COV886 Special Module in Algorithms: Computational Social Choice

Lecture 12

Fairness and Efficiency

April 02, 2022 | Rohit Vaish

Reminder about starting recording

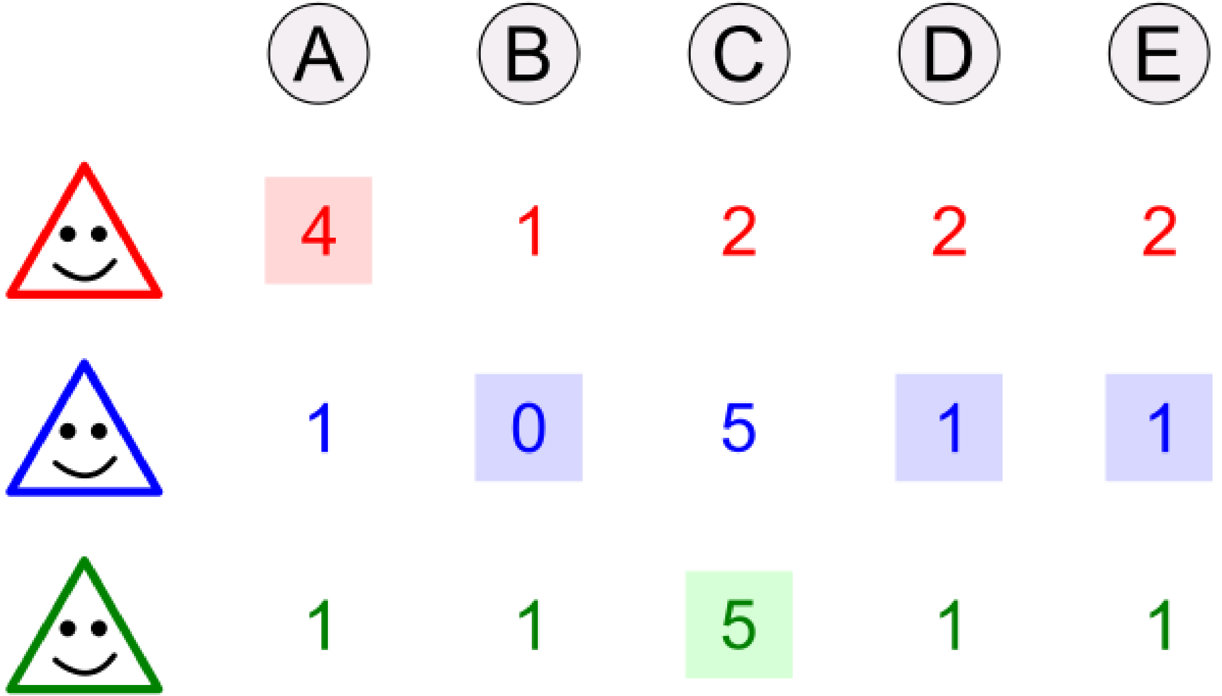
The Model

	A	B	C	D	E
	4	1	2	2	2
	1	0	5	1	1
	1	1	5	1	1

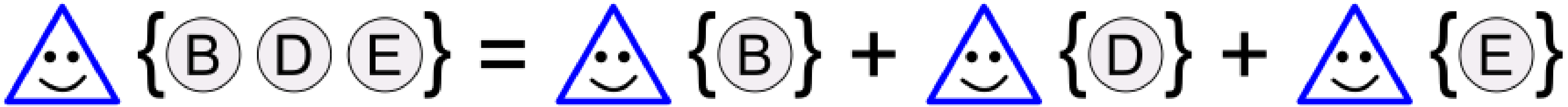
The Model

	A	B	C	D	E
	4	1	2	2	2
	1	0	5	1	1
	1	1	5	1	1

The Model



Additive valuations



$= 0+1+1 = 2$

Envy-Freeness Up To One Good [Budish, 2011]

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

	A	B	C
	4	1	2
	1	1	5

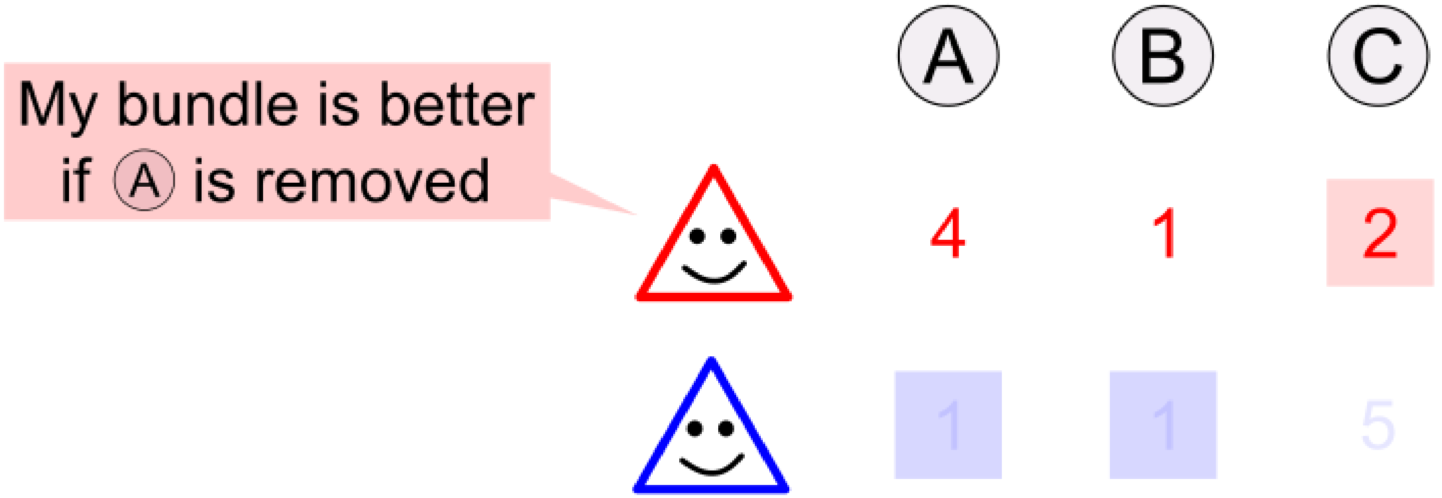
Envy-Freeness Up To One Good [Budish, 2011]

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

	A	B	C
	4	1	2
	1	1	5

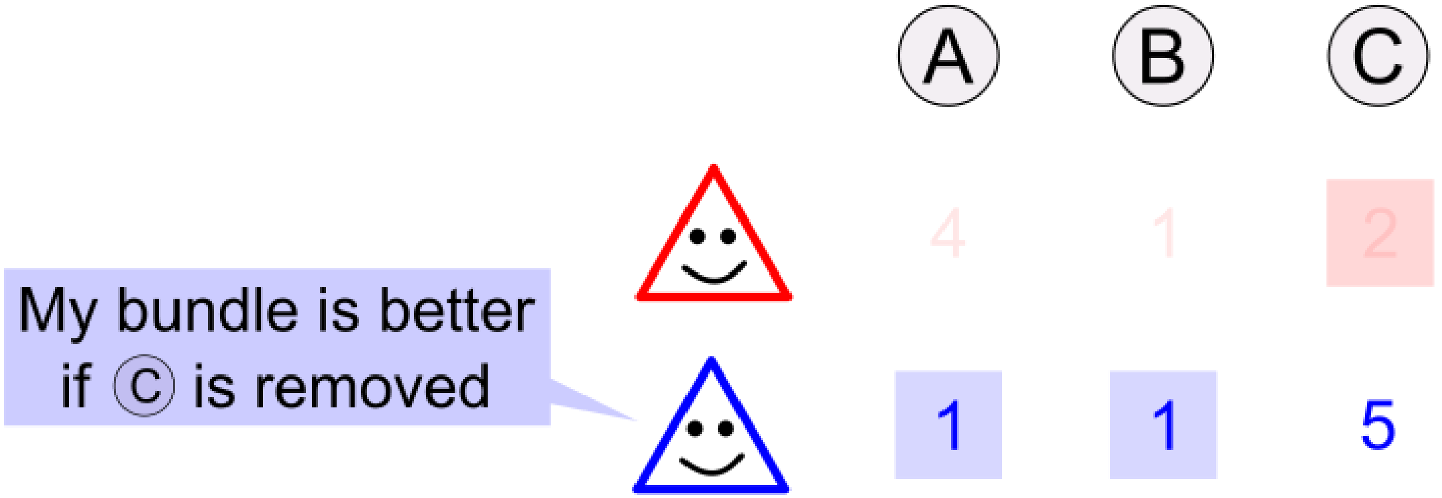
Envy-Freeness Up To One Good [Budish, 2011]

