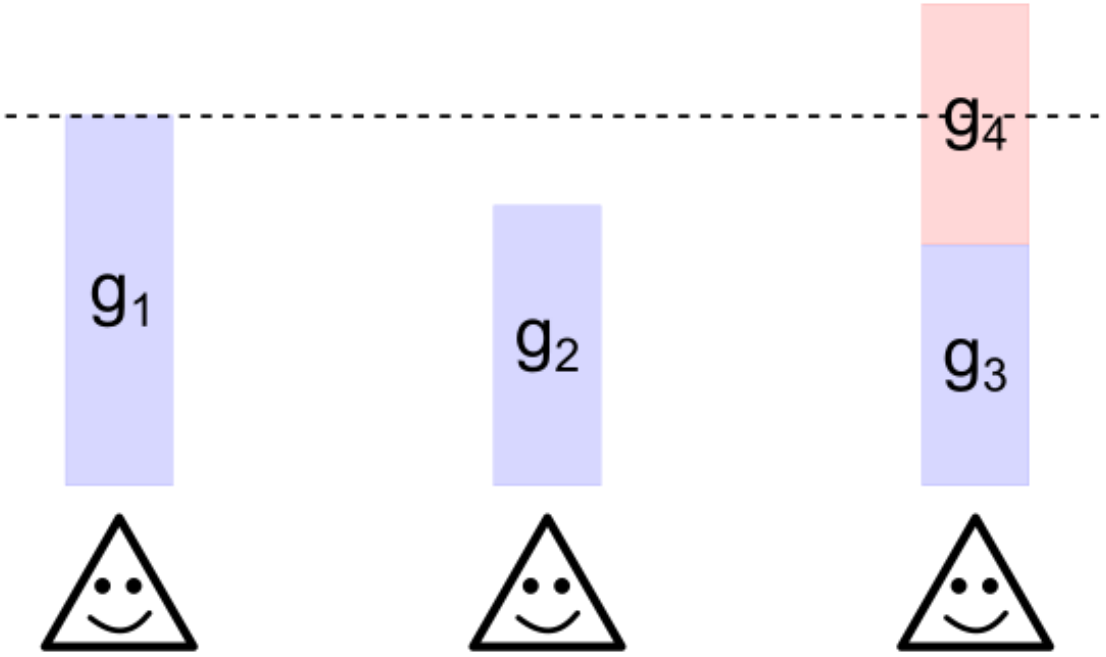


Why EFX?

Most recent good = least valued.

For identical agents with additive valuations over goods, an EFX allocation always exists and can be efficiently computed.

- Allocate the goods in non-increasing order of values.
- Each new good is assigned to the least-happy agent.



Why EFX?

Most recent good = least valued.

Envy-free up to most recent good.

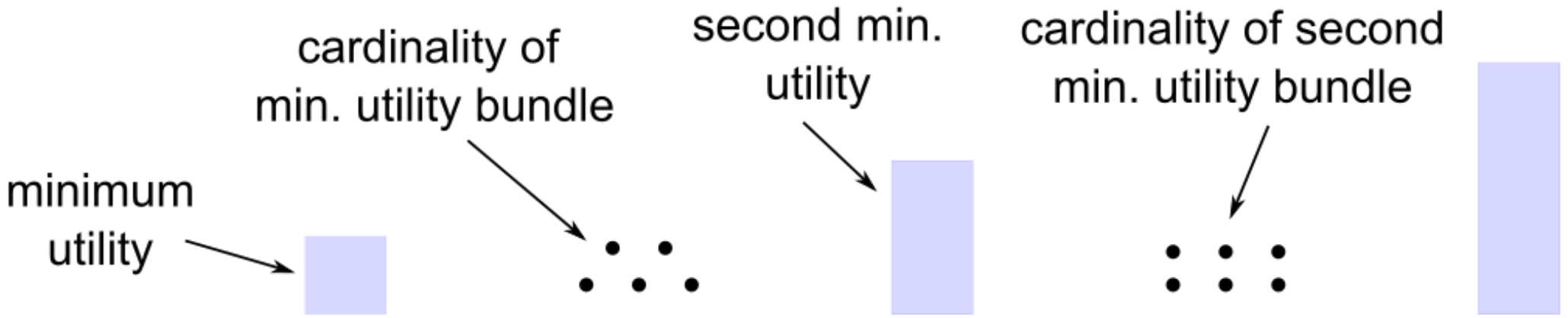
[Plaut and Roughgarden; *SODA* 2018; *SIDMA* 2020]

For identical agents with monotone valuations over goods, an EFX allocation always exists.

[Plaut and Roughgarden; *SODA* 2018; *SIDMA* 2020]

For identical agents with monotone valuations over goods, an EFX allocation always exists.

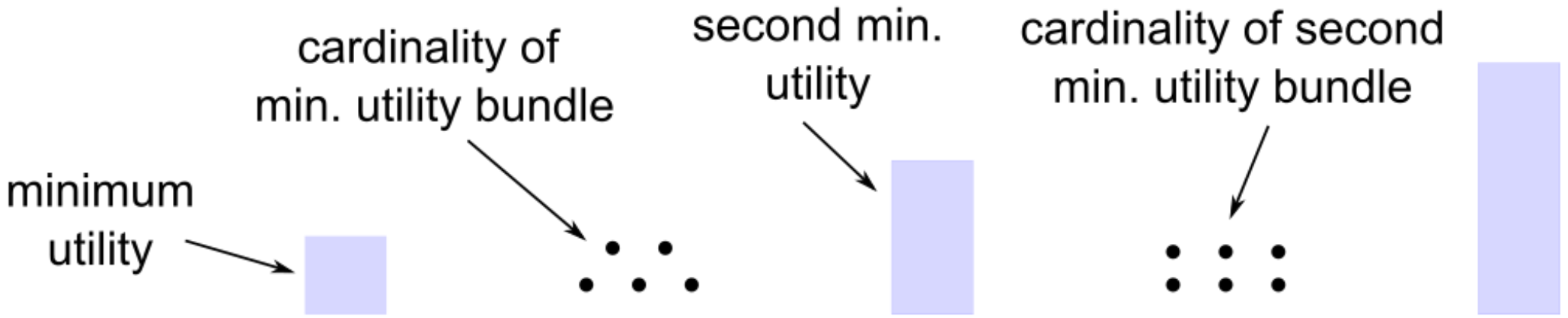
Leximin++: Allocation that lexicographically maximizes



[Plaut and Roughgarden; *SODA* 2018; *SIDMA* 2020]

For identical agents with monotone valuations over goods, an EFX allocation always exists.

Leximin++: Allocation that lexicographically maximizes



[Plaut and Roughgarden; *SODA* 2018; *SIDMA* 2020]

Finding an EFX allocation can take exponential-in-#goods value queries even for two identical agents with submodular valuations.

What about EFX for non-identical agents?

What about EFX for non-identical agents?

Always exists [Plaut and Roughgarden; *SODA* 2018, *SIDMA* 2020]

identical valuations result + "cut and choose"

Exists for two "types" of agents [Mahara, *ESA* 2021]

What about EFX for non-identical agents?

Always exists [Plaut and Roughgarden, *SODA* 2018, *SIDMA* 2020]

identical valuations result + "cut and choose"

Exists for two "types" of agents [Mahara, *ESA* 2021]

Exists for additive [Chaudhury, Garg, and Mehlhorn, *EC* 2020]

iteratively allocate goods + sophisticated update rules + potential argument

What about EFX for non-identical agents?

Always exists [Plaut and Roughgarden, *SODA* 2018, *SIDMA* 2020]

identical valuations result + "cut and choose"

Exists for two "types" of agents [Mahara, *ESA* 2021]

Exists for additive [Chaudhury, Garg, and Mehlhorn, *EC* 2020]

iteratively allocate goods + sophisticated update rules + potential argument

?

What about EFX for non-identical agents?

Always exists [Plaut and Roughgarden, *SODA* 2018, *SIDMA* 2020]

identical valuations result + "cut and choose"

Exists for two "types" of agents [Mahara, *ESA* 2021]

Exists for additive [Chaudhury, Garg, and Mehlhorn, *EC* 2020]

iteratively allocate goods + sophisticated update rules + potential argument

?

Exists for "2 value" additive instances

[Amanatidis, Birmpas, Filos-Ratsikas, Hollender, and Voudouris, *IJCAI* 2020, *TCS* 2021; Garg and Murhekar, *SAGT* 2021]

Fairness via Charity

EFX-with-charity

[Caragiannis, Gravin, and Huang, *EC* 2019; Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

A partition $(A_1, \ldots, A_n, P)$ into $n+1$ bundles satisfies EFX-with-charity if

- the partial allocation $(A_1, \ldots, A_n)$ is EFX,
- no agent envies the pool P of unallocated items, and
- $|P| < n$.

EFX-with-charity

[Caragiannis, Gravin, and Huang, *EC* 2019; Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

A partition $(A_1, \ldots, A_n, P)$ into $n+1$ bundles satisfies EFX-with-charity if

- the partial allocation $(A_1, \ldots, A_n)$ is EFX,
- no agent envies the pool P of unallocated items, and
- $|P| < n$.

For monotone valuations, an EFX-with-charity allocation always exists.

Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

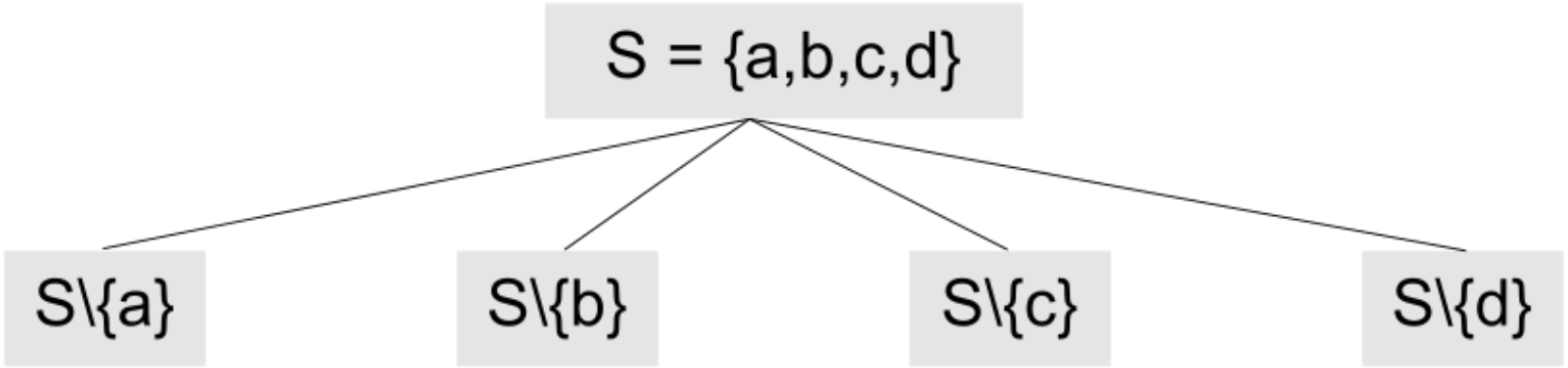
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

$S = \{a,b,c,d\}$

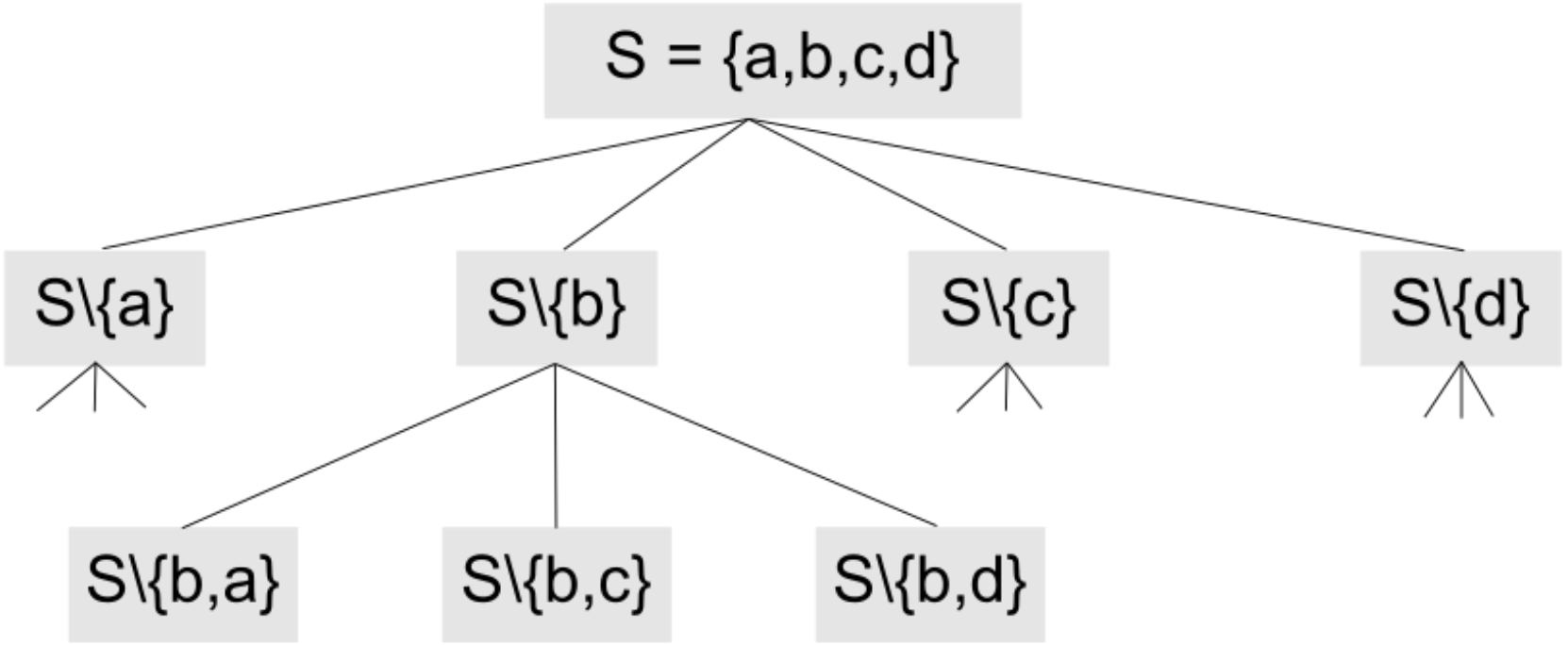
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



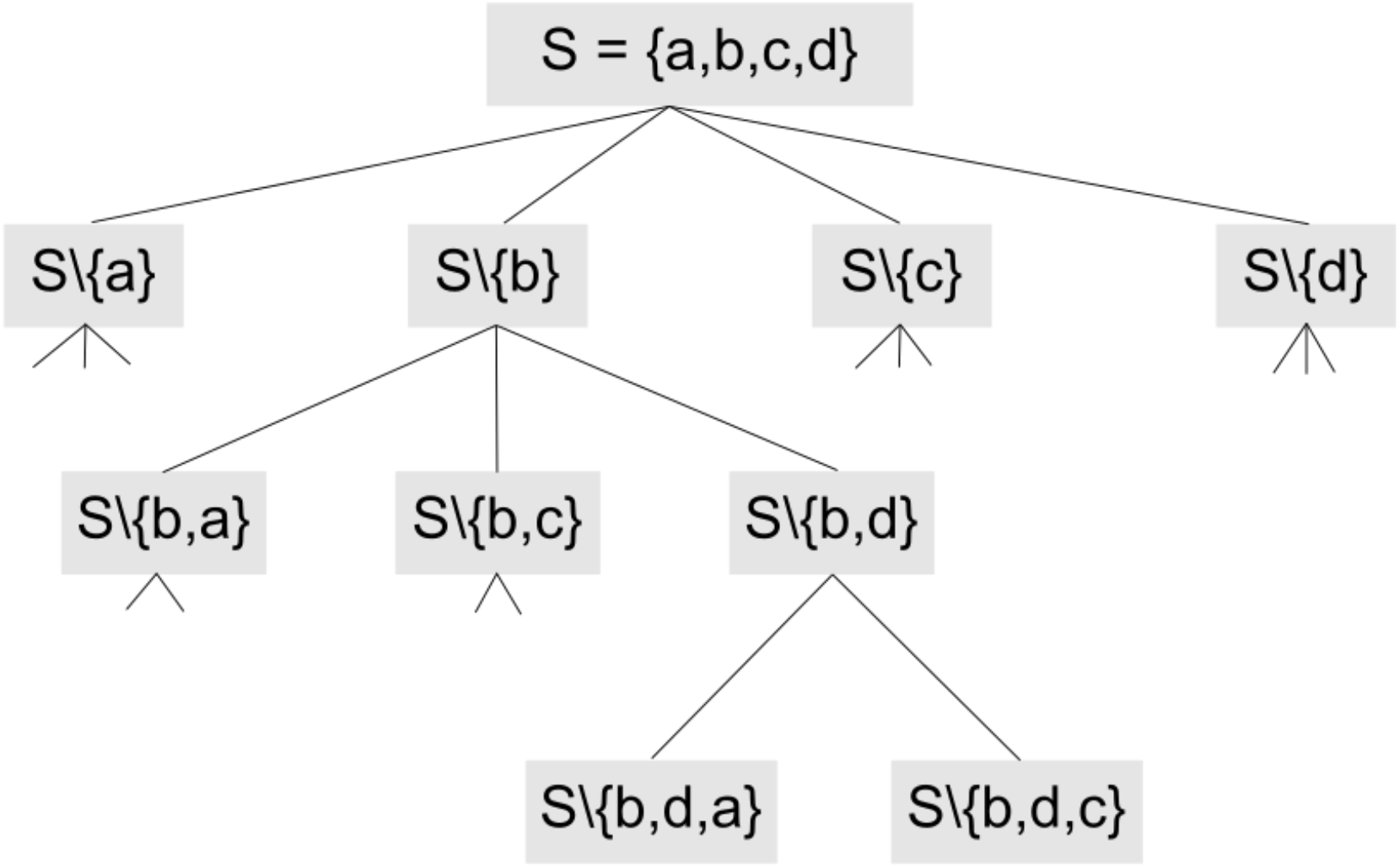
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



