Almost Envy-Free Allocations with Connected Bundles

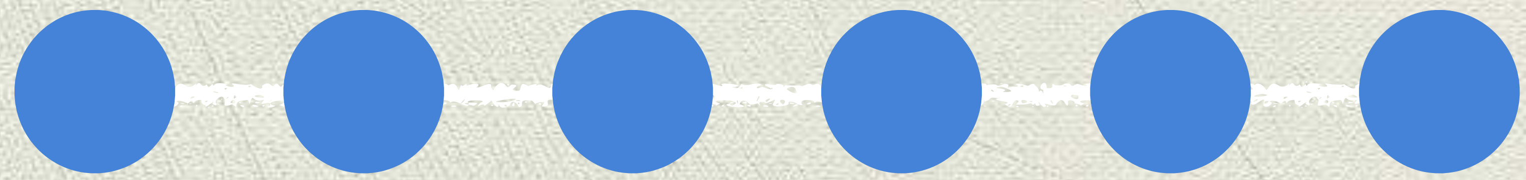
Dominik Peters
University of Oxford

Version 2018-10-31

based on joint work with
Vittorio Bilò, Ioannis Caragiannis, Michele Flammini, Ayumi Igarashi,
Gianpiero Monaco, Cosimo Vinci, and William S. Zwicker

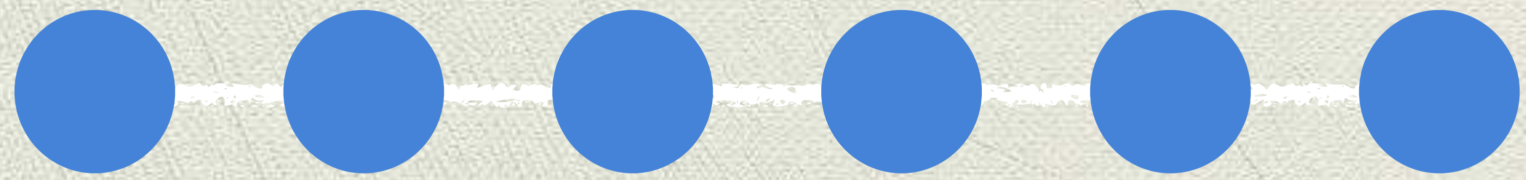
Allocation of a Graph

- Finite set of indivisible items
- Agents have preferences over bundles of items
  - always monotone, usually additive
- Goal: Allocate items to agents
- Items are arranged in a graph, only allowed to hand out *connected* bundles



Classical Fairness Concepts

- *Fair Division of a Graph*
  IJCAI-17, *Bouveret, Cechlárová, Elkind, Igarashi, and P*.
- NP-complete to decide existence of envy-free or proportional allocations, even on a path
- Tractable if there are few player types
- For a star, proportionality is easy, but envy-free is hard



Parameterised Results

Fair Division of a Graph [IJCAI-17]

- **Proportionality on paths: XP wrt # of types**
  Dynamic programming: "Does there exist an allocation of the first q items such that h_i players of type i are happy?"

- **Proportionality on paths: FPT wrt # of players**
  Dynamic programming and matching.

- **Envy-free on paths: XP wrt # of types**
  In an envy-free solution, all players of the same type need to have the same utility. Guess each type's utility (there are only $\binom{m}{2}$ options). Then dynamic programming.

MMS

Fair Division of a Graph [IJCAI-17]