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



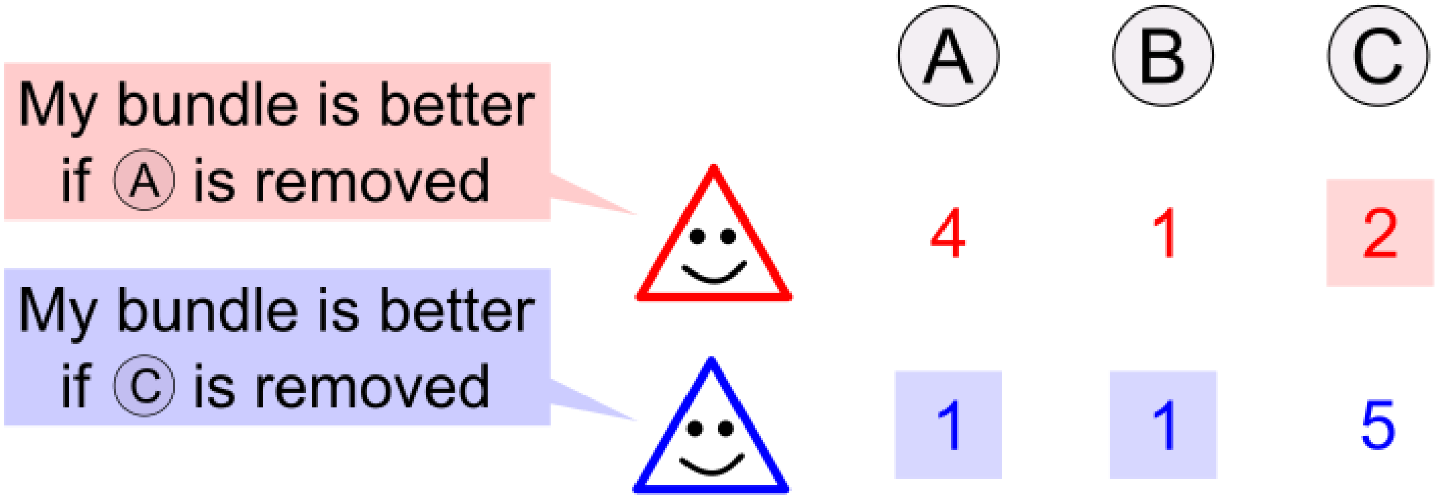
Envy-Freeness Up To One Good [Budish, 2011]

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



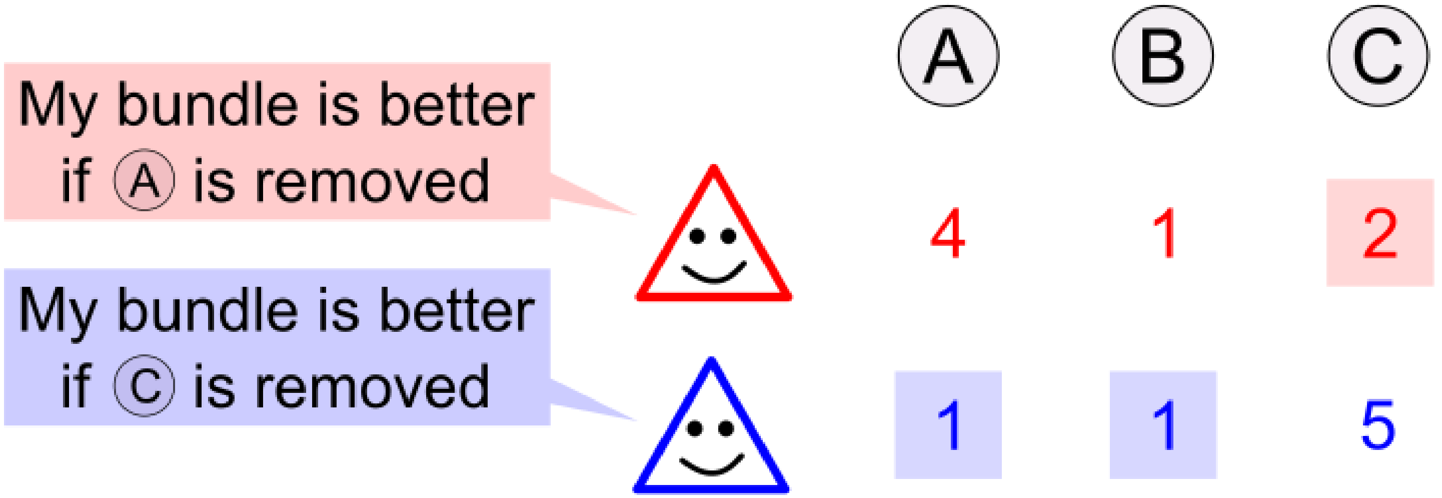
Envy-Freeness Up To One Good [Budish, 2011]

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



Envy-Freeness Up To One Good [Budish, 2011]

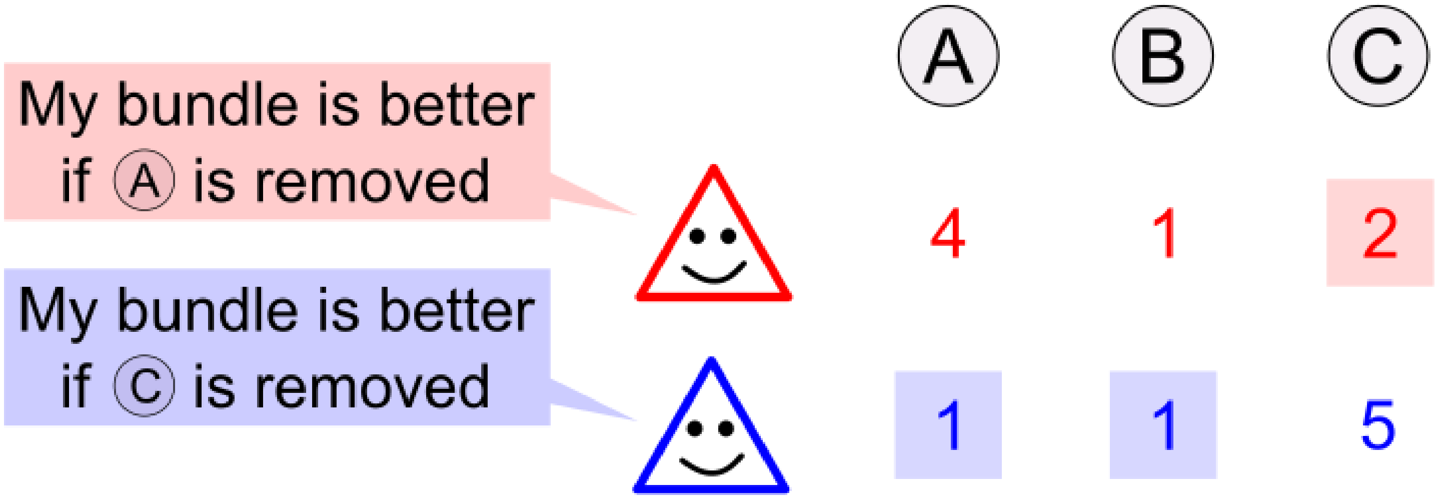
Envy can be eliminated by removing some good in the envied bundle.