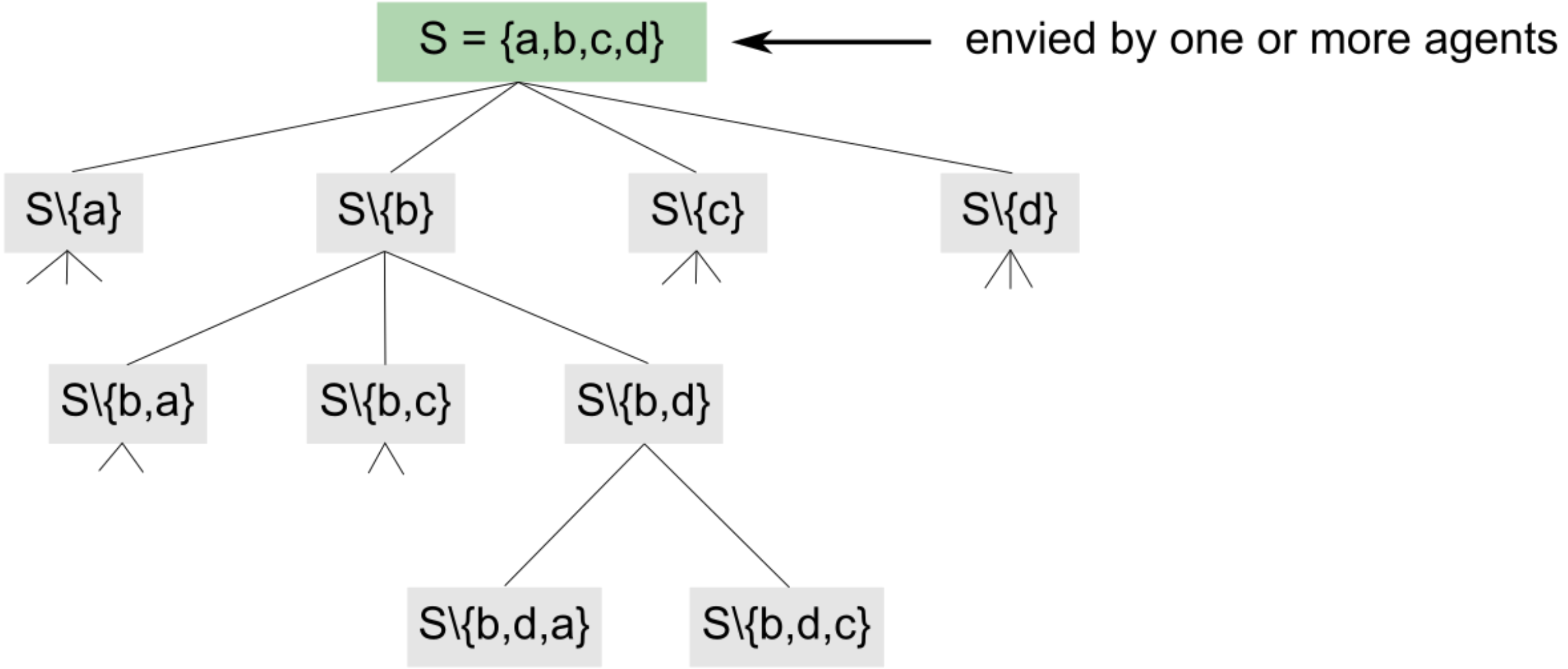
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



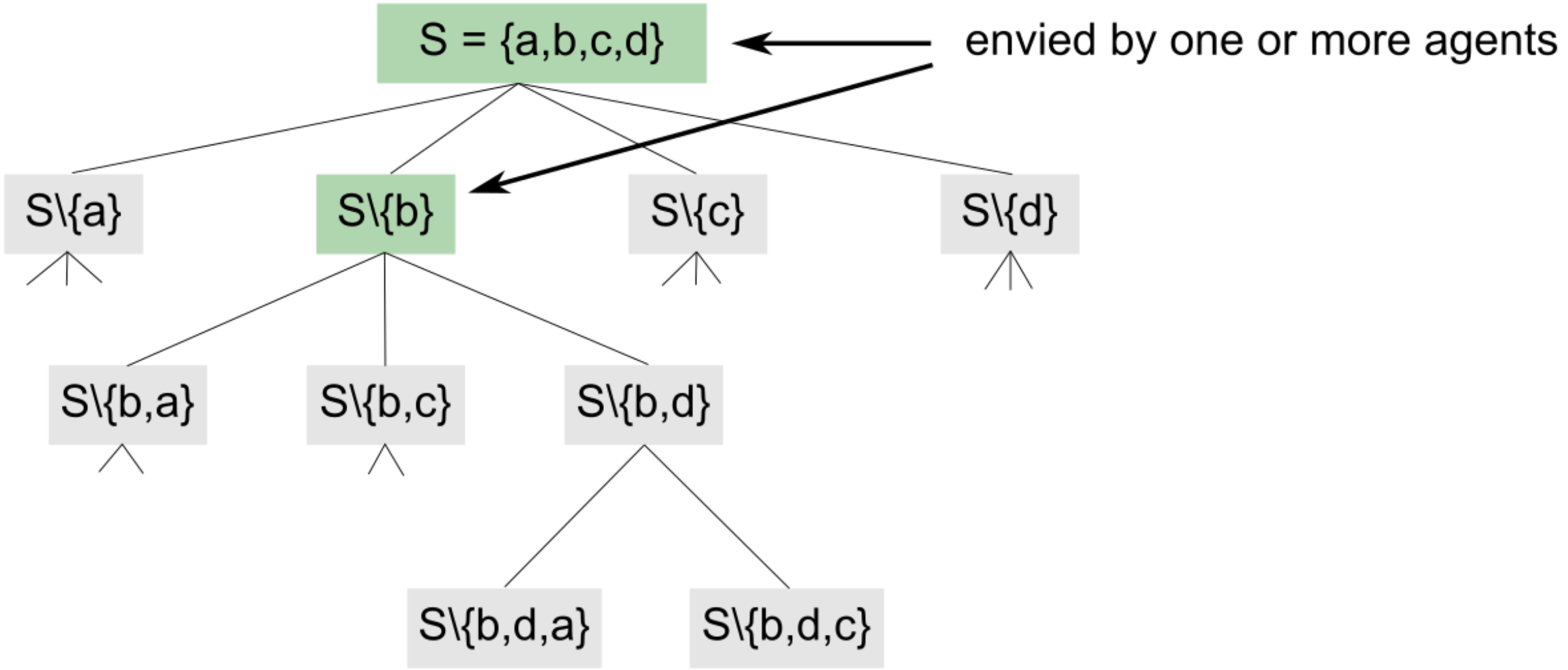
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



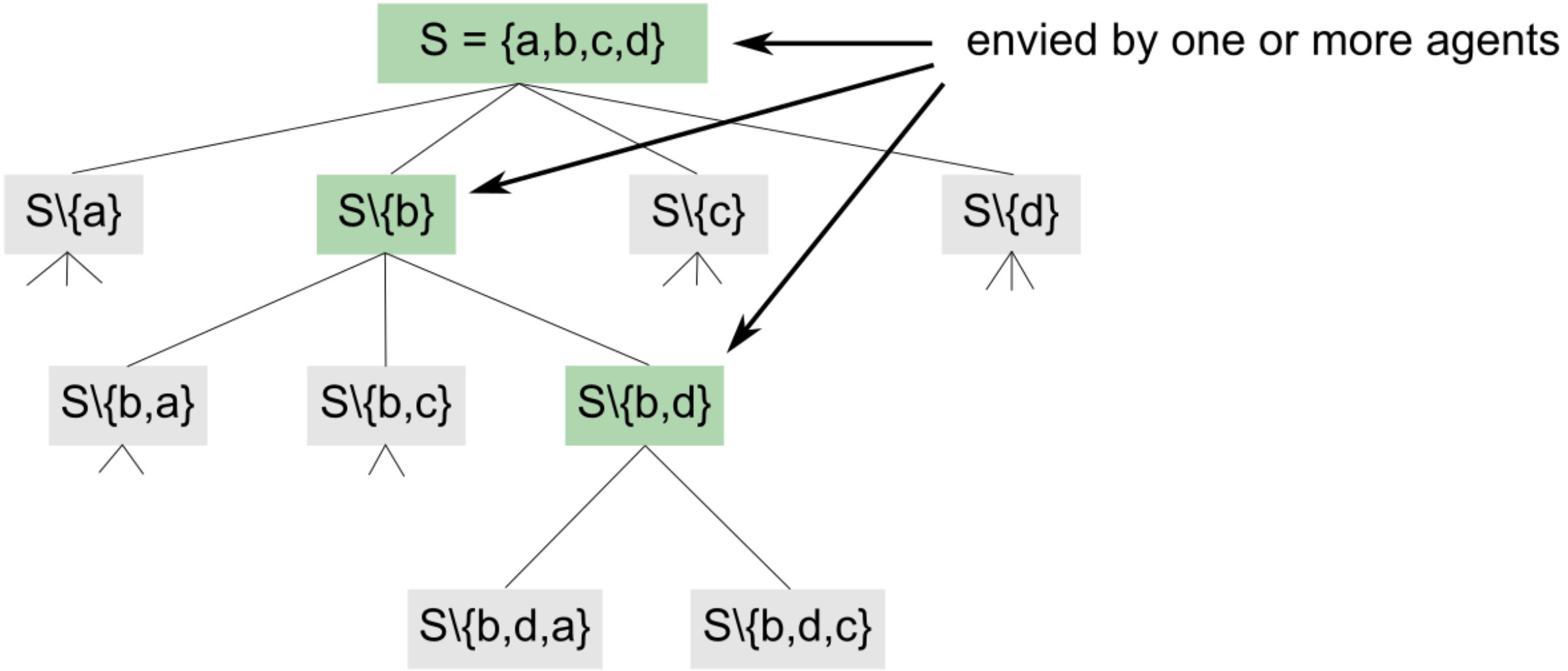
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



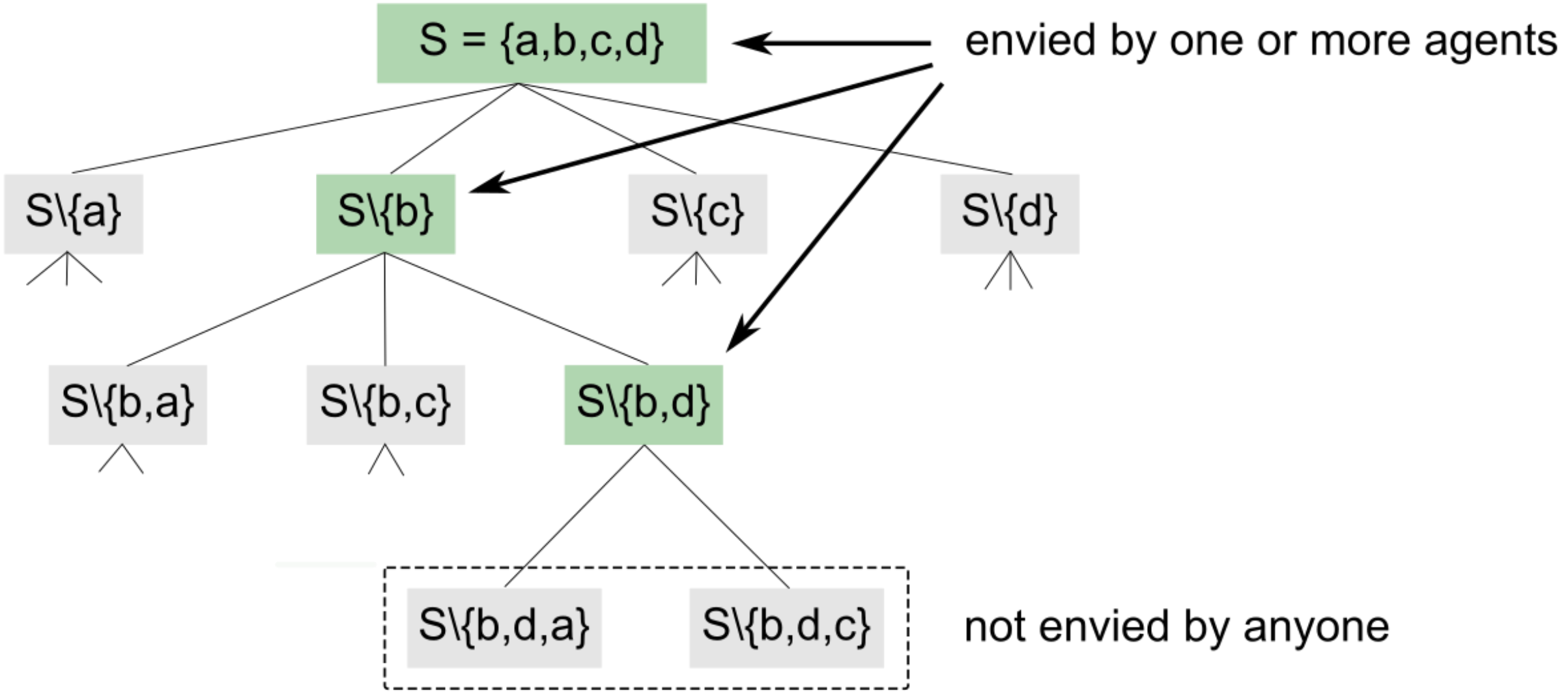
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