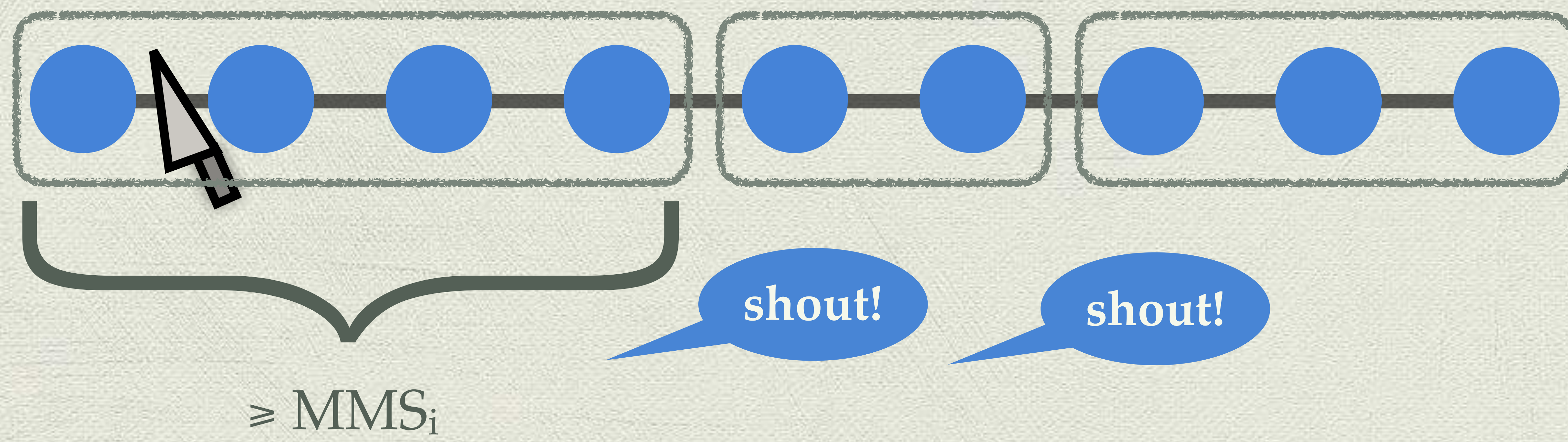
- The maximin share of a player is

$$\mathrm{mms}_i(I) = \max_{(P_1,\dots,P_n)\in \Pi_n} \min_{j\in[n]} u_i(P_j).$$

- Intuition: Cut into n pieces, choose last.
- An MMS allocation must give each player utility ≥ mms_i
- Due to Budish [2011]. Need not exist for general graphs [Kurokawa et al., EC-14, AAAI-16, JACM-18].
- Maximise over connected partitions only => MMS values smaller than normal
- Adaptation of a moving knife protocol produces an allocation where every player receives at least their MMS share.

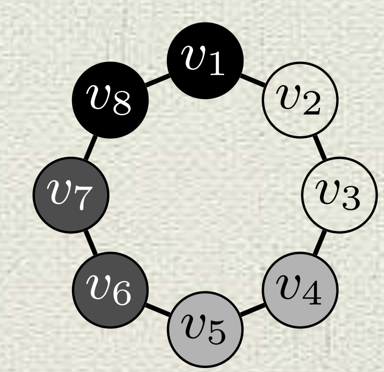
MMS: last diminisher

Fair Division of a Graph [IJCAI-17]



Algorithm also useful for α-MMS
in standard setting [Ghodsi et al., EC-18]

MMS need not exist on a cycle [IJCAI-17]
but approximations exist [Long & Truszczynski, IJCAI-18]



EF1

- An allocation is envy free if $u_i(B_i) \geqslant u_i(B_j)$ for all i,j.
- Envy-freeness need not exist —> Relaxations?
- Budish [2011] proposes envy-freeness up to one good
  - For each i,j, there is $o \in B_j$ with $u_i(B_i) \geqslant u_i(B_j \setminus \{o\})$
- Without connectivity constraints, EF1 always exists
  - Envy-graph algorithm due to Lipton et al. [EC-04]
  - Round-robin procedure [Caragiannis et al., EC-16]
  - Maximum Nash welfare [Caragiannis et al., EC-16]

EF1 on a path

- We show that EF1 exists on a path
  - when there are 2 agents (cut-and-choose)
  - when there are 3 agents (Stromquist's procedure)
  - when there are 4 agents (Sperner's lemma)
  - when valuations are identical (≈ leximin)
- By Sperner's lemma, EF2 always exists
- Existence extends to graphs with Hamiltonian path
- Existence does not require additive valuations