Allocation $A = (A_1, \ldots, A_n)$ is EF1 if for every pair of agents i, k, there exists a good $j \in A_k$ such that $v_i(A_i) \geq v_i(A_k \setminus \{j\})$.

Envy-Freeness Up To One Good [Budish, 2011]

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

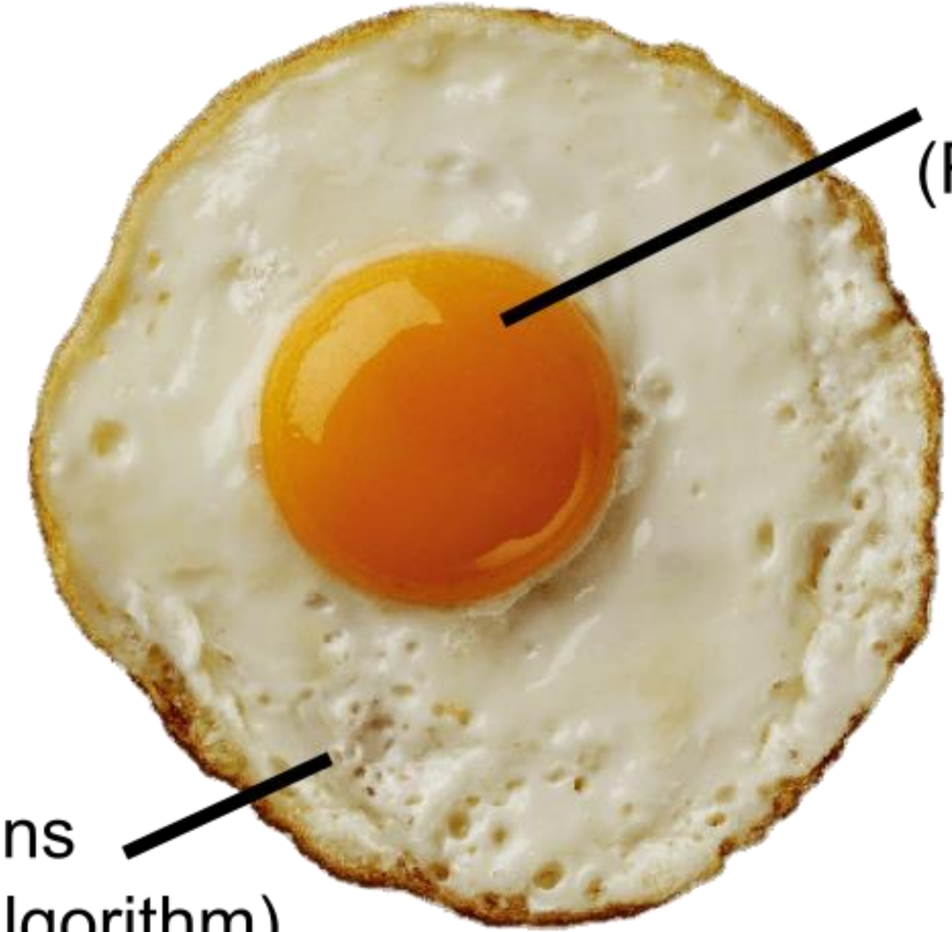


Allocation $A = (A_1, \ldots, A_n)$ is EF1 if for every pair of agents i, k, there exists a good $j \in A_k$ such that $v_i(A_i) \geq v_i(A_k \setminus \{j\})$.

Guaranteed to exist and efficiently computable

Last Time

Algorithms for finding an EF1 allocation



Additive valuations
(Round-robin algorithm)

Monotone valuations
(Envy-cycle elimination algorithm)