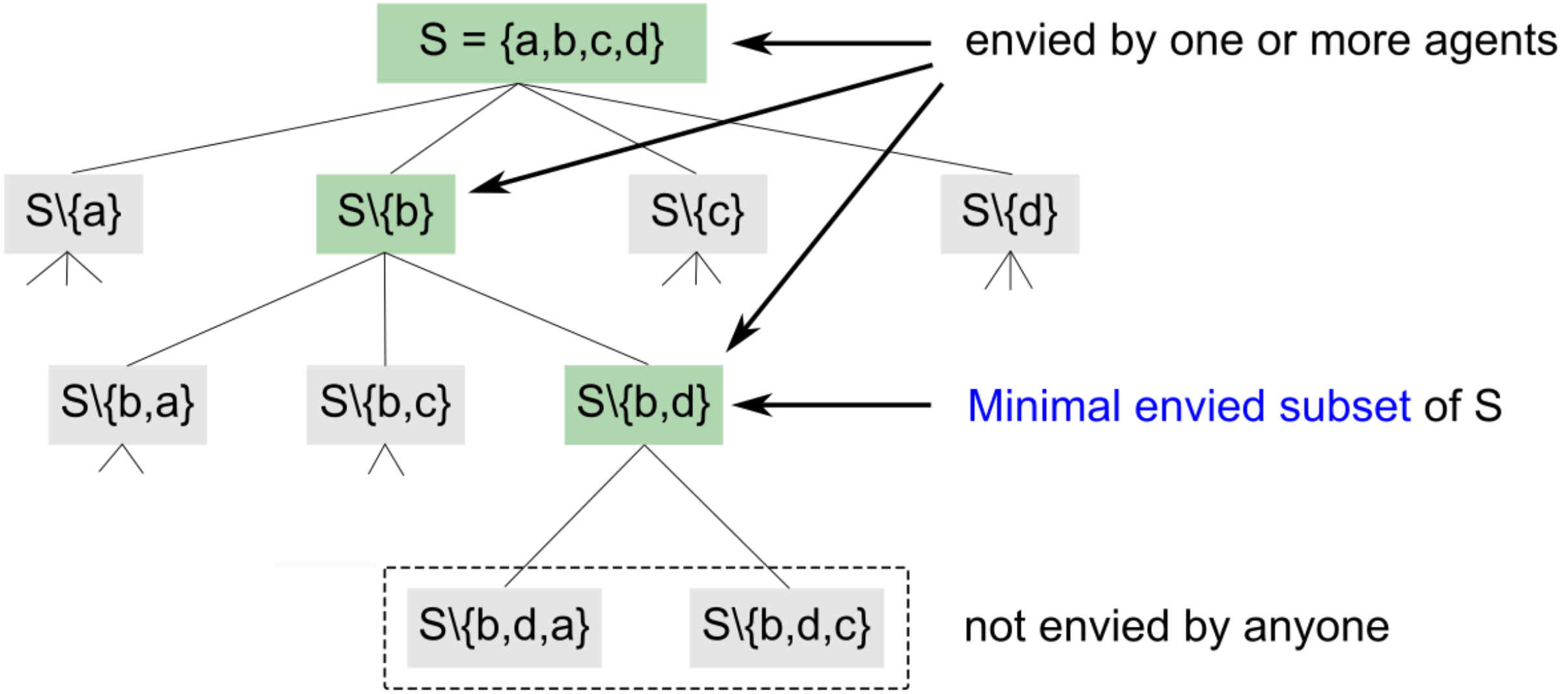
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



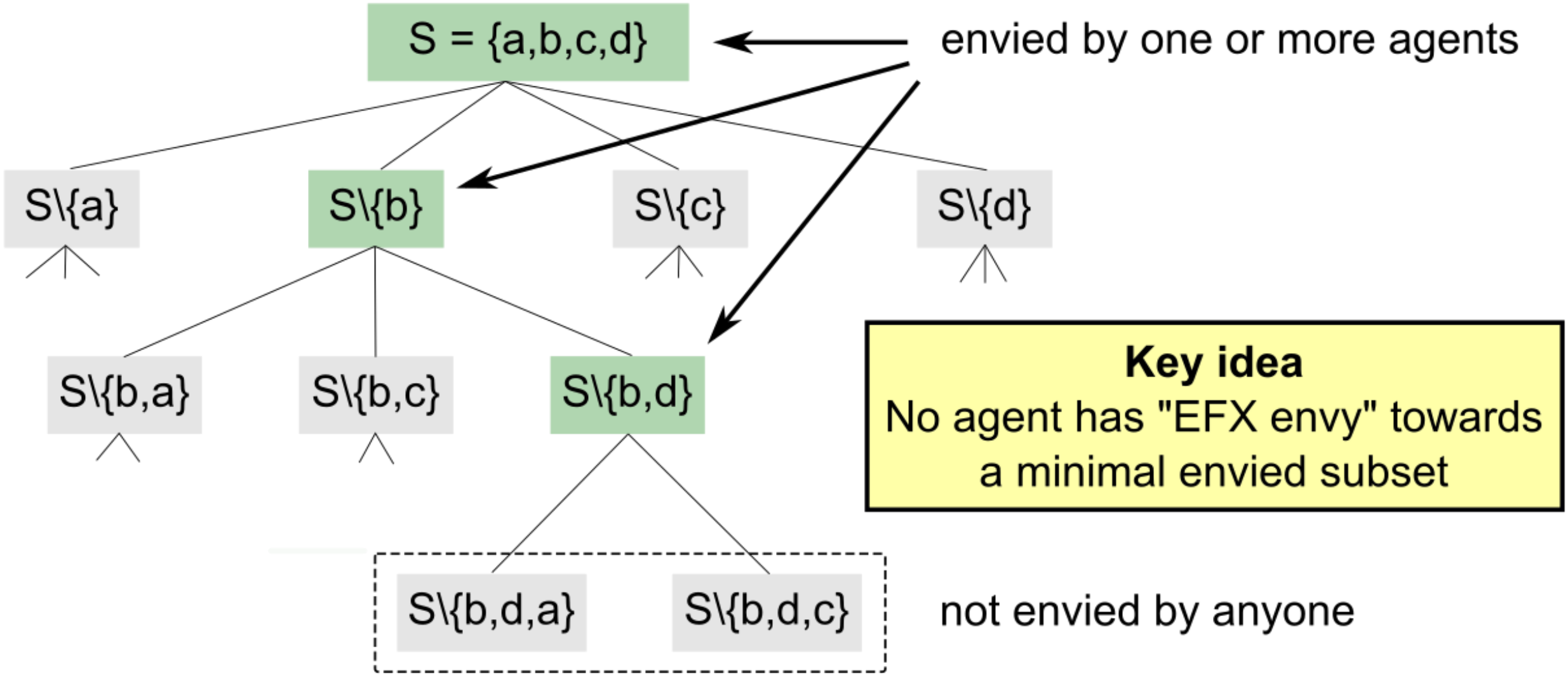
Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

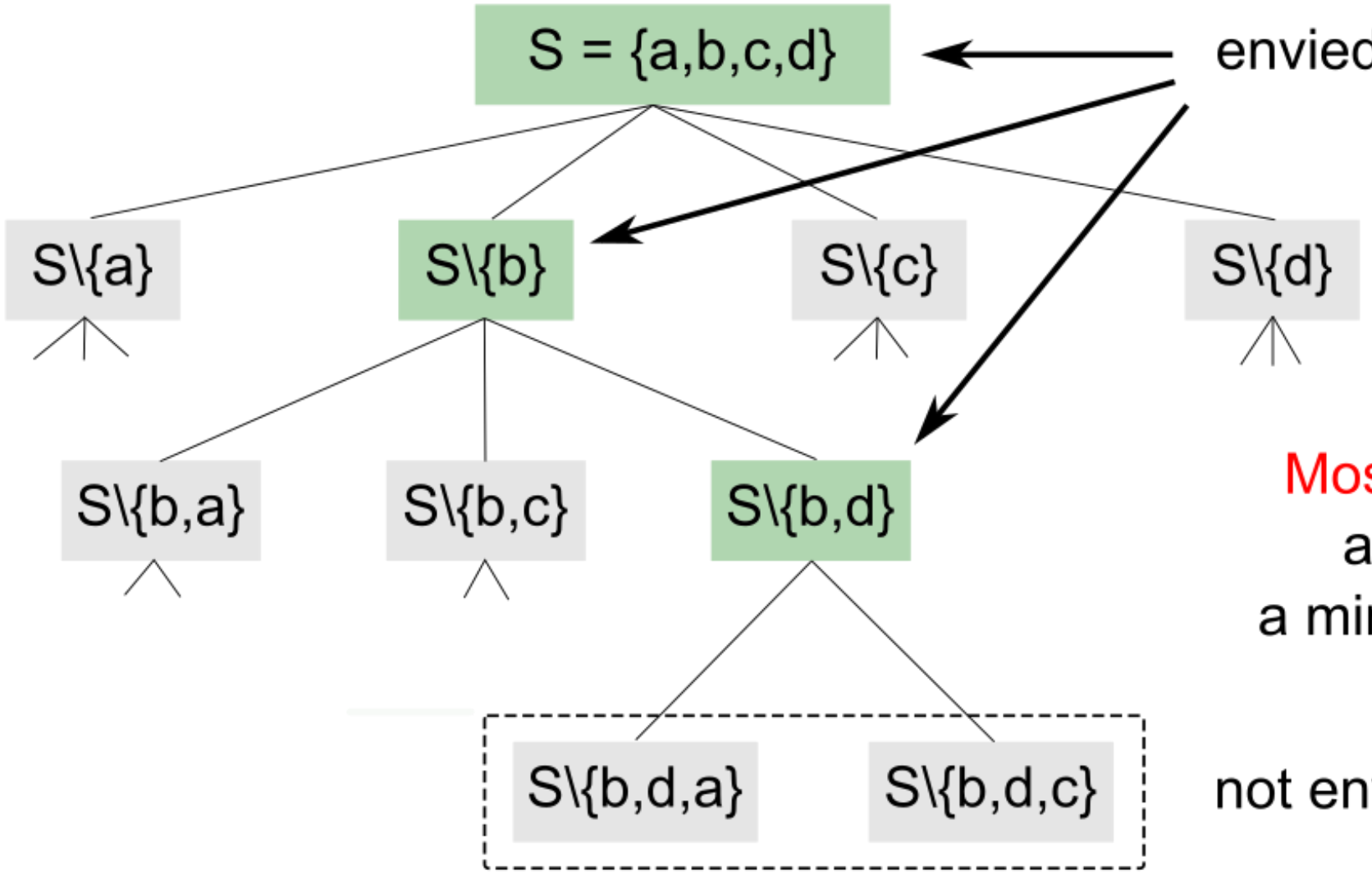


Key idea

No agent has "EFX envy" towards a minimal envied subset

Minimal Envied Subset

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]



envied by one or more agents

Most envious agent of S =
any agent that envies
a minimal envied subset of S

not envied by anyone

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

A partition $(A_1, \ldots, A_n, P)$ into $n+1$ bundles satisfies EFX-with-charity if

- the partial allocation $(A_1, \ldots, A_n)$ is EFX,
- no agent envies the pool P of unallocated items, and
- $|P| < n$.

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

A partition $(A_1, \ldots, A_n, P)$ into $n+1$ bundles satisfies EFX-with-charity if

- the partial allocation $(A_1, \ldots, A_n)$ is EFX, ← always maintain this
- no agent envies the pool P of unallocated items, and
- $|P| < n$.

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

A partition $(A_1, \ldots, A_n, P)$ into $n+1$ bundles satisfies EFX-with-charity if

- the partial allocation $(A_1, \ldots, A_n)$ is EFX, ← always maintain this
- no agent envies the pool P of unallocated items, and
- $|P| < n$.

make progress towards these

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

Trivially EFX

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

Trivially EFX

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

Trivially EFX

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Maintain EFX

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

Trivially EFX

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Maintain EFX

Otherwise, return the current allocation.

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P). ← Trivially EFX

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

← Maintain EFX

Otherwise, return the current allocation. ← EFX-with-charity

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Otherwise, return the current allocation.

Achieving EFX-with-charity

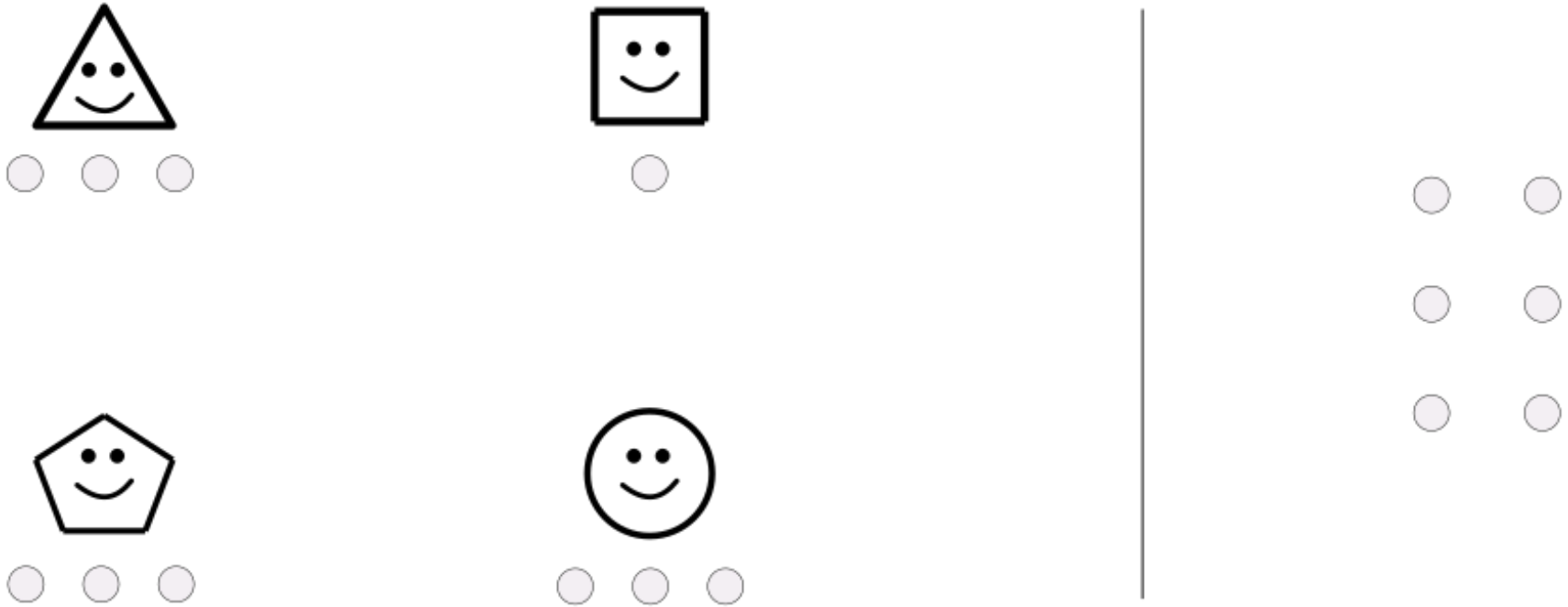
[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 1: If an unallocated good can be given to an agent while maintaining EFX, then do so.