+MMS

Asides

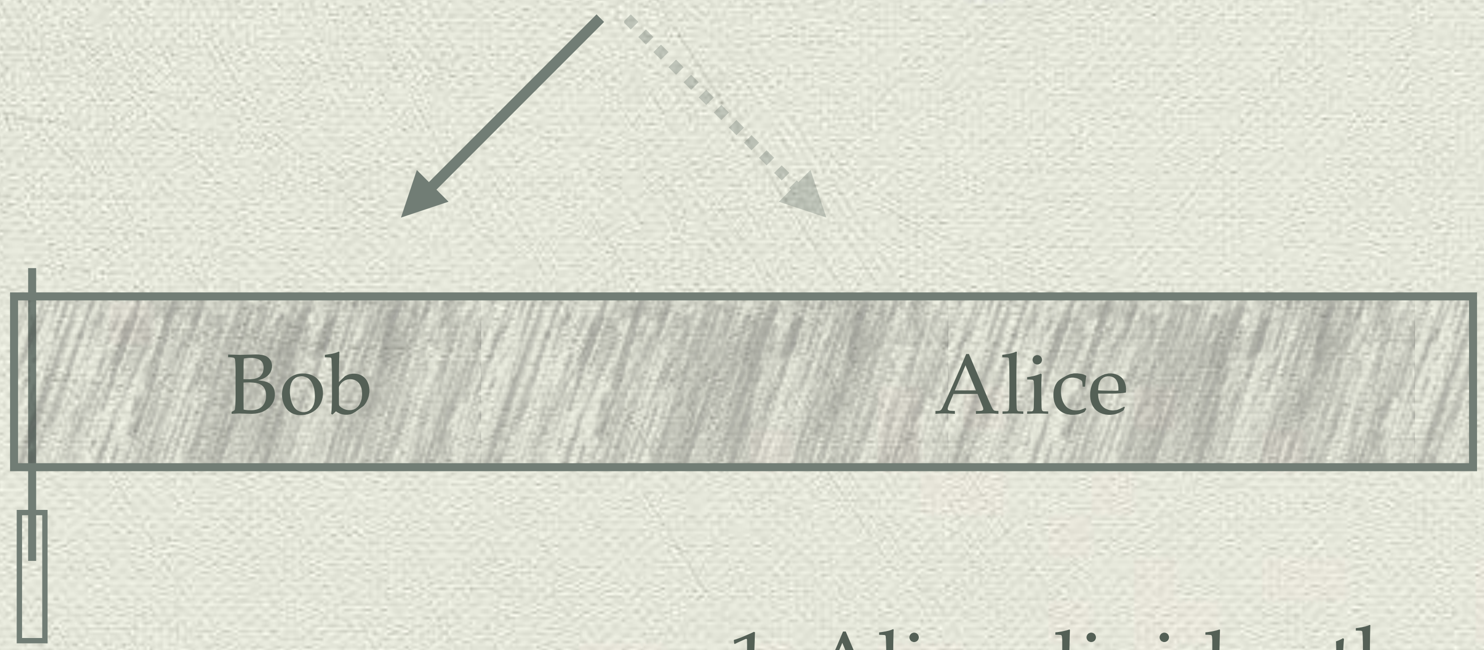
- EFx does not exist:
  Consider 2—3—1—3.

- Can try to get EF1 by rounding a fractional EF allocation, but this approach only gives EF1 for binary additive valuations.

Cut-and-choose protocol

Divisible cake:

2. Bob chooses preferred piece and receives it



1. Alice divides the cake into two equally-valued pieces

3. Alice receives other piece

Lumpy tie

v is an agent i's lumpy tie if

and

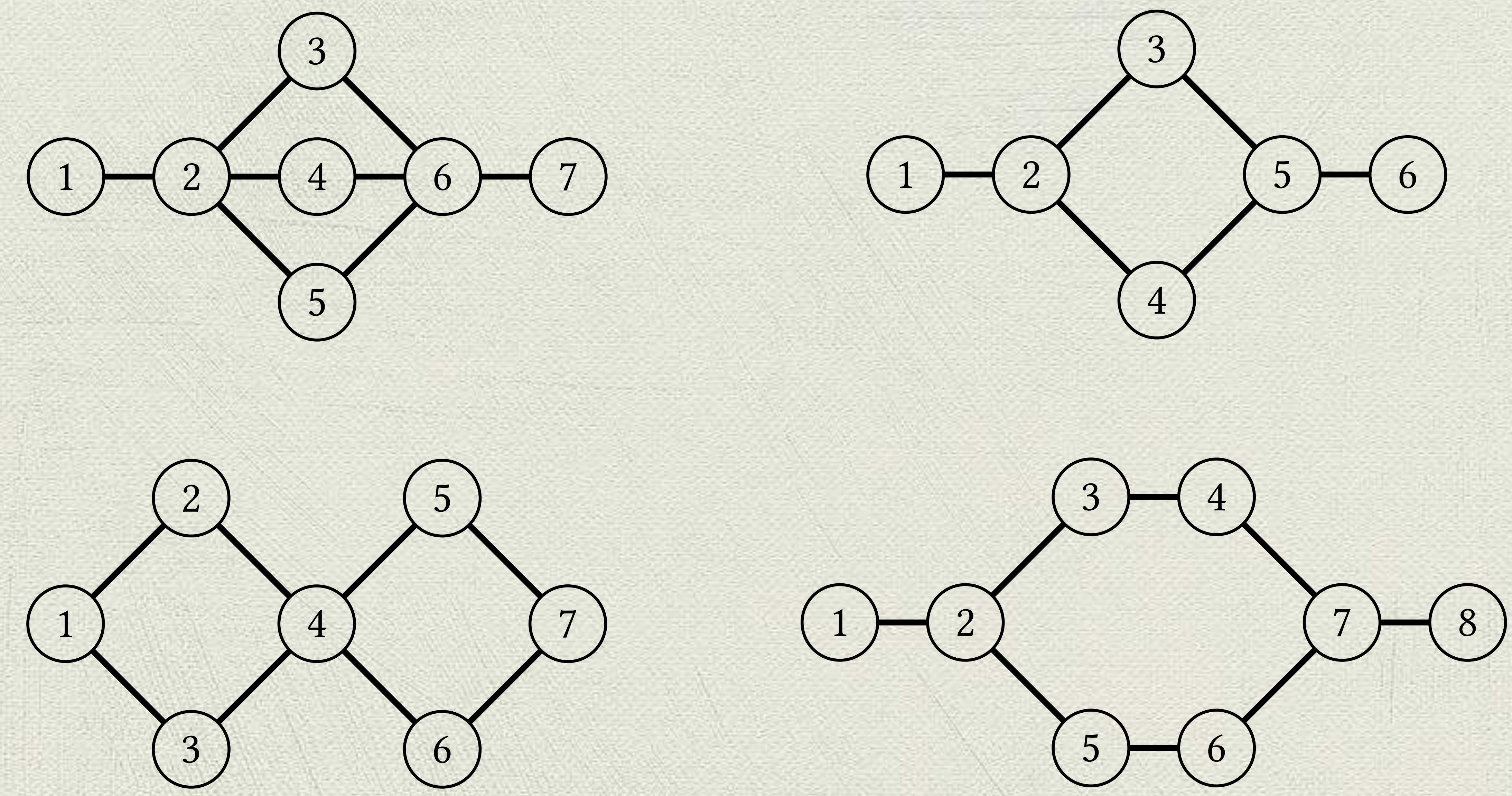
Discrete cut-and-choose

- Alice selects her lumpy tie v and takes it

- Bob selects either the left or right piece
- Alice receives v and the remaining piece
- This is EF1.

works for all traceable graphs
— any others?