WHEN A COMPLETE ALLOCATION

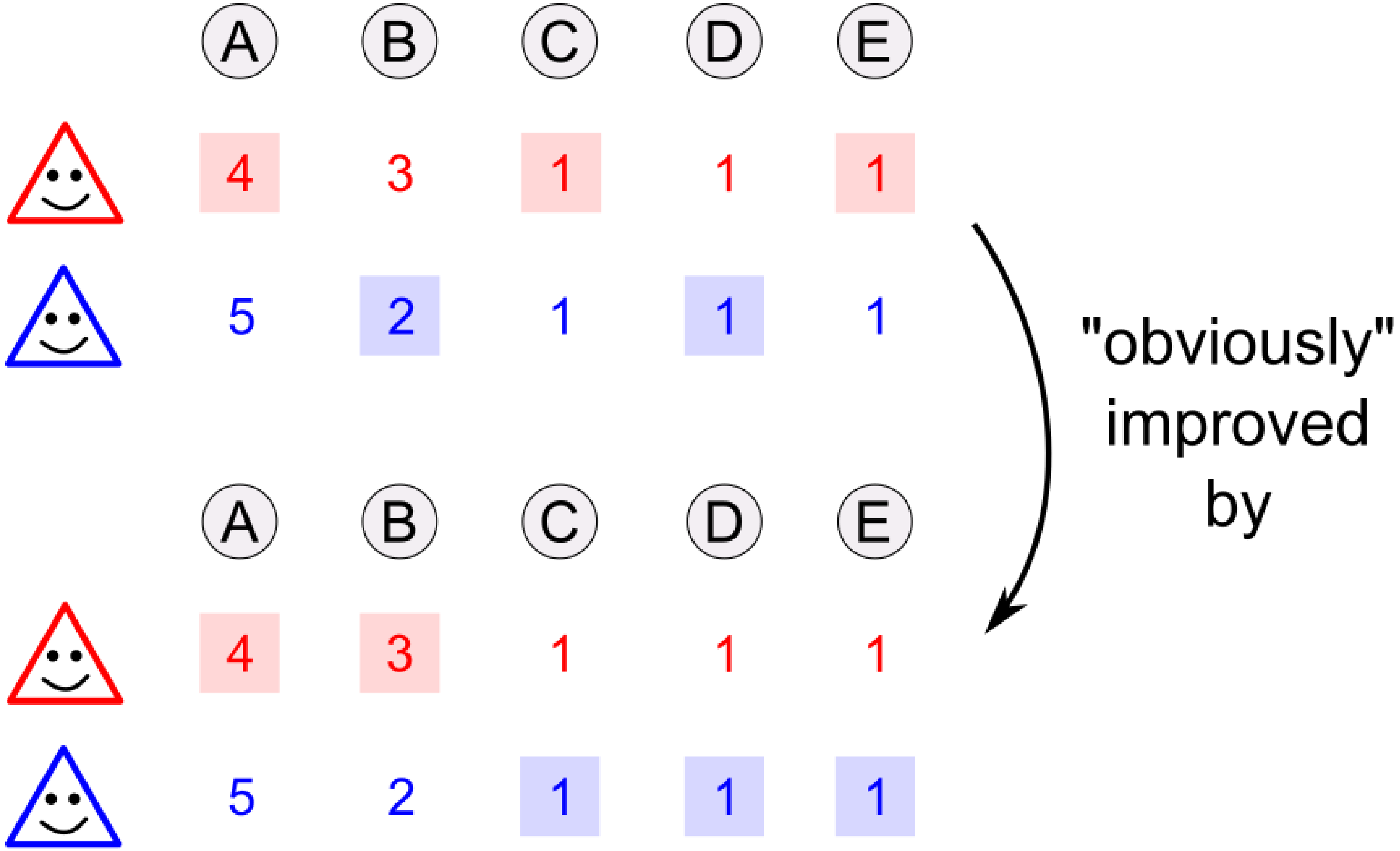
SIMPLY ISN'T ENOUGH

"Obvious" Improvement

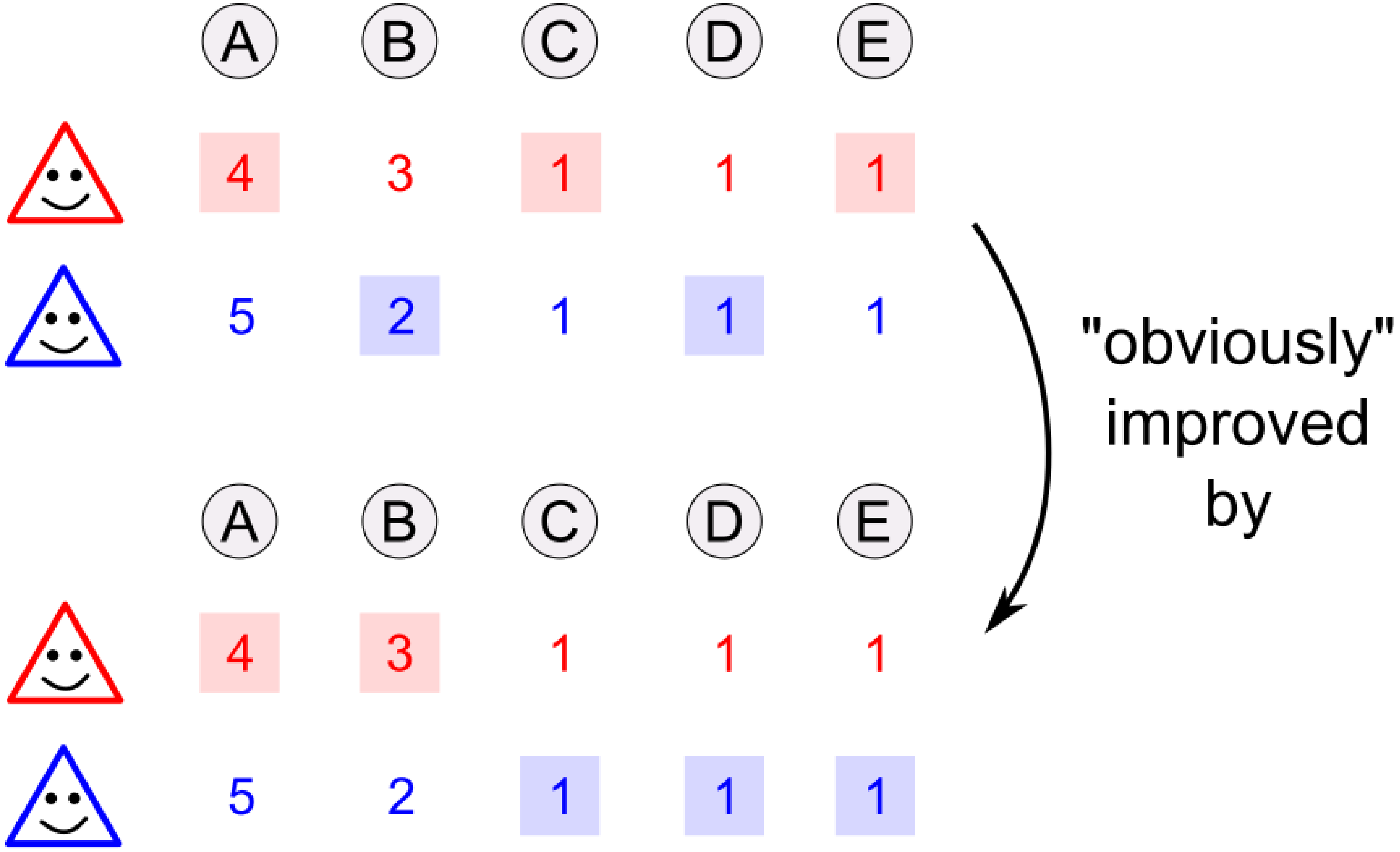
"Obvious" Improvement

	A	B	C	D	E
	4	3	1	1	1
	5	2	1	1	1

"Obvious" Improvement



"Obvious" Improvement



Strictly improving someone without hurting anyone else

Pareto Optimality

Pareto Optimality

To make someone better off, someone else must be made worse off.

Pareto Optimality

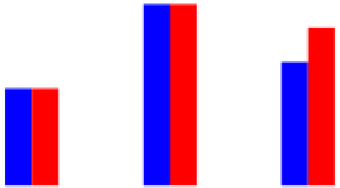
To make someone better off, someone else must be made worse off.

	A	B	C
	4	1	2
	1	1	5

Pareto Optimality

To make someone better off, someone else must be made worse off.

	A	B	C
	4	1	2
	1	1	5

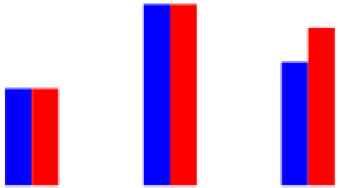
Allocation B *Pareto improves* A if 

Pareto Optimality

To make someone better off, someone else must be made worse off.

	A	B	C
	4	1	2
	1	1	5

Allocation B *Pareto improves* A if

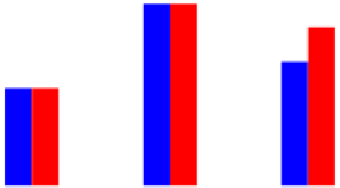


Allocation A is *Pareto optimal* (PO) if no other allocation B Pareto improves it.

Pareto Optimality

To make someone better off, someone else must be made worse off.

	A	B	C
	4	1	2
	1	1	5

Allocation B *Pareto improves* A if 

Allocation A is *Pareto optimal* (PO) if no other allocation B Pareto improves it.

 Guaranteed to exist and efficiently computable

Is EF1 compatible with Pareto optimality?