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

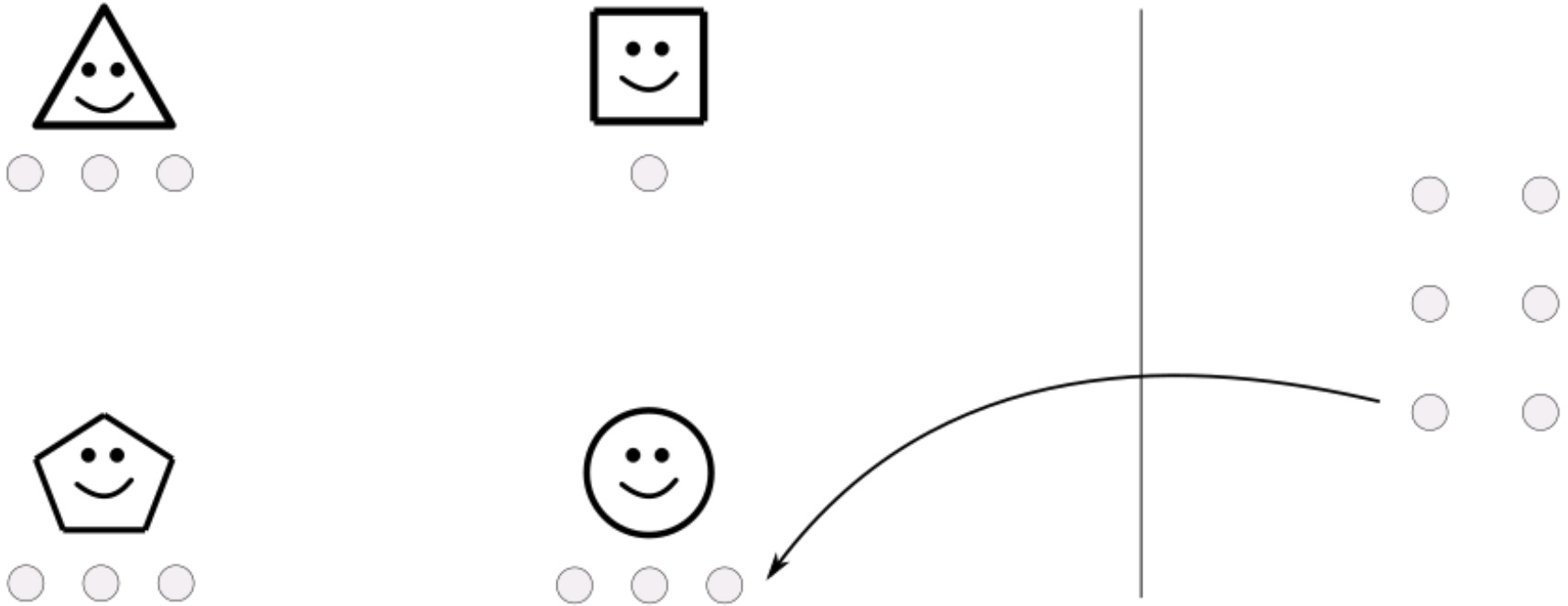
Rule 1: If an unallocated good can be given to an agent while maintaining EFX, then do so.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

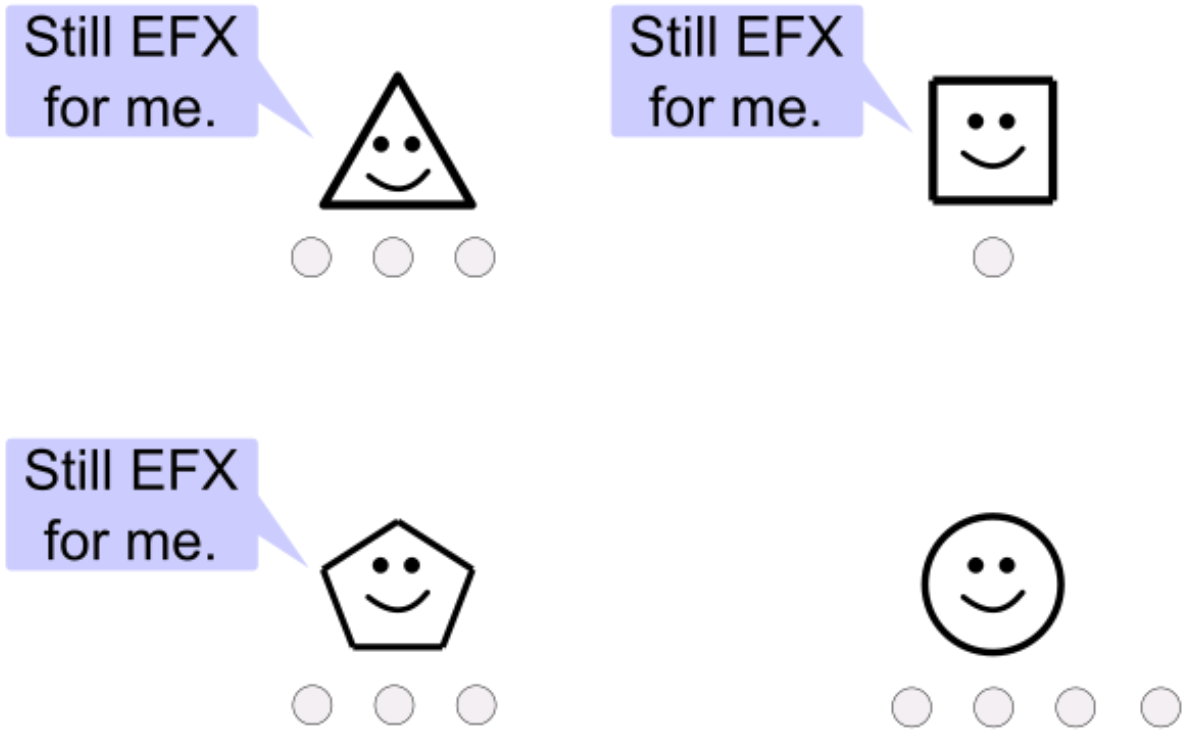
Rule 1: If an unallocated good can be given to an agent while maintaining EFX, then do so.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 1: If an unallocated good can be given to an agent while maintaining EFX, then do so.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Otherwise, return the current allocation.

Achieving EFX-with-charity

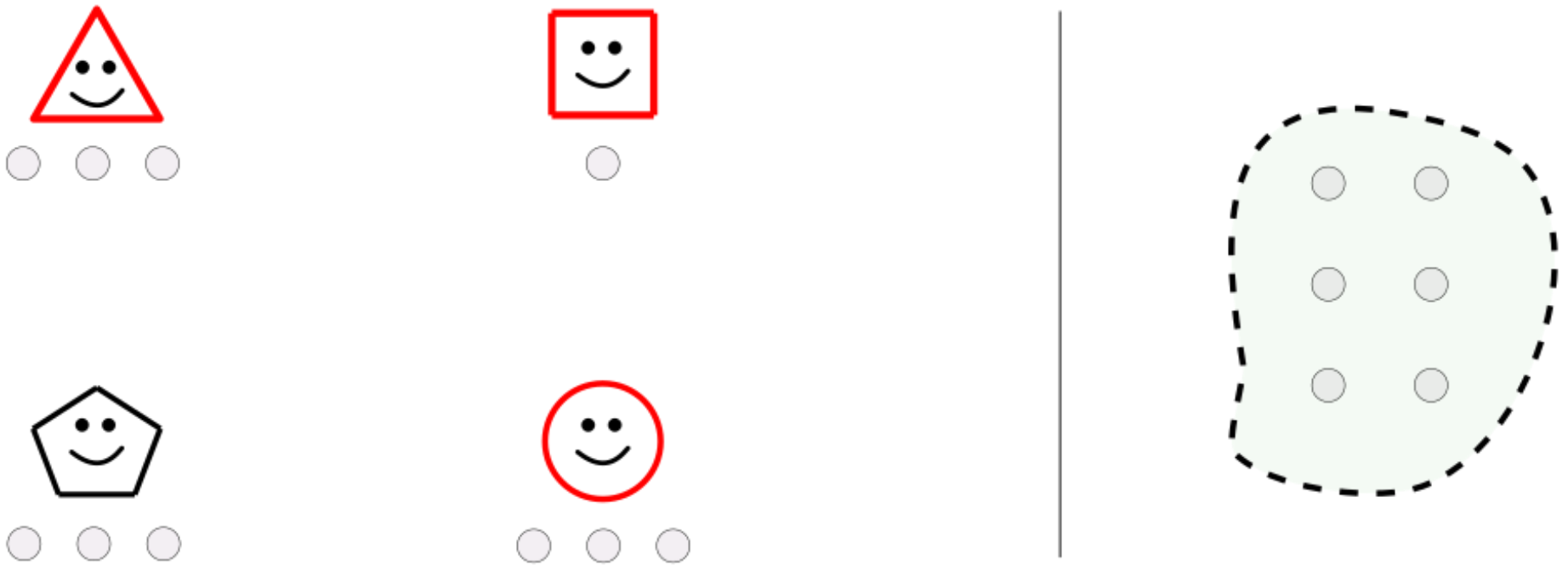
[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

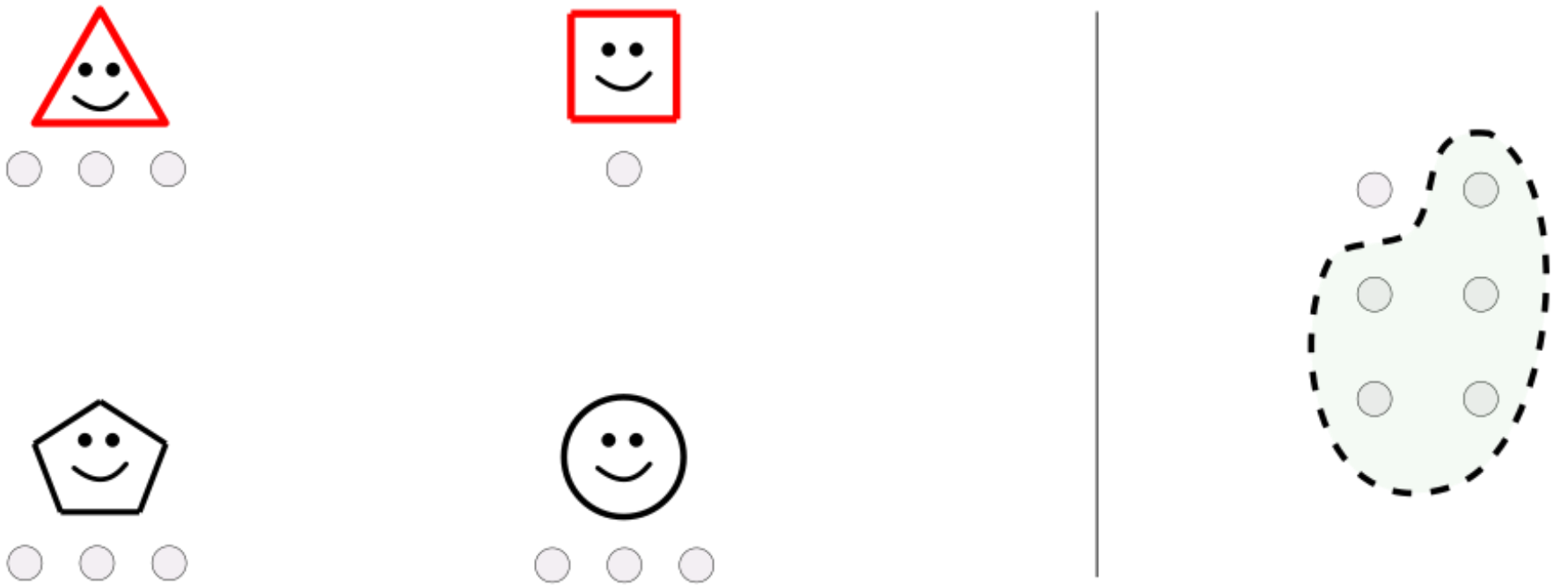
Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

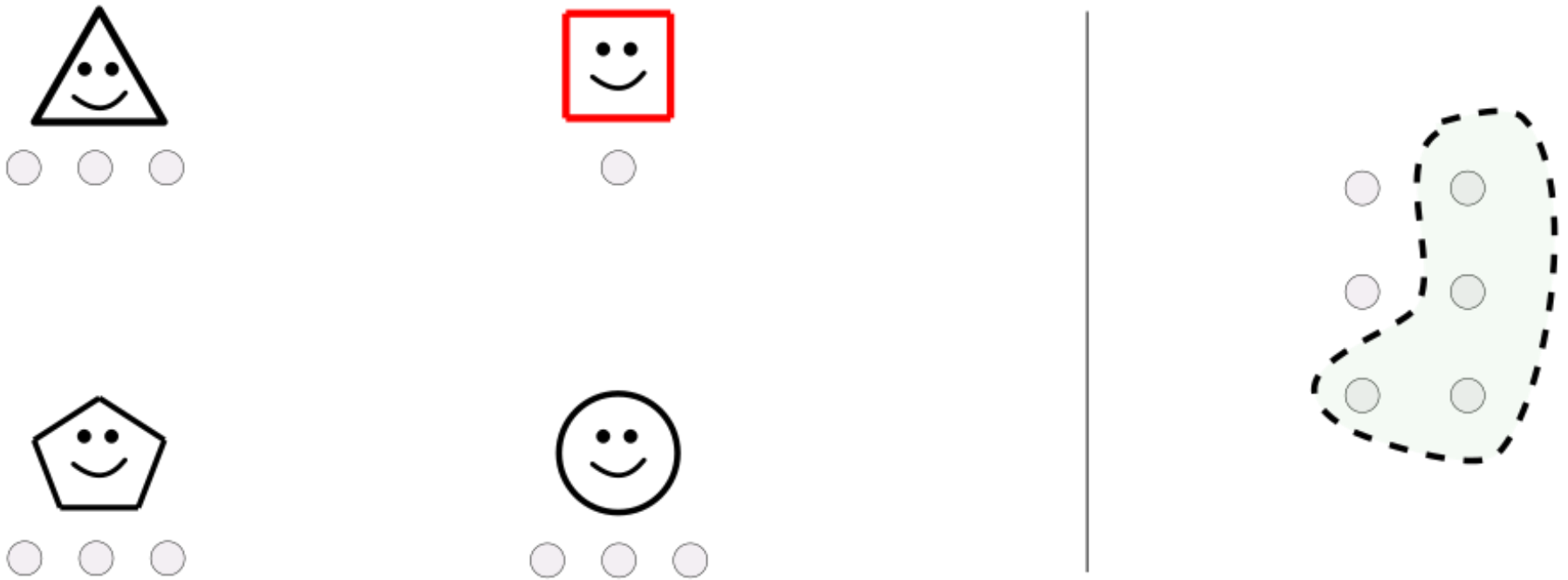
Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

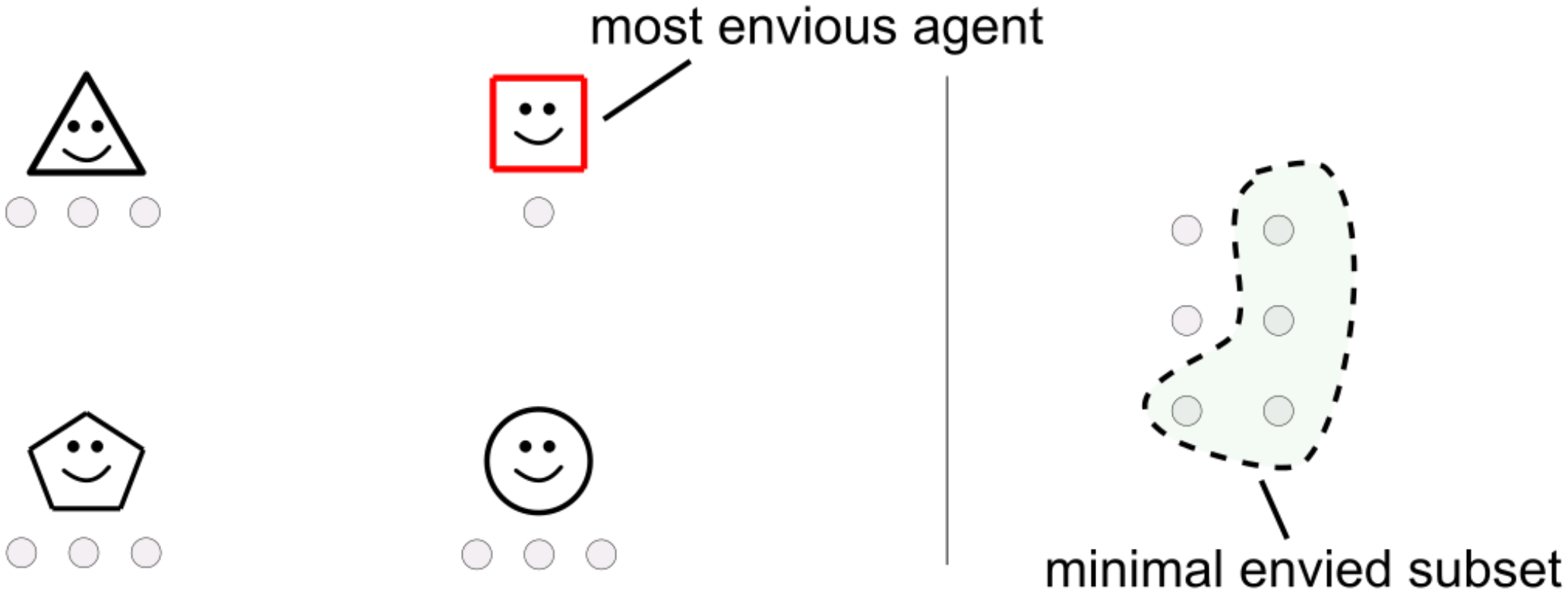
Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