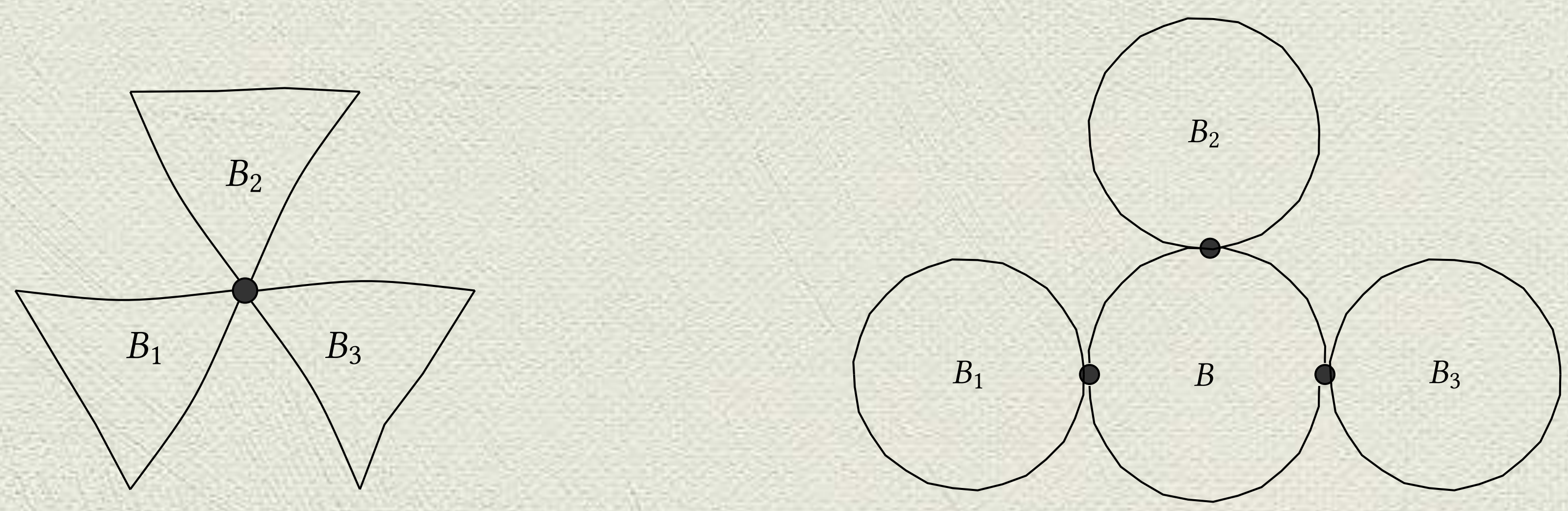
Cut-and-choose: bipolar numbering



Characterisation

THEOREM 3.8. *The following conditions are equivalent for every connected graph G:*

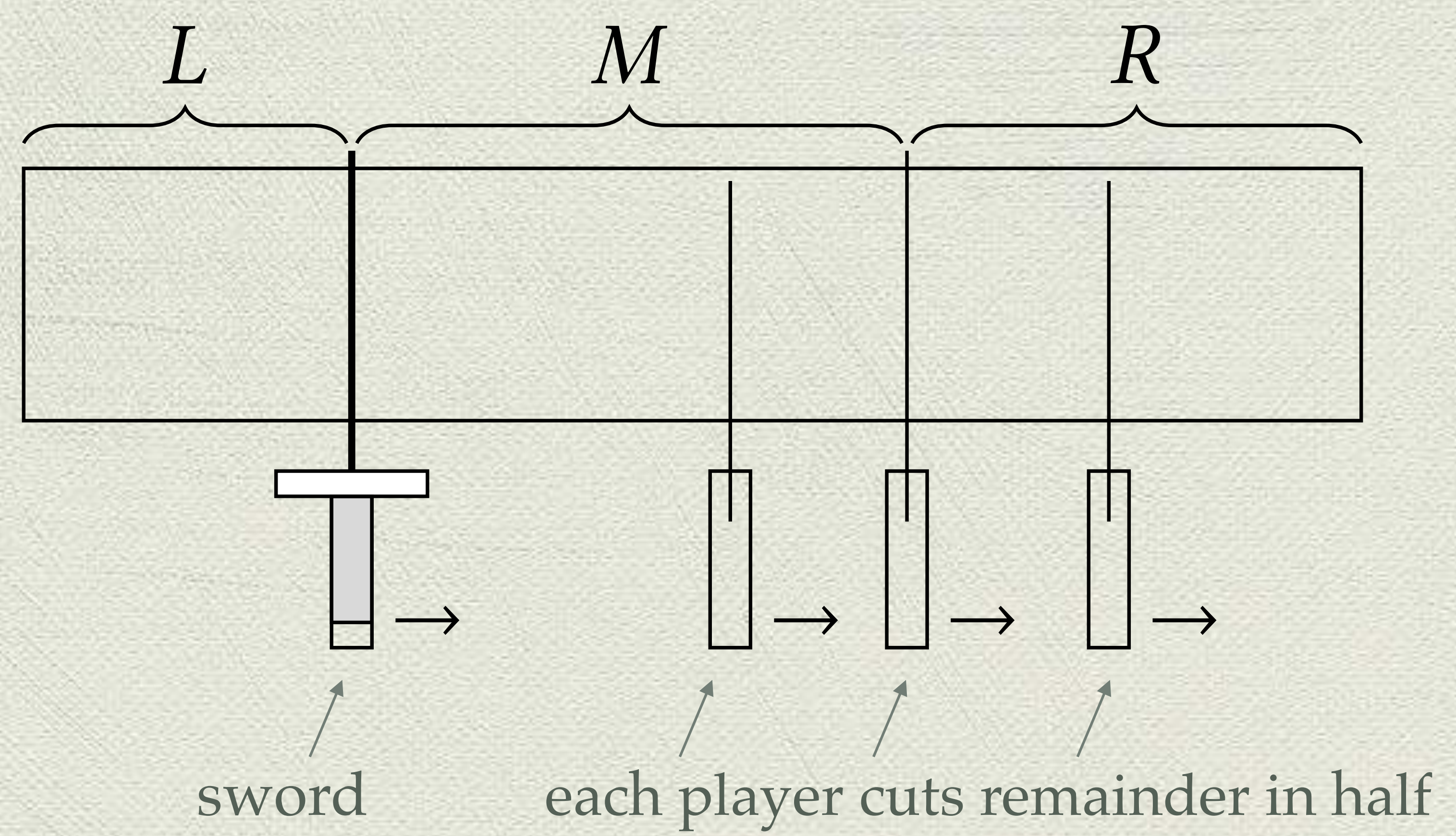
(1) G admits a bipolar numbering.
(2) G guarantees EF1 for two agents.
(3) G guarantees EF1 for two agents with identical, additive, binary valuations.