Round Robin Fails Pareto Optimality

Round Robin Fails Pareto Optimality

	A	B	C	D	E
	4	3	1	1	1
	5	2	1	1	1

Round Robin Fails Pareto Optimality

	A	B	C	D	E
	4	3	1	1	1
	5	2	1	1	1

Round Robin Fails Pareto Optimality

	A	B	C	D	E
	4	3	1	1	1
	5	2	1	1	1

Round Robin Fails Pareto Optimality

	A	B	C	D	E
	4	3	1	1	1
	5	2	1	1	1

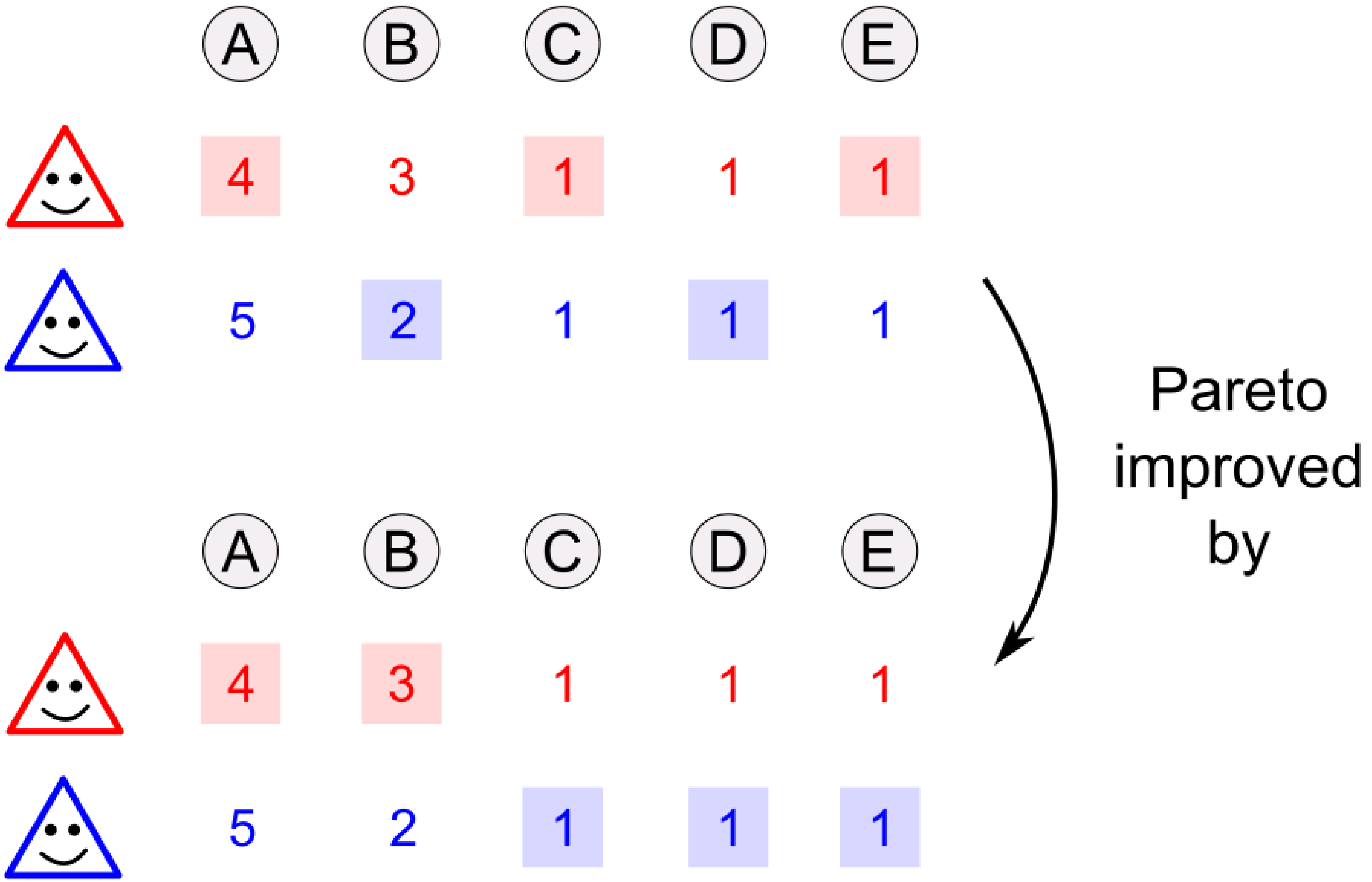
Round Robin Fails Pareto Optimality

	A	B	C	D	E
	4	3	1	1	1
	5	2	1	1	1

Round Robin Fails Pareto Optimality

	A	B	C	D	E
	4	3	1	1	1
	5	2	1	1	1

Round Robin Fails Pareto Optimality



Envy-Cycle Elimination Fails Pareto Optimality

Envy-Cycle Elimination Fails Pareto Optimality

	A	B	C
	4	3	1
	5	2	1

Envy-Cycle Elimination Fails Pareto Optimality

	A	B	C
	4	3	1
	5	2	1

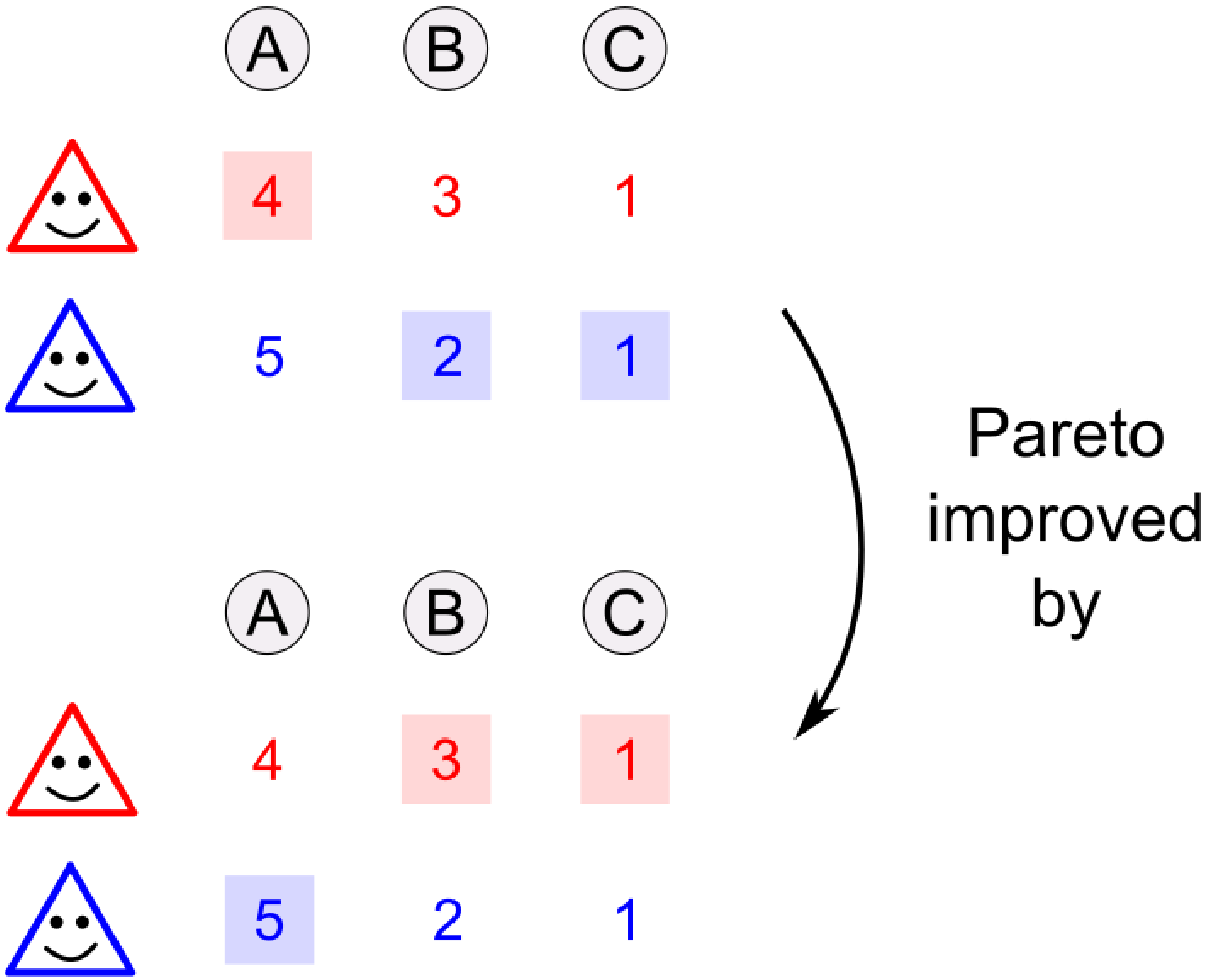
Envy-Cycle Elimination Fails Pareto Optimality

	A	B	C
	4	3	1
	5	2	1

Envy-Cycle Elimination Fails Pareto Optimality

	A	B	C
	4	3	1
	5	2	1

Envy-Cycle Elimination Fails Pareto Optimality



Another natural strategy for EF1+PO

Another natural strategy for EF1+PO

Start with an EF1 allocation, and repeatedly make Pareto improvements to it.

Another natural strategy for EF1+PO

Start with an EF1 allocation, and repeatedly make Pareto improvements to it.

Exercise: Pareto improvement can fail to preserve EF1.

Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$

Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$

	A	B	C
	4	1	2
	1	1	5

Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$

	A	B	C
	4	1	2
	1	1	5

NSW = 2

Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$

	A	B	C
	4	1	2
	1	1	5

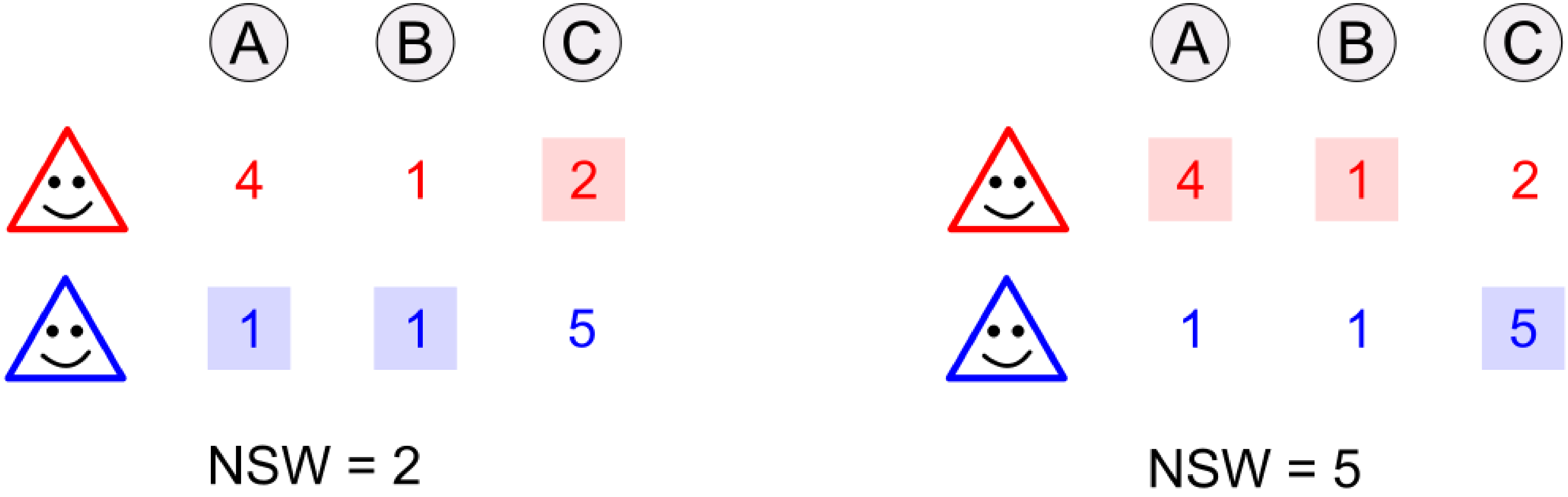
NSW = 2

	A	B	C
	4	1	2
	1	1	5

Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

$$\text{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$



Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$

	A	B	C
	4	1	2
	1	1	5

NSW = 2

	A	B	C
	4	1	2
	1	1	5

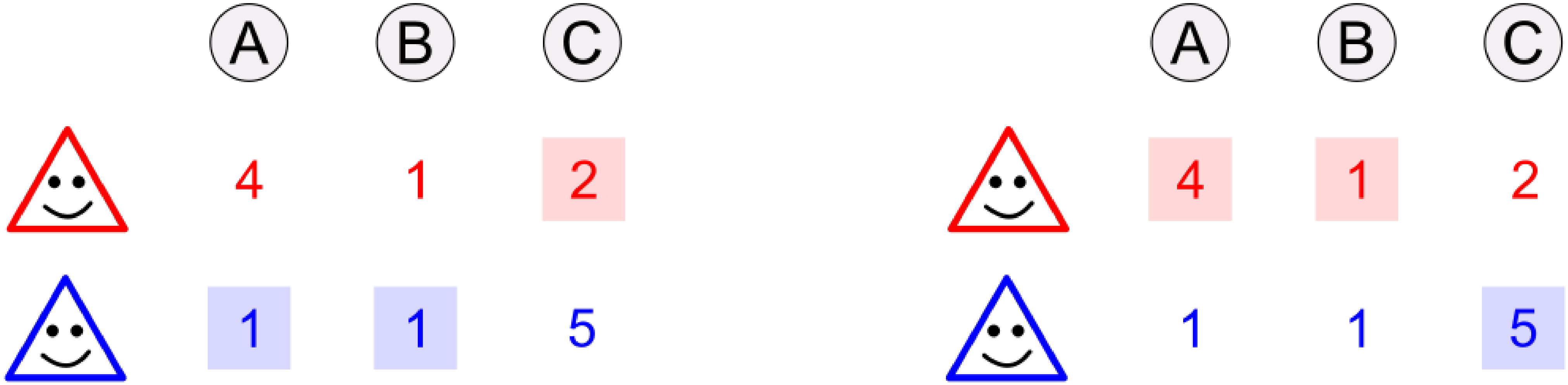
NSW = 5

A Nash optimal allocation is one that maximizes Nash social welfare.*

Nash Social Welfare

[Nash, 1950; Kaneko and Nakamura, 1979]

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n)\right)^{\frac{1}{n}}$$



NSW = 2

NSW = 5

A Nash optimal allocation is one that maximizes Nash social welfare.*

*If optimal is 0, then find the largest set of agents who can simulataneously be given positive utility and maximize the geometric mean with respect to only those agents.

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n) \right)^{\frac{1}{n}}$$

Why PO?

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

$$\mathrm{NSW}(A) = \left(v_1(A_1) \cdot v_2(A_2) \cdot \ldots v_n(A_n)\right)^{\frac{1}{n}}$$

Why PO?

Pareto improvement strictly improves NSW.

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

Let A be a Nash optimal allocation.

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

Let A be a Nash optimal allocation.

Suppose, for contradiction, that A is not EF1.

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

Let A be a Nash optimal allocation.

Suppose, for contradiction, that A is not EF1.

$$\text{for some agents } i, k \text{ and every } g \in A_k,\ v_i(A_i) < v_i(A_k \setminus \{g\})$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

Let A be a Nash optimal allocation.

Suppose, for contradiction, that A is not EF1.

$$\text{for some agents } i, k \text{ and every } g \in A_k,\ v_i(A_i) < v_i(A_k \setminus \{g\})$$

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

Let A be a Nash optimal allocation.

Suppose, for contradiction, that A is not EF1.

$$\text{for some agents } i, k \text{ and every } g \in A_k,\ v_i(A_i) < v_i(A_k \setminus \{g\})$$

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\})>0} v_k(\{g\})/v_i(\{g\}).$$

Does such a good g* always exist?

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

	A	B	C
agent i	4	2	1
agent k	1	1	5

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\})>0} v_k(\{g\})/v_i(\{g\}).$$

	A	B	C
agent i	4	2	1
agent k	1	1	5

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\})>0} v_k(\{g\})/v_i(\{g\}).$$

We will show that transferring g* from agent k to agent i improves NSW.

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

Let us transfer g* from agent k to agent i to obtain the allocation B.

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

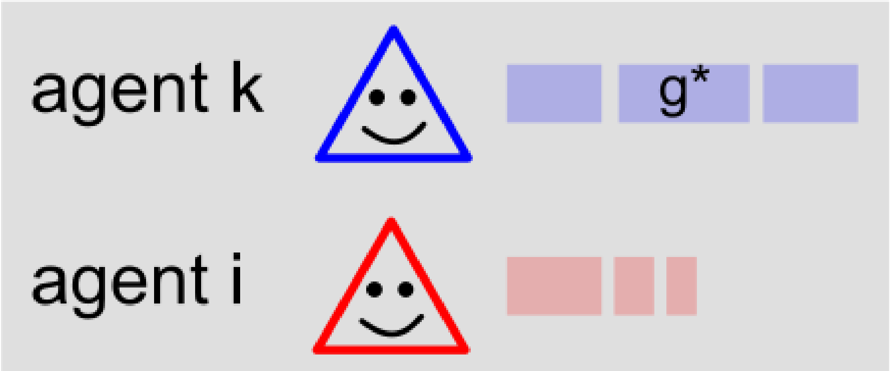
Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

Let us transfer g* from agent k to agent i to obtain the allocation B.

Allocation A



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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

Let us transfer g* from agent k to agent i to obtain the allocation B.

Allocation A　　　　Allocation B

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

Let us transfer g* from agent k to agent i to obtain the allocation B.