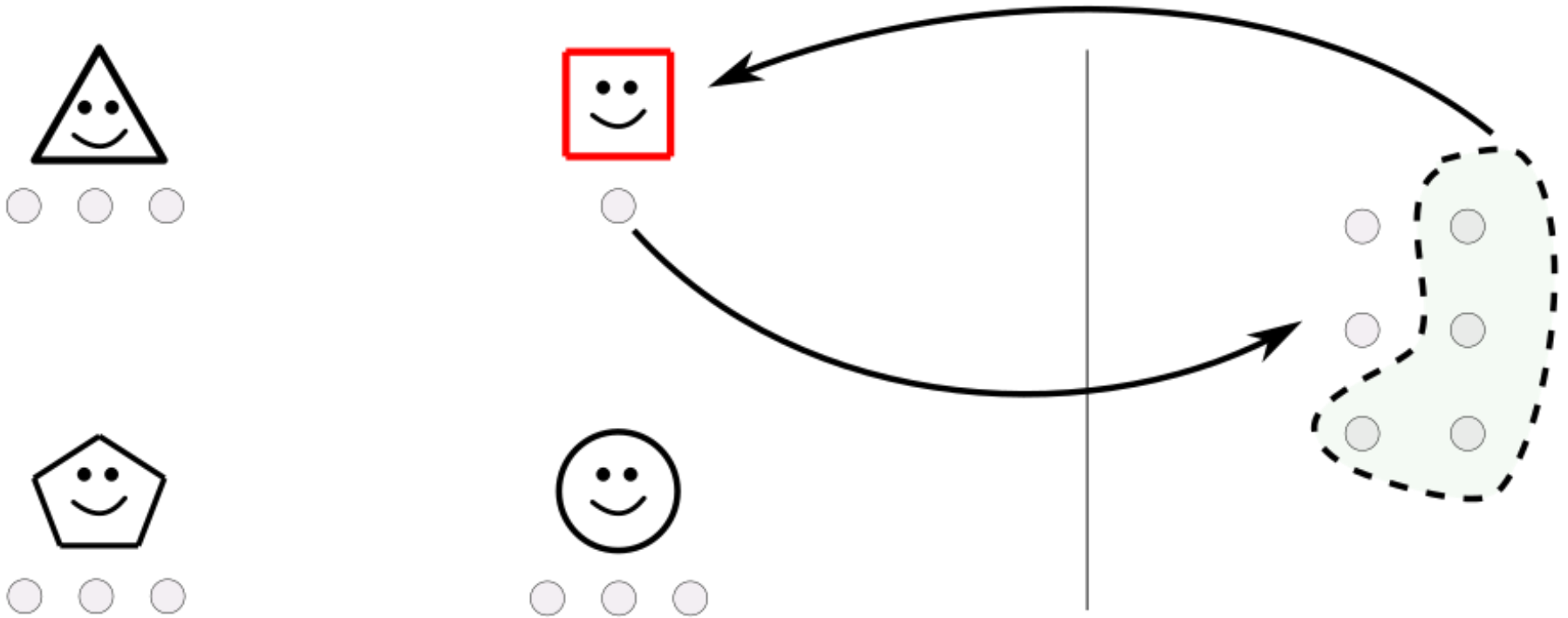
Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

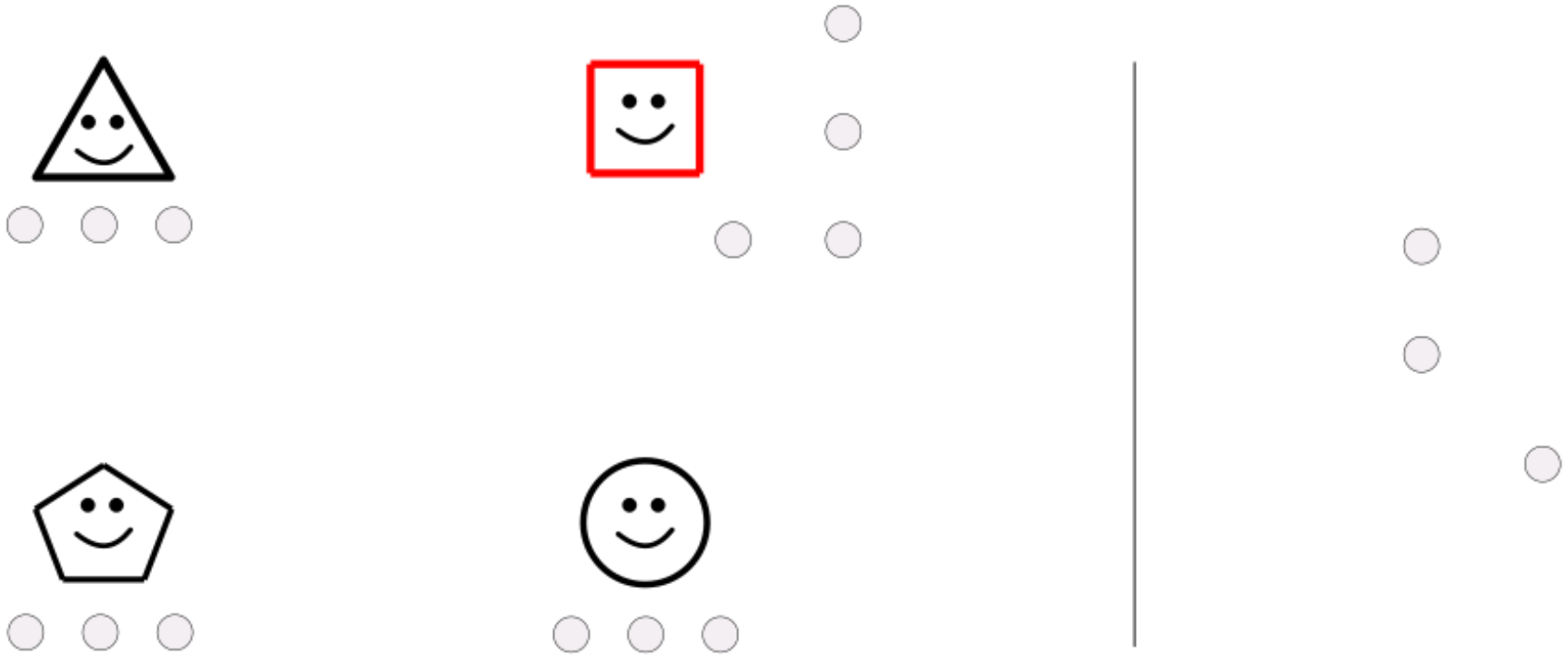
Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Otherwise, return the current allocation.

Achieving EFX-with-charity

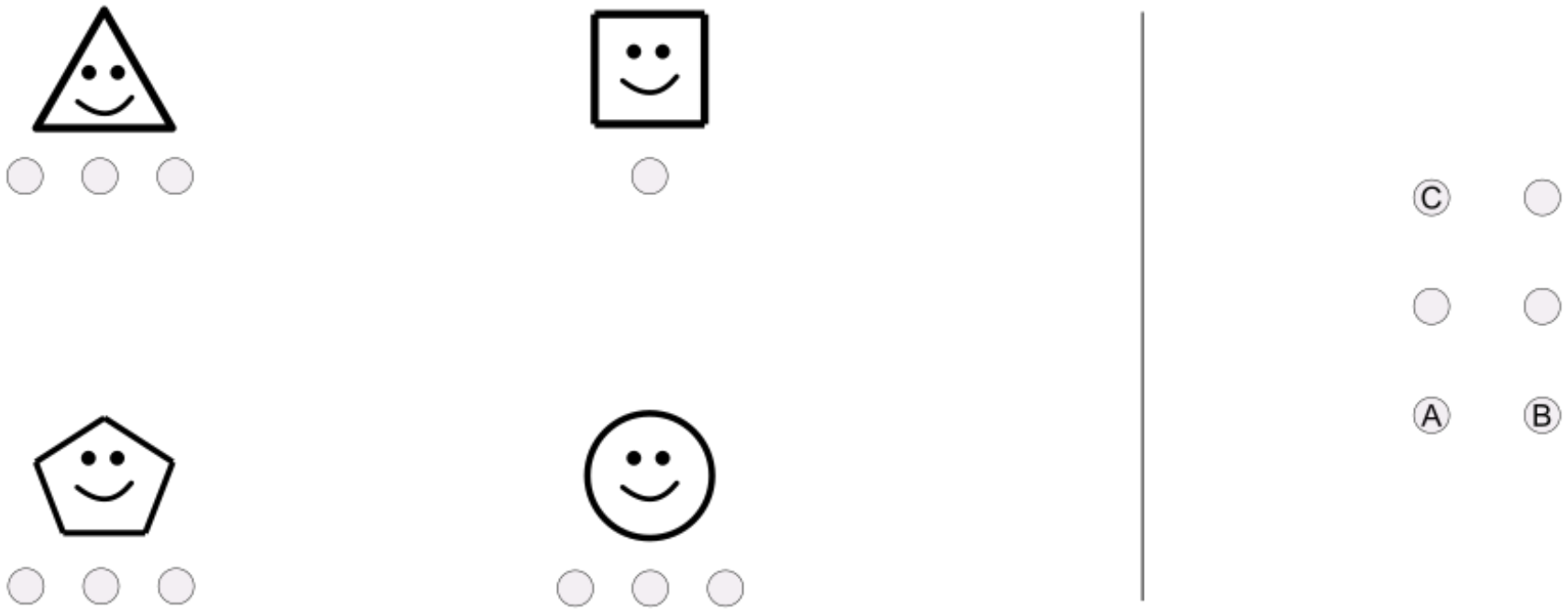
[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 3: If $|P| \geq n$, then resolve a cycle of most envious agents.

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

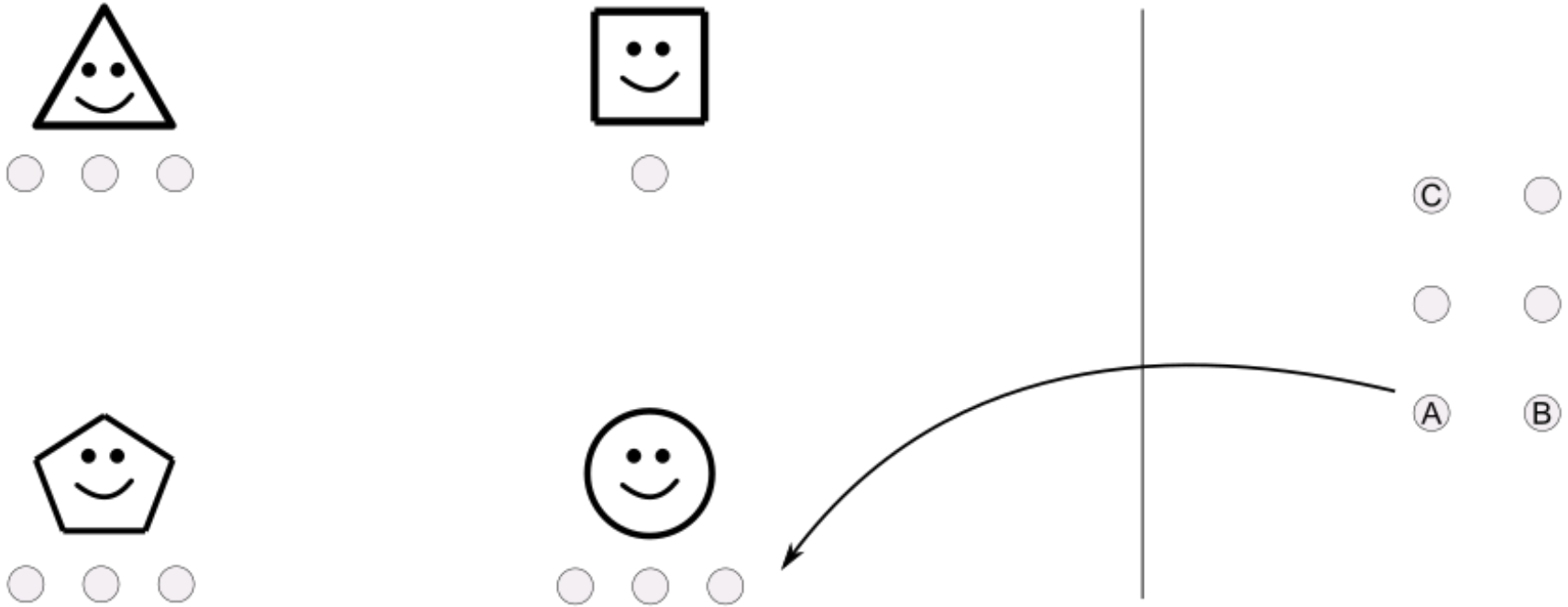
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

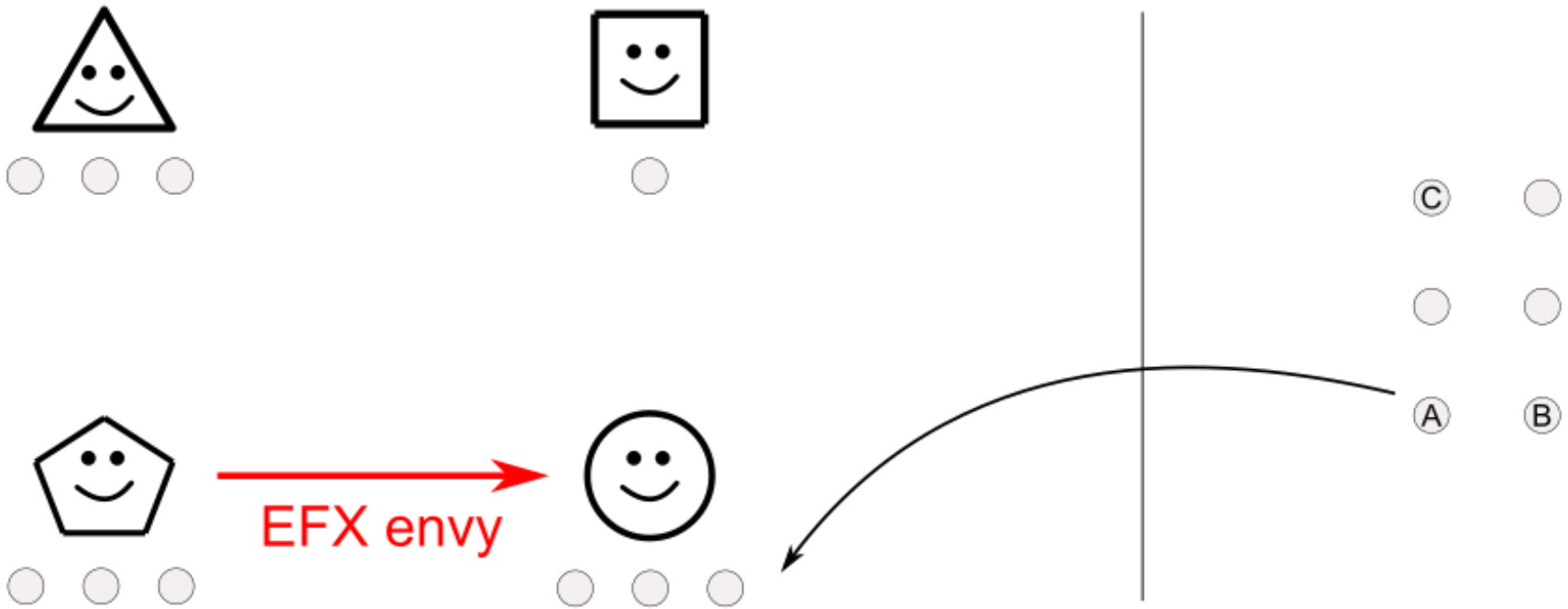
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

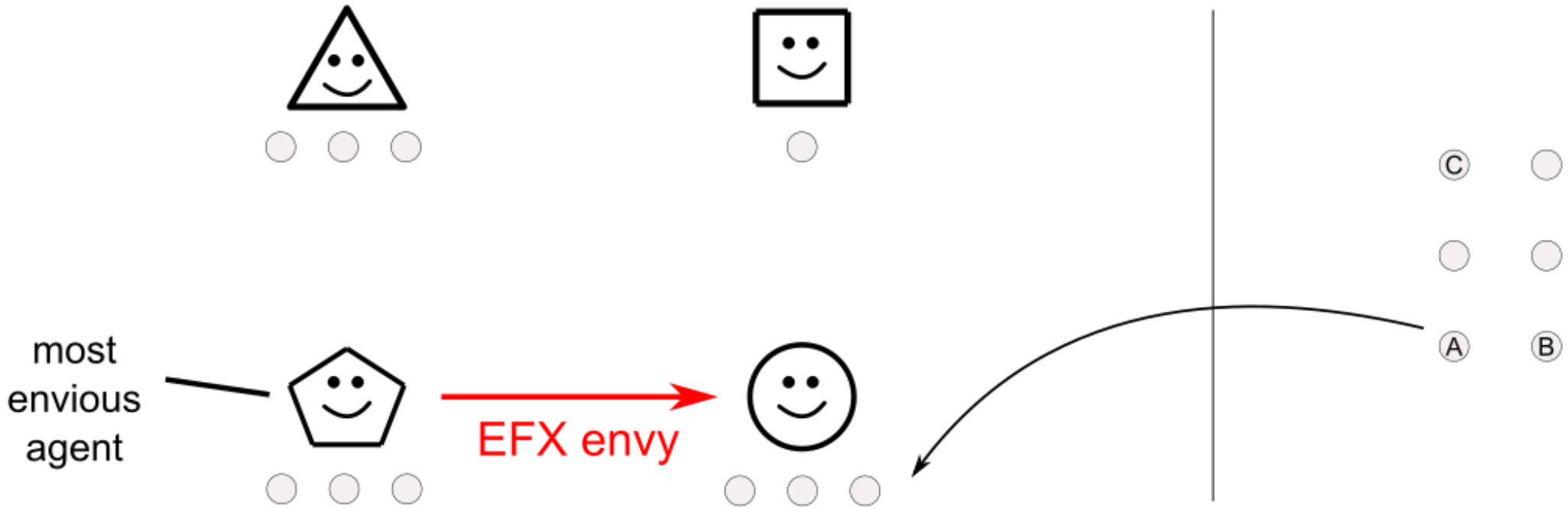
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