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

Let us transfer g* from agent k to agent i to obtain the allocation B.

$$v_i(B_i) = v_i(A_i) + v_i(g^*)$$

$$v_k(B_k) = v_k(A_k) - v_k(g^*)$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\})>0} v_k(\{g\})/v_i(\{g\}).$$

Let us transfer g* from agent k to agent i to obtain the allocation B.

$$v_i(B_i) = v_i(A_i) + v_i(g^*)$$
$$v_k(B_k) = v_k(A_k) - v_k(g^*)$$

We will show that NSW(B) > NSW(A).

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\})>0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1 \Leftrightarrow v_k(B_k) \cdot v_i(B_i) > v_k(A_k) \cdot v_i(A_i)$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1 \Leftrightarrow v_k(B_k) \cdot v_i(B_i) > v_k(A_k) \cdot v_i(A_i)$$

$$\Leftrightarrow (v_k(A_k) - v_k(g^*)) \cdot (v_i(A_i) + v_i(g^*)) > v_k(A_k) \cdot v_i(A_i)$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1 \Leftrightarrow v_k(B_k) \cdot v_i(B_i) > v_k(A_k) \cdot v_i(A_i)$$

$$\Leftrightarrow (v_k(A_k) - v_k(g^*)) \cdot (v_i(A_i) + v_i(g^*)) > v_k(A_k) \cdot v_i(A_i)$$

$$\Leftrightarrow \left(1 - \frac{v_k(g^*)}{v_k(A_k)}\right) \cdot \left(1 + \frac{v_i(g^*)}{v_i(A_i)}\right) > 1$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\mathrm{NSW}(B)}{\mathrm{NSW}(A)} > 1 \Leftrightarrow v_k(B_k) \cdot v_i(B_i) > v_k(A_k) \cdot v_i(A_i)$$

$$\Leftrightarrow (v_k(A_k) - v_k(g^*)) \cdot (v_i(A_i) + v_i(g^*)) > v_k(A_k) \cdot v_i(A_i)$$

$$\Leftrightarrow \left(1 - \frac{v_k(g^*)}{v_k(A_k)}\right) \cdot \left(1 + \frac{v_i(g^*)}{v_i(A_i)}\right) > 1$$

$$\Leftrightarrow v_k(A_k) > \frac{v_k(g^*)}{v_i(g^*)} [v_i(A_i) + v_i(g^*)]$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1 \Leftrightarrow v_k(A_k) > \frac{v_k(g^*)}{v_i(g^*)}[v_i(A_i) + v_i(g^*)]$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\})>0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1 \Leftrightarrow v_k(A_k) > \frac{v_k(g^*)}{v_i(g^*)}[v_i(A_i) + v_i(g^*)]$$

By our choice of g*: $\frac{v_k(g^*)}{v_i(g^*)} \leq \frac{\sum_{g \in A_k} v_k(g)}{\sum_{g \in A_k} v_i(g)} = \frac{v_k(A_k)}{v_i(A_k)}$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1 \Leftrightarrow v_k(A_k) > \frac{v_k(g^*)}{v_i(g^*)}[v_i(A_i) + v_i(g^*)]$$

By our choice of g*: $$\frac{v_k(g^*)}{v_i(g^*)} \leq \frac{\sum_{g \in A_k} v_k(g)}{\sum_{g \in A_k} v_i(g)} = \frac{v_k(A_k)}{v_i(A_k)}$$

By EF1 violation: $$v_i(A_i) < v_i(A_k) - v_i(g^*)$$

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\mathrm{NSW}(B)}{\mathrm{NSW}(A)} > 1 \Leftrightarrow v_k(A_k) > \frac{v_k(g^*)}{v_i(g^*)}[v_i(A_i) + v_i(g^*)]$$

By our choice of g*: $$\frac{v_k(g^*)}{v_i(g^*)} \leq \frac{\sum_{g \in A_k} v_k(g)}{\sum_{g \in A_k} v_i(g)} = \frac{v_k(A_k)}{v_i(A_k)}$$

By EF1 violation: $$v_i(A_i) < v_i(A_k) - v_i(g^*)$$

combining these

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\}) / v_i(\{g\}).$$

$$\frac{\mathrm{NSW}(B)}{\mathrm{NSW}(A)} > 1 \Leftrightarrow v_k(A_k) > \frac{v_k(g^*)}{v_i(g^*)}\left[v_i(A_i) + v_i(g^*)\right]$$ ← gives this

By our choice of g*: $$\frac{v_k(g^*)}{v_i(g^*)} \leq \frac{\sum_{g \in A_k} v_k(g)}{\sum_{g \in A_k} v_i(g)} = \frac{v_k(A_k)}{v_i(A_k)}$$

By EF1 violation: $$v_i(A_i) < v_i(A_k) - v_i(g^*)$$

combining these

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

Any Nash optimal allocation satisfies Pareto optimality (PO) and envy-freeness up to one good (EF1).

Why EF1?

$$\text{Let } g^* \in \arg\min_{g \in A_k : v_i(\{g\}) > 0} v_k(\{g\})/v_i(\{g\}).$$

$$\frac{\text{NSW}(B)}{\text{NSW}(A)} > 1 \Leftrightarrow v_k(A_k) > \frac{v_k(g^*)}{v_i(g^*)}[v_i(A_i) + v_i(g^*)]$$

← gives this

By our choice of g*: $\frac{v_k(g^*)}{v_i(g^*)} \leq \frac{\sum_{g \in A_k} v_k(g)}{\sum_{g \in A_k} v_i(g)} = \frac{v_k(A_k)}{v_i(A_k)}$

By EF1 violation: $v_i(A_i) < v_i(A_k) - v_i(g^*)$

← combining these

■

Ok, so an EF1+PO allocation always exists.

But what about computation?

Can a Nash optimal allocation be efficiently computed?

Can a Nash optimal allocation be efficiently computed?

[Lee, *Info. Proc. Lett.* 2017]

Maximizing Nash social welfare is APX-hard.

Can a Nash optimal allocation be efficiently computed?

[Lee, *Info. Proc. Lett.* 2017]

Maximizing Nash social welfare is APX-hard.

- Even for bounded valuations

Can a Nash optimal allocation be efficiently computed?

[Lee, *Info. Proc. Lett.* 2017]

Maximizing Nash social welfare is APX-hard.

- Even for bounded valuations

Can an EF1+PO allocation be efficiently computed?

Can a Nash optimal allocation be efficiently computed?

[Lee, *Info. Proc. Lett.* 2017]

Maximizing Nash social welfare is APX-hard.

- Even for bounded valuations

Can an EF1+PO allocation be efficiently computed?

[Barman, Krishnamurthy, and Vaish, *EC* 2018]

An EF1+PO allocation can be computed in pseudopolynomial time.

Can a Nash optimal allocation be efficiently computed?

[Lee, *Info. Proc. Lett.* 2017]

Maximizing Nash social welfare is APX-hard.

- Even for bounded valuations

Can an EF1+PO allocation be efficiently computed?

[Barman, Krishnamurthy, and Vaish, *EC* 2018]

An EF1+PO allocation can be computed in pseudopolynomial time.

- Running time depends on $v_{i,j}$'s rather than log $v_{i,j}$'s

Can a Nash optimal allocation be efficiently computed?

[Lee, *Info. Proc. Lett.* 2017]

Maximizing Nash social welfare is APX-hard.

- Even for bounded valuations

Can an EF1+PO allocation be efficiently computed?

[Barman, Krishnamurthy, and Vaish, *EC* 2018]

An EF1+PO allocation can be computed in pseudopolynomial time.

- Running time depends on $v_{i,j}$'s rather than log $v_{i,j}$'s
- Polynomial time for bounded valuations

Can a Nash optimal allocation be efficiently computed?

[Lee, *Info. Proc. Lett.* 2017]

Maximizing Nash social welfare is APX-hard.