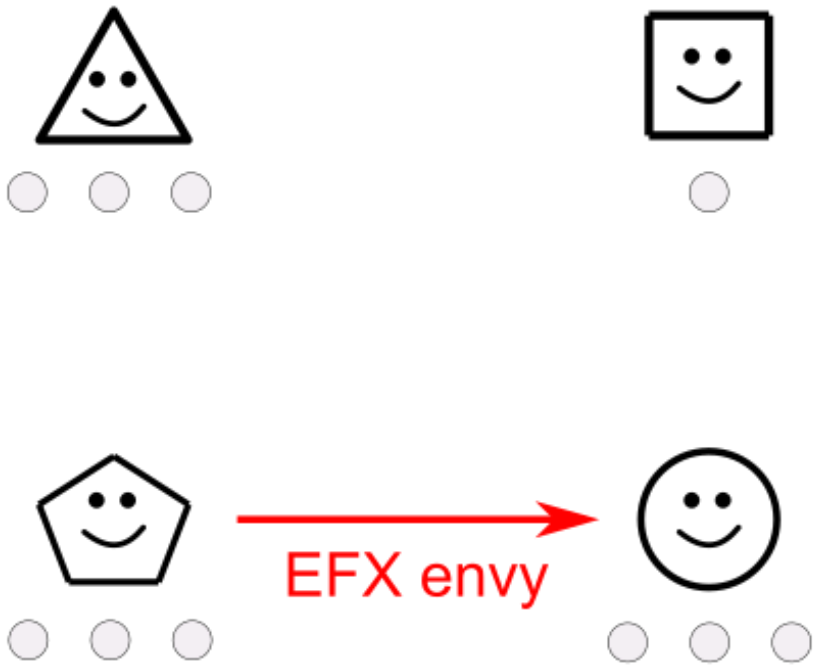
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

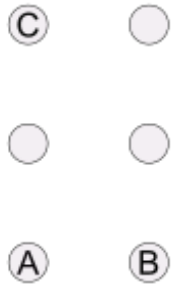


Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

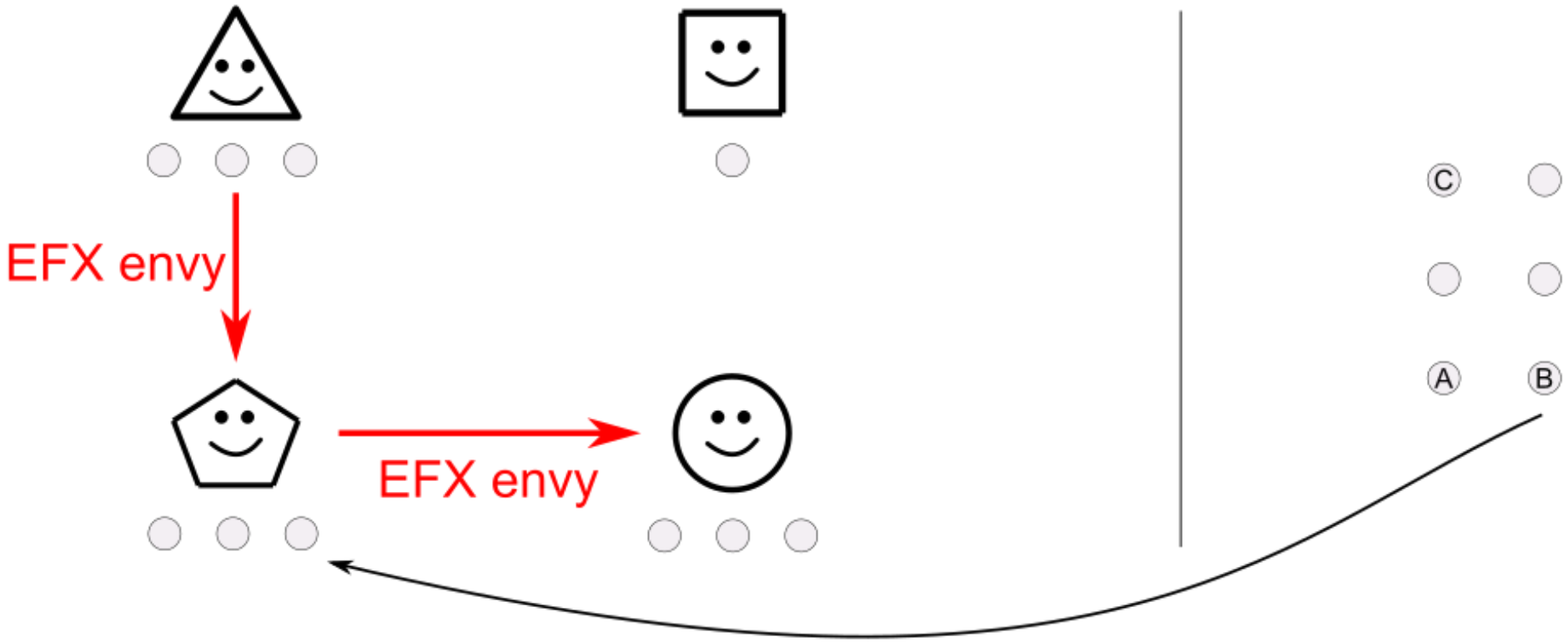




Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

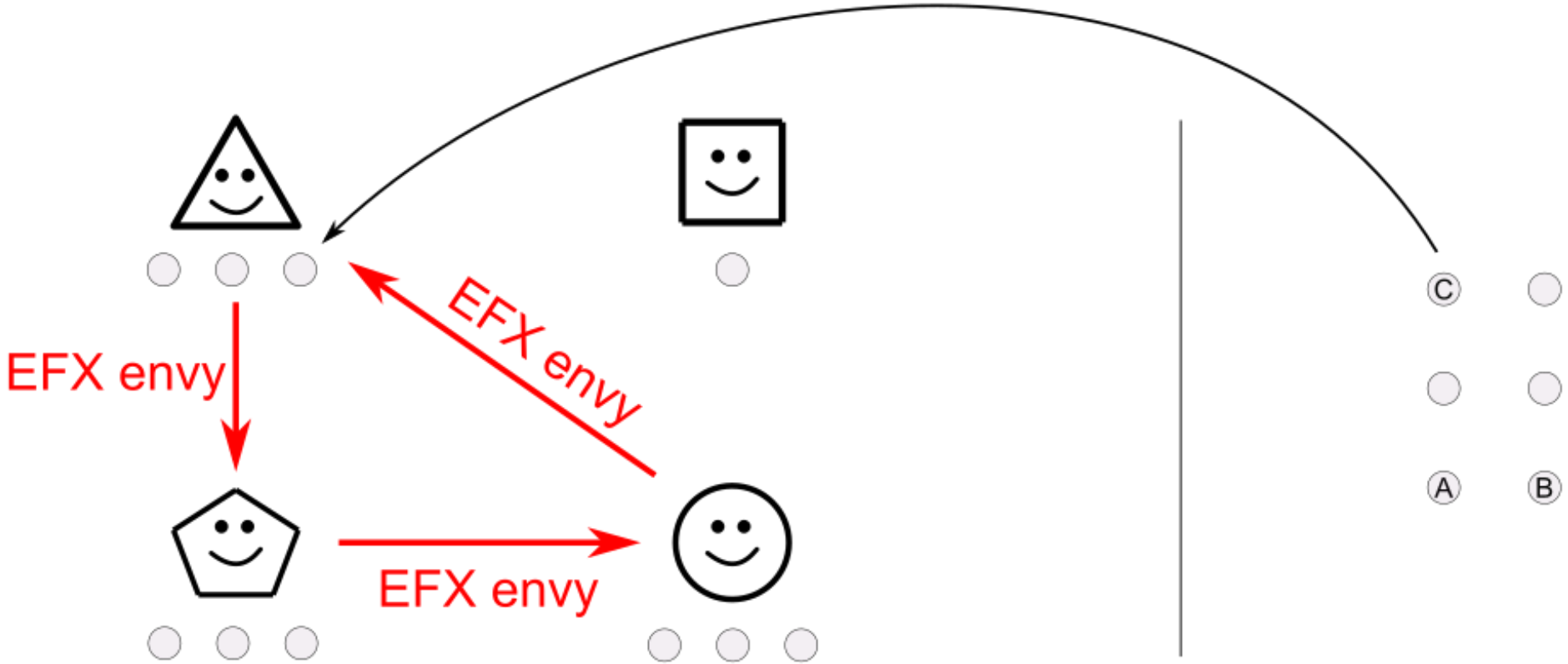
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

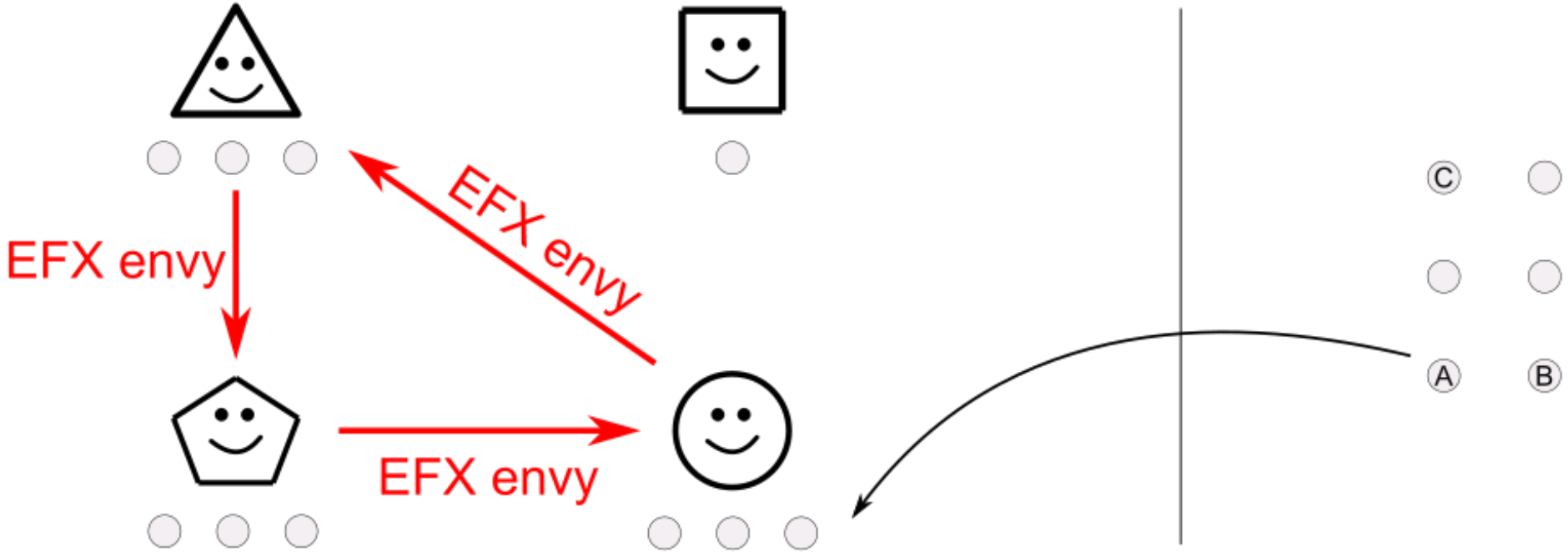
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

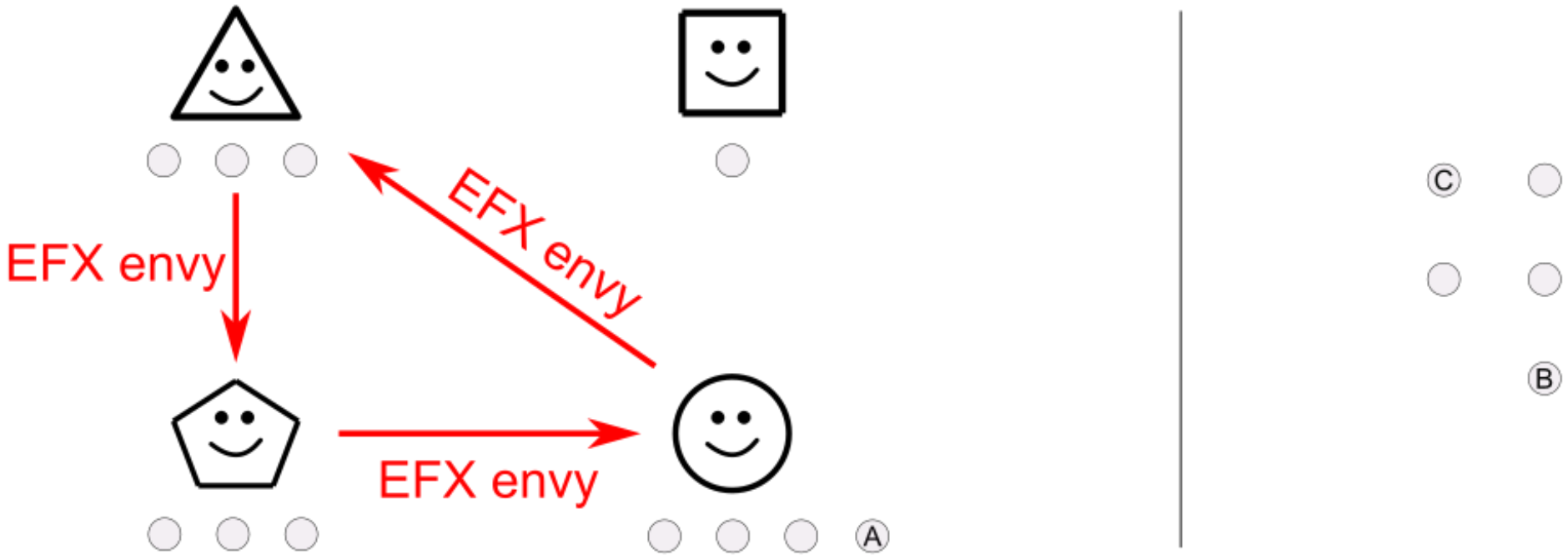
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

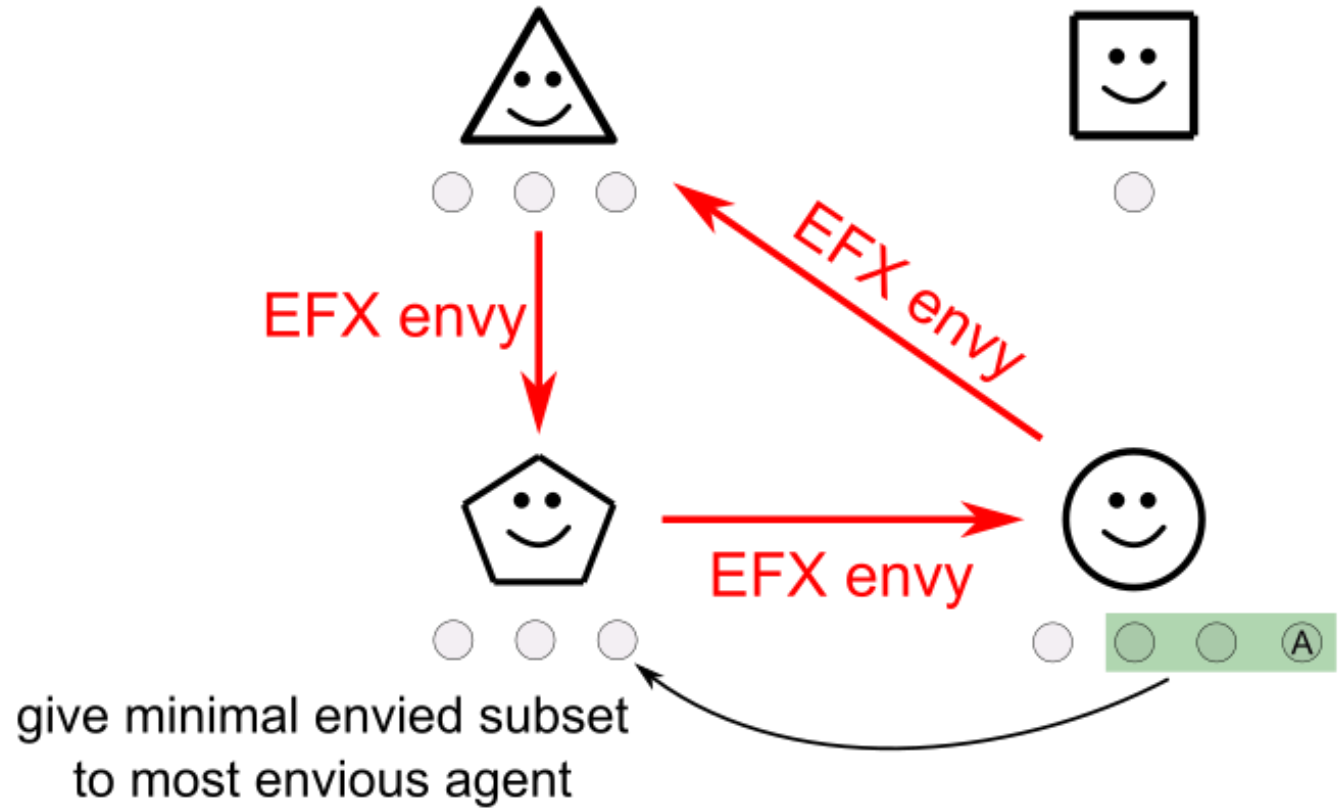
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

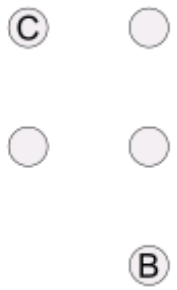


Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

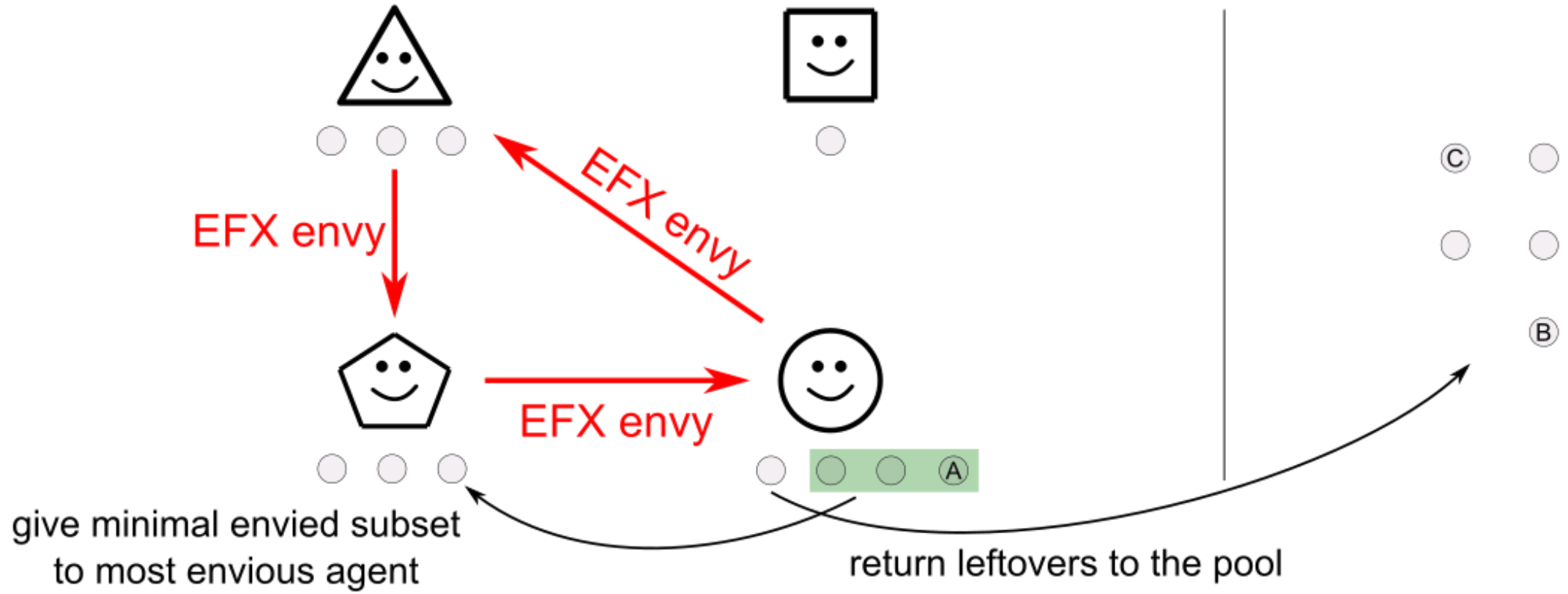




Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

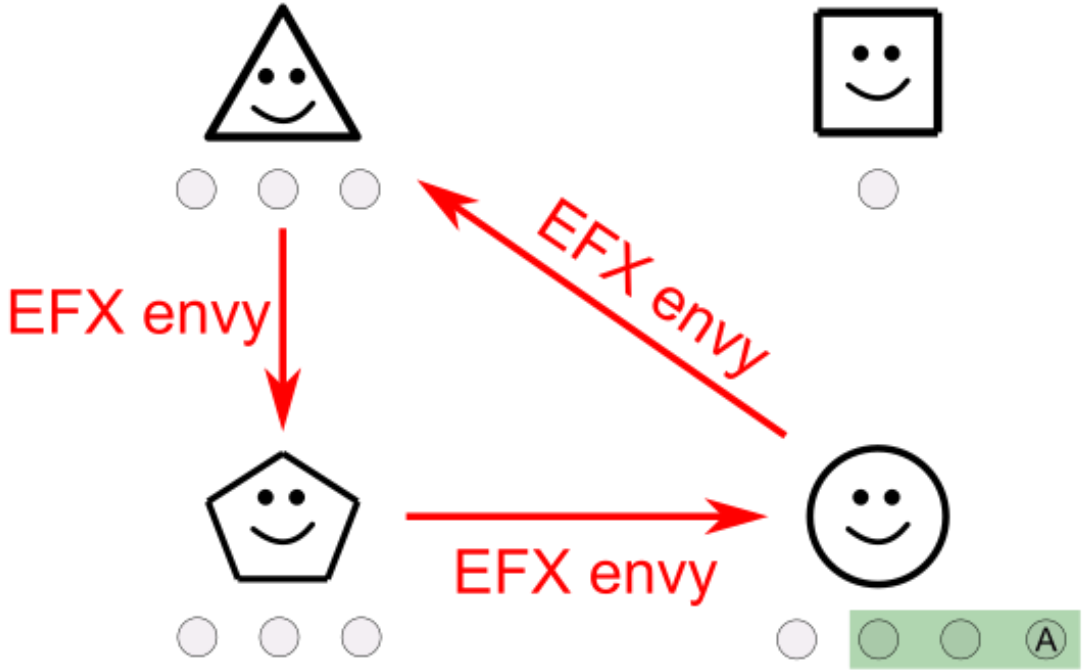
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

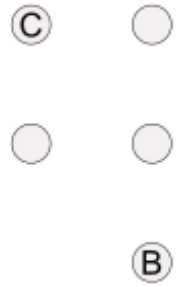


Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

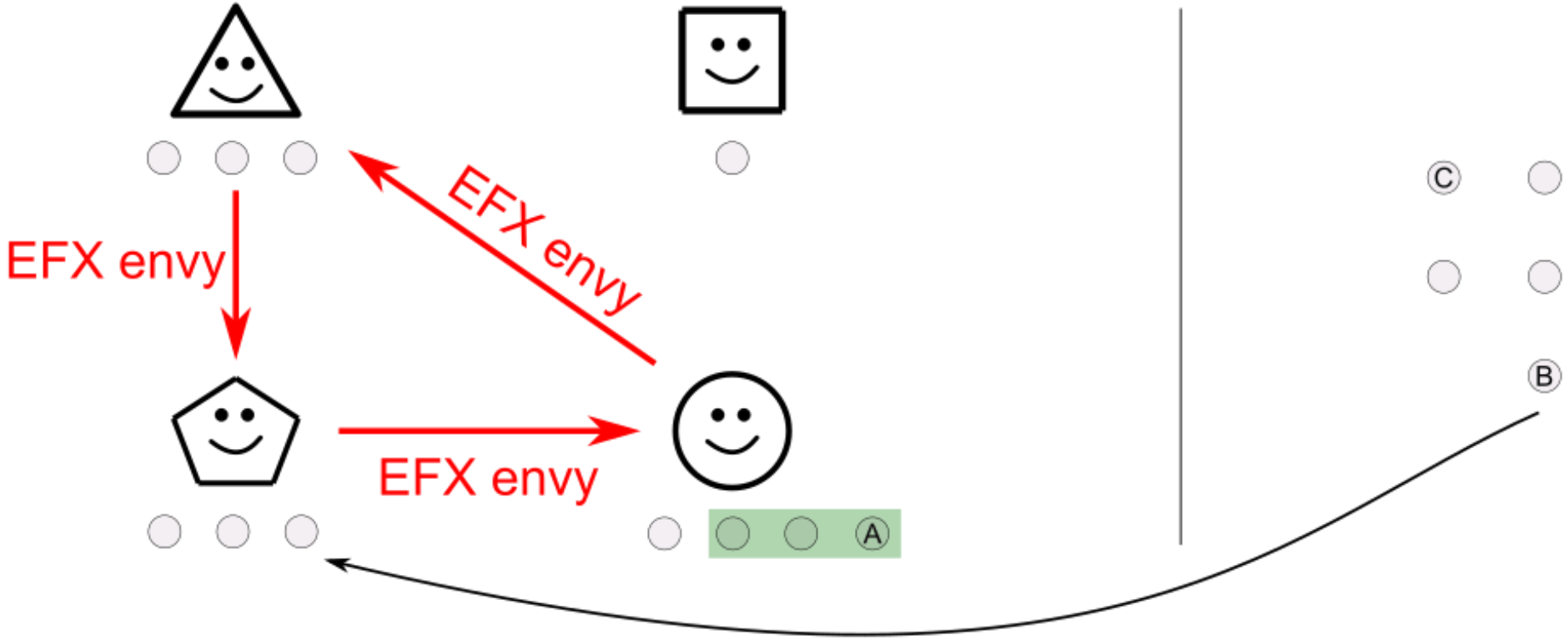




Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

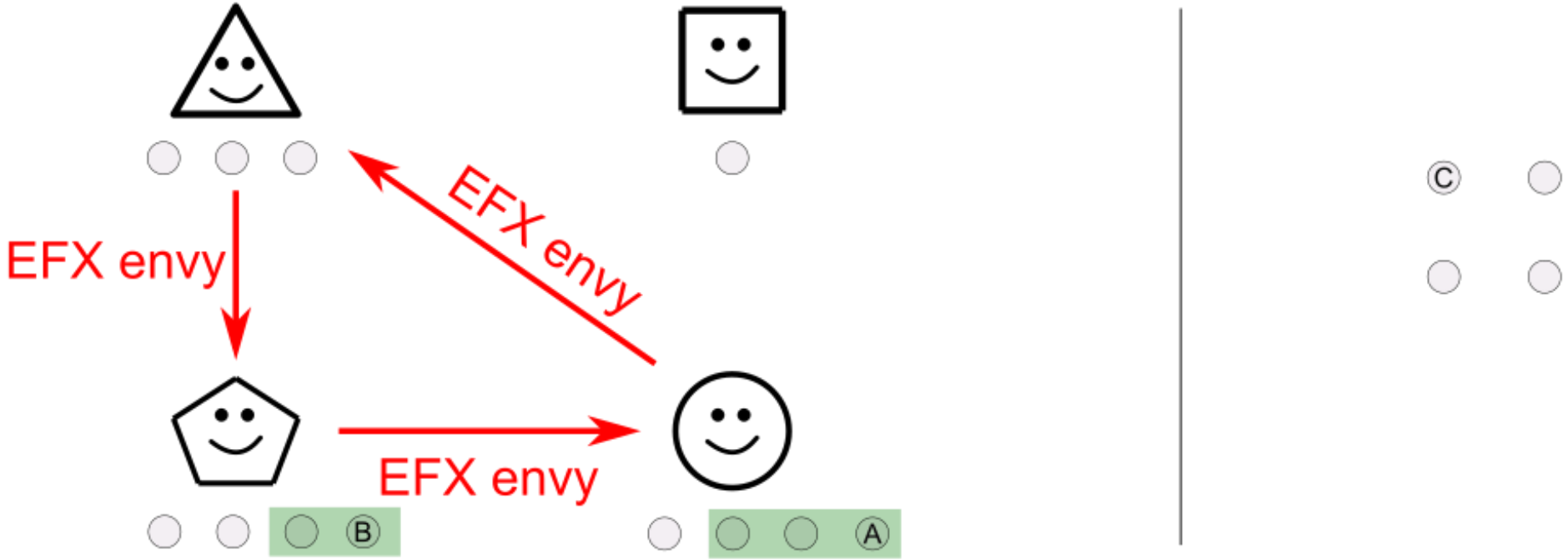
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

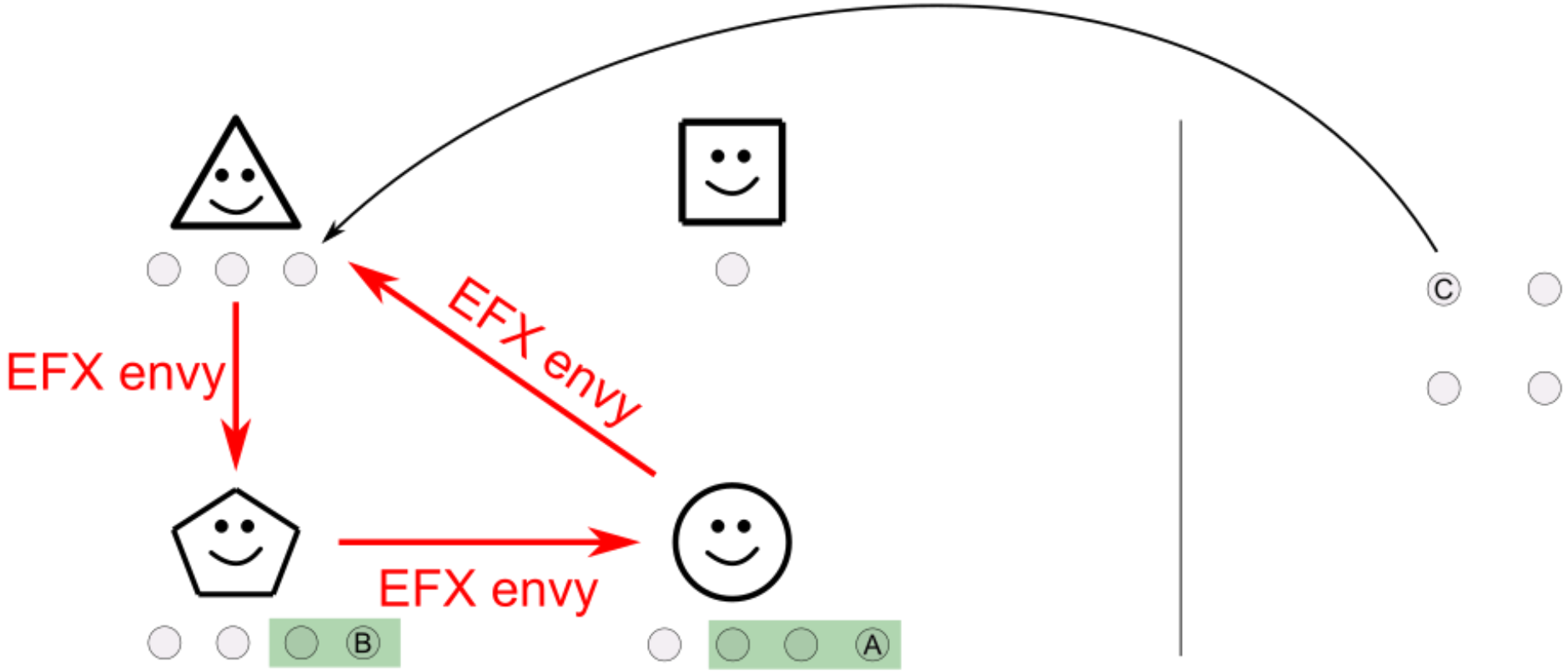
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