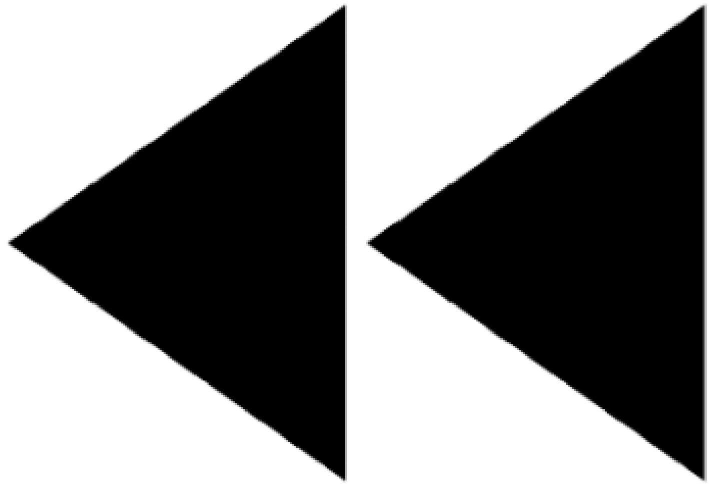
- Even for bounded valuations

Can an EF1+PO allocation be efficiently computed?

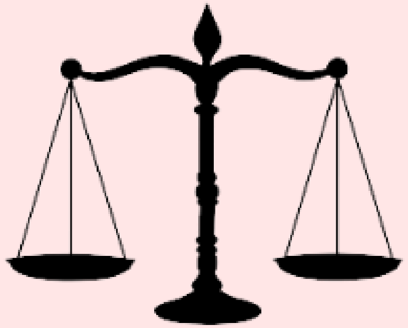
[Barman, Krishnamurthy, and Vaish, *EC* 2018]

An EF1+PO allocation can be computed in pseudopolynomial time.

- Running time depends on $v_{i,j}$'s rather than log $v_{i,j}$'s
- Polynomial time for bounded valuations
- A 0.69-approximation to Nash social welfare objective

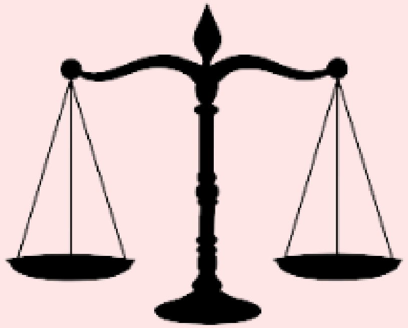



VOTE



[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

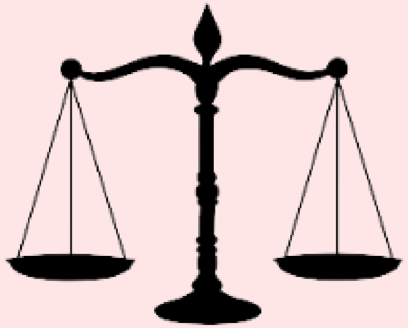


[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating
certain voting rules



[Lec 1+2]

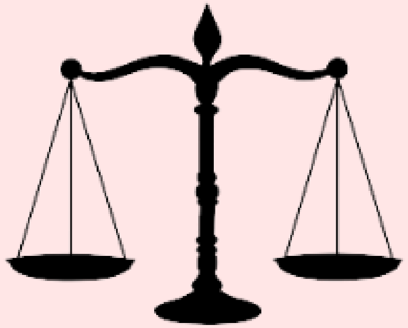
Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences



[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences

[Lec 5]

Top-trading cycle algorithm

[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences

[Lec 5]

Top-trading cycle algorithm

[Lec 6]

Deferred-acceptance algorithm

[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences

[Lec 5]

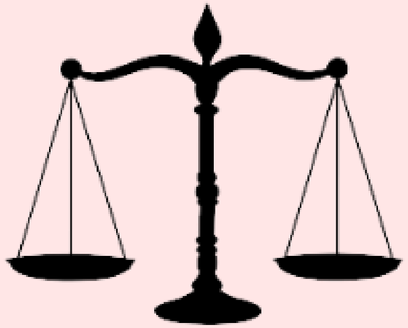
Top-trading cycle algorithm

[Lec 6]

Deferred-acceptance algorithm

[Lec 7]

Inconspicuous manipulation



[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences

[Lec 5]

Top-trading cycle algorithm

[Lec 6]

Deferred-acceptance algorithm

[Lec 7]

Inconspicuous manipulation

[Lec 8]

Geometric decomposition via linear programming

[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences

[Lec 5]

Top-trading cycle algorithm

[Lec 6]

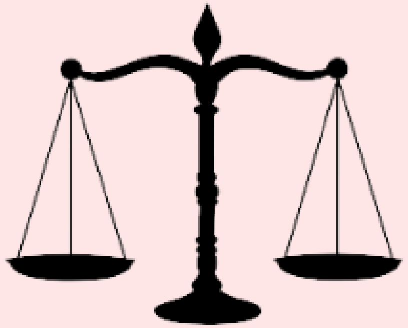
Deferred-acceptance algorithm

[Lec 7]

Inconspicuous manipulation

[Lec 8]

Geometric decomposition via linear programming



[Lec 9]

Envy-free cake-cutting

[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences

[Lec 5]

Top-trading cycle algorithm

[Lec 6]

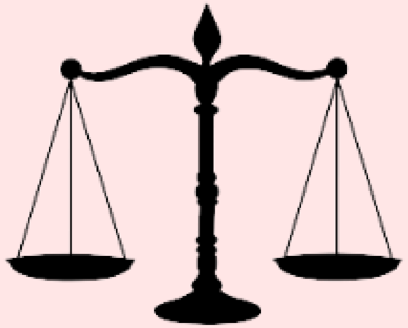
Deferred-acceptance algorithm

[Lec 7]

Inconspicuous manipulation

[Lec 8]

Geometric decomposition via linear programming



[Lec 9]

Envy-free cake-cutting

[Lec 10]

Rent division via Sperner's lemma

[Lec 1+2]

Voting rules:
Good, bad, and dictatorial

[Lec 3]

NP-hardness of manipulating certain voting rules

[Lec 4]

Detecting single-peaked preferences

[Lec 5]

Top-trading cycle algorithm

[Lec 6]

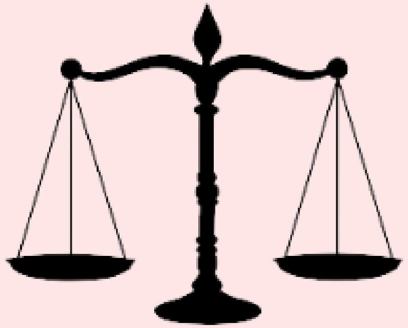
Deferred-acceptance algorithm

[Lec 7]

Inconspicuous manipulation

[Lec 8]

Geometric decomposition via linear programming



[Lec 9]

Envy-free cake-cutting

[Lec 10]

Rent division via Sperner's lemma

[Lec 11+12]

Fair division of the indivisibles

Next Semester

COL866: Special Topics in Algorithms

- A deeper dive into topics in CS + Economics
- Great starting point for a research project or thesis in this area

Reminders

Project presentations in online mode on Apr 03 (Sun).

Project reports due by Apr 10 (Sun).

Thank you!

References

- Maximum Nash Welfare is EF1 and PO.

  Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang
  “The Unreasonable Fairness of Maximum Nash Welfare”
  ACM Transactions on Economics and Computation, 7(3), 2019 pg 1-32
  https://dl.acm.org/doi/10.1145/3355902

- APX-hardness of maximizing Nash social welfare

  Euiwoong Lee
  “*APX-Hardness of Maximizing Nash Social Welfare with Indivisible Items*”
  Information Processing Letters, 122, 2017 pg 17-20
  https://www.sciencedirect.com/science/article/pii/S0020019017300212

References

- Finding an EF1 and PO allocation in pseudopolynomial time.

  Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish
  “Finding Fair and Efficient Allocations”
  EC 2018, pg 557-574
  https://dl.acm.org/doi/10.1145/3219166.3219176