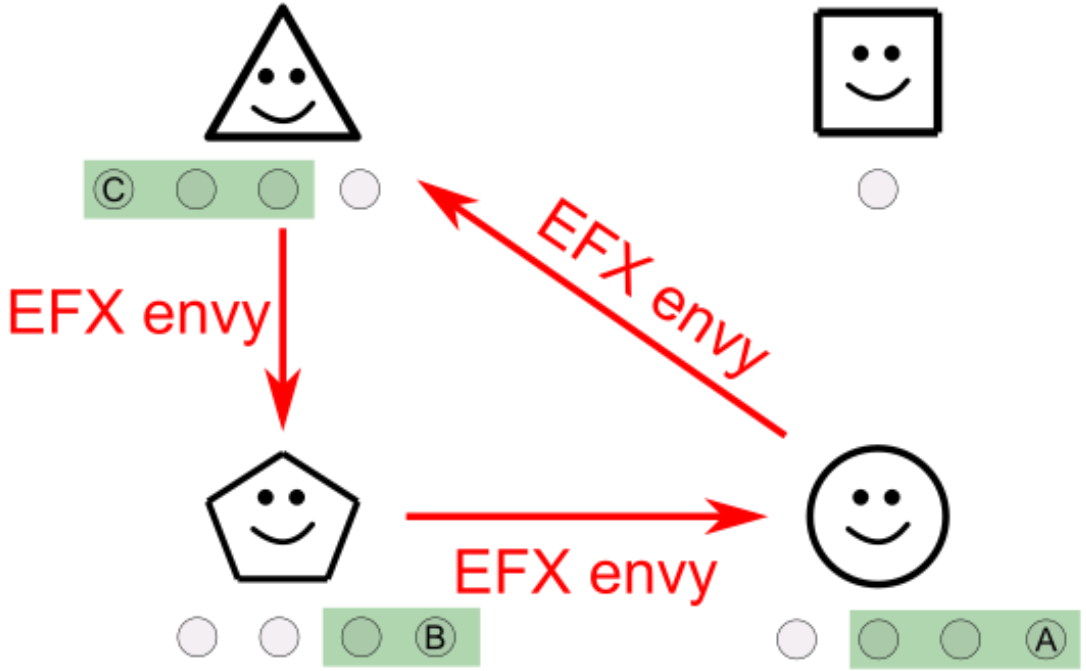
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

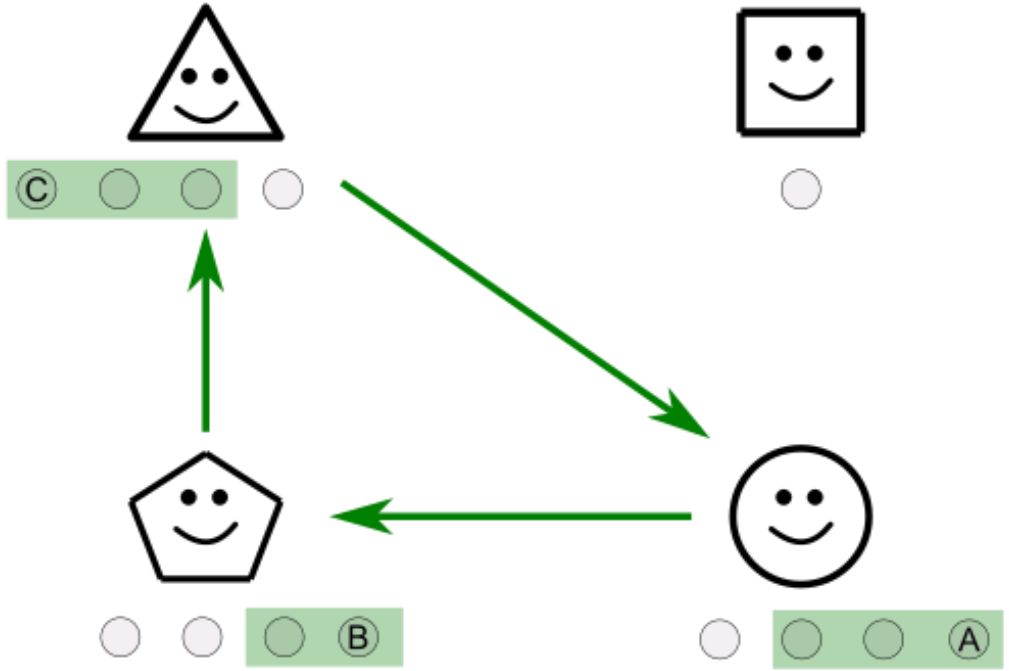
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

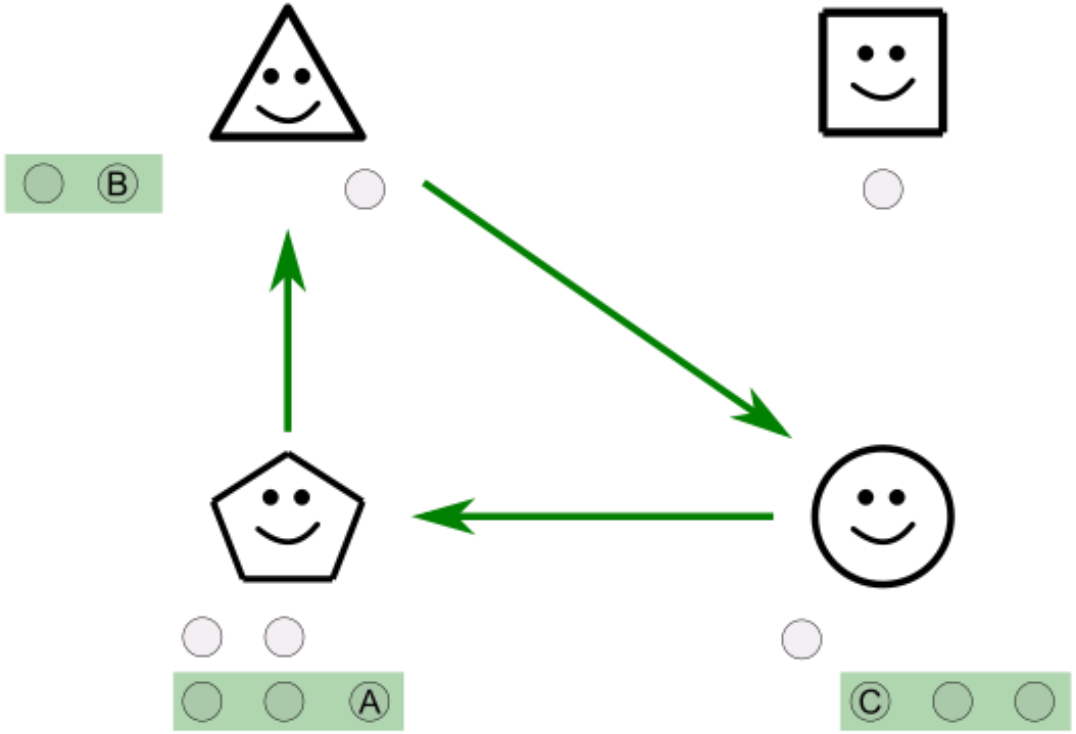
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

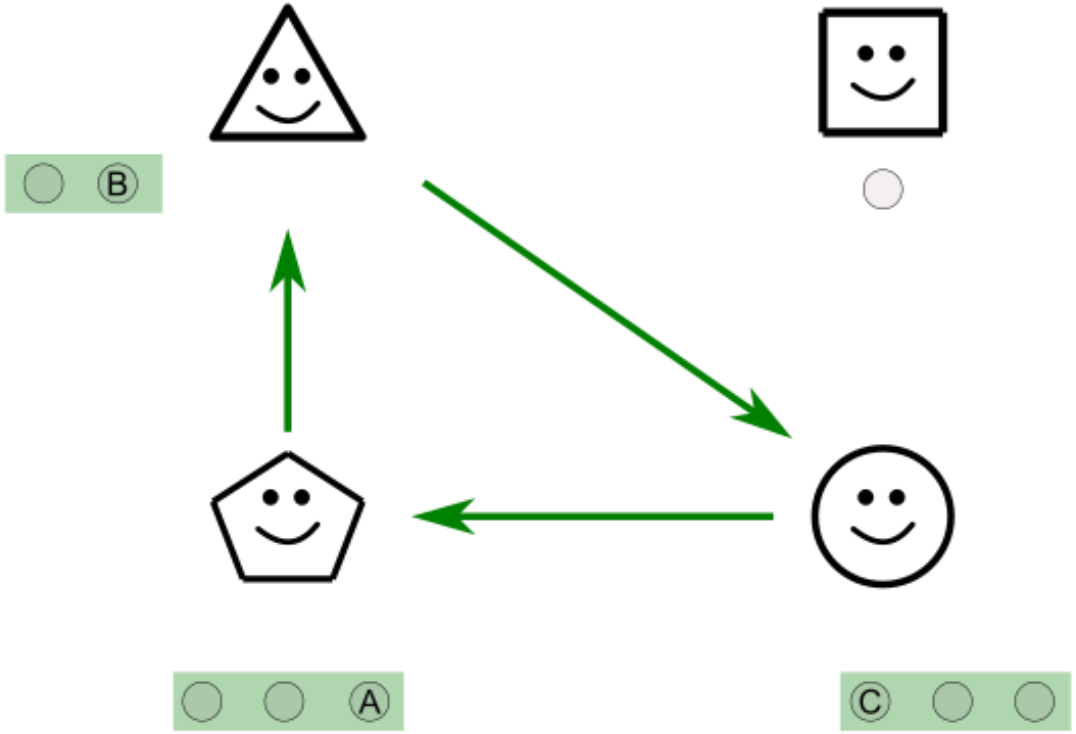
Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

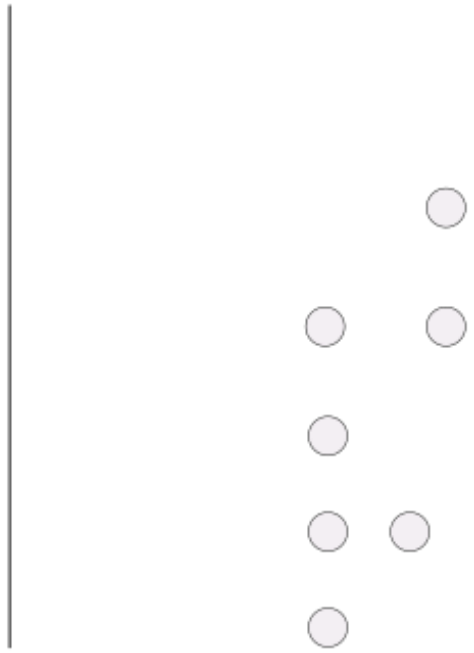


Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.

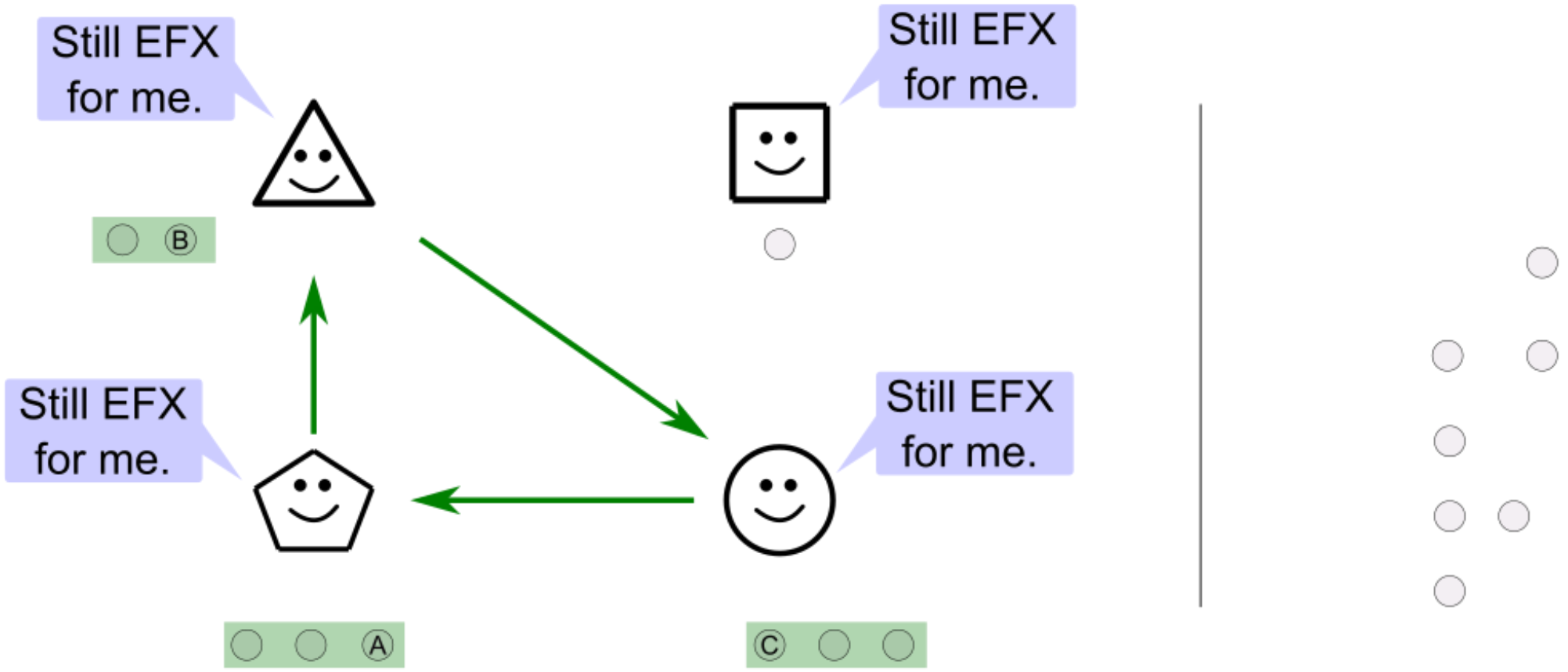




Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Rule 3: If |P| ≥ n, then resolve a cycle of most envious agents.



Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Otherwise, return the current allocation.

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Start with everything unallocated (i.e., all goods in the pool P).

As long as one of the following rules is applicable, apply it.

- Rule 1
- Rule 2
- Rule 3

Otherwise, return the current allocation.

If the algorithm terminates:

- EFX among main agents (invariant)
- No envy towards pool (Rule 2)
- $|P|<n$ (Rule 3)

$\Rightarrow$ EFX-with-charity

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Why does the algorithm terminate?

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Why does the algorithm terminate?

Rule 1: If an unallocated good can be given to an agent while maintaining EFX, then do so.

Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).

Rule 3: If $|P| \geq n$, then resolve a cycle of most envious agents.

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Why does the algorithm terminate?

Rule 1: If an unallocated good can be given to an agent while maintaining EFX, then do so.

Social welfare ↕ , $|P| \downarrow$

Rule 2: If any agent envies the pool P, then give a minimal envied subset of P to the most envious agent (and return old bundle to P).

Social welfare $\uparrow$, $|P| \updownarrow$

Rule 3: If $|P| \geq n$, then resolve a cycle of most envious agents.

Social welfare $\uparrow$, $|P| \updownarrow$

Achieving EFX-with-charity

[Chaudhury, Kavitha, Mehlhorn, and Sgouritsa, *SODA* 2020, *SICOMP* 2021]

Why does the algorithm terminate?

Potential function

$$\phi(A) = m \cdot \sum_i v_i(A_i) - |P|$$

$$m = \text{no. of goods}$$

EFX-with-charity in polynomial time

Next Time

Indivisible Chores

Quiz

Quiz

Construct an instance with two agents where no EFX allocation is Pareto optimal.