(a) A cut vertex adjacent to three blocks (b) A block adjacent to three cut vertices

Fig. 3. Tridents.

Stromquist's Moving-Knife

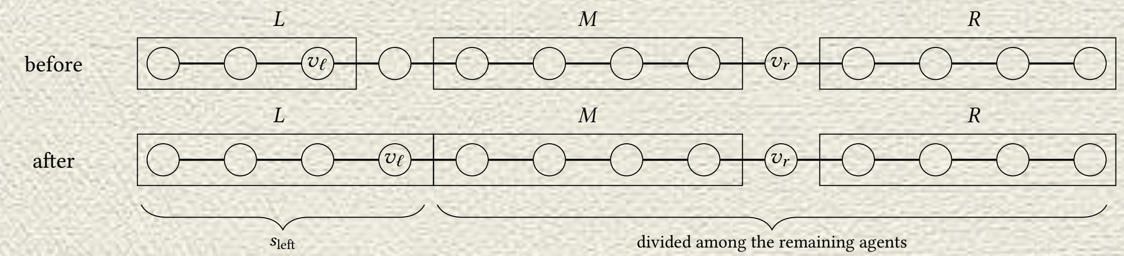


player shouts if L is better than M and R

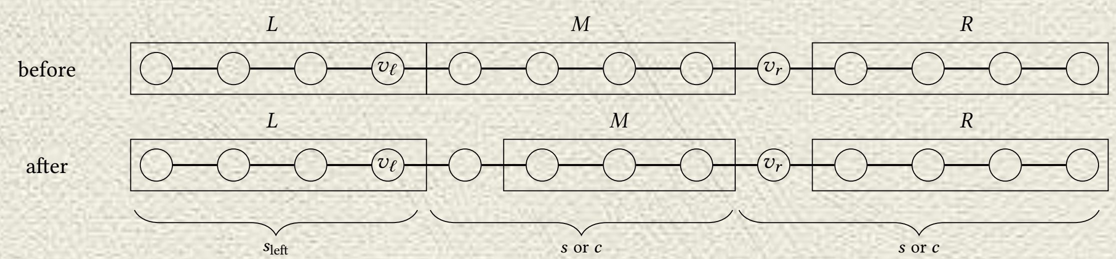
Discrete moving-knife protocol for $n = 3$ agents over a sequence $P = (v_1, v_2, \ldots, v_m)$:
An agent $i \in N$ is a *shouter* if L is best among L, M, R, so that $u_i(L) \geqslant u_i(M)$ and $u_i(L) \geqslant u_i(R)$.

Step 1. Initialize $\ell = 0$ and set r so that v_r is the median lumpy tie over the subsequence $P(v_2, v_m)$. Initialize $L = \emptyset$, $M = \{v_2, v_3, \ldots, v_{r-1}\}$, and $R = \{v_{r+1}, v_{r+2}, \ldots, v_m\}$.

Step 2. Add an additional item to L, i.e., set $\ell = \ell + 1$ and $L = \{v_1, v_2, \ldots, v_\ell\}$. If no agent shouts, go to Step 3. If some agent s_{left} shouts, s_{left} receives the left bundle L. Allocate the remaining items according to $\mathsf{Lumpy}(N \setminus \{s_{\text{left}}\}, v_r, P(v_{\ell+1}, v_m))$.



Step 3. Delete the left-most point of the middle bundle, i.e., set $M = \{v_{\ell+2}, v_{\ell+3}, \ldots, v_{r-1}\}$. If the number of shouters is smaller than two, go to Step 4. If at least two agents shout, we show (next page) that there is a shouter s who is a middle agent over $P(v_{\ell+1}, v_m)$. Then, allocate L to a shouter s_{left} distinct from s. Let the agent c distinct from s and s_{left} choose his preferred bundle among $\{v_{\ell+1}\} \cup M$ and $\{v_r\} \cup R$. Agent s receives the other bundle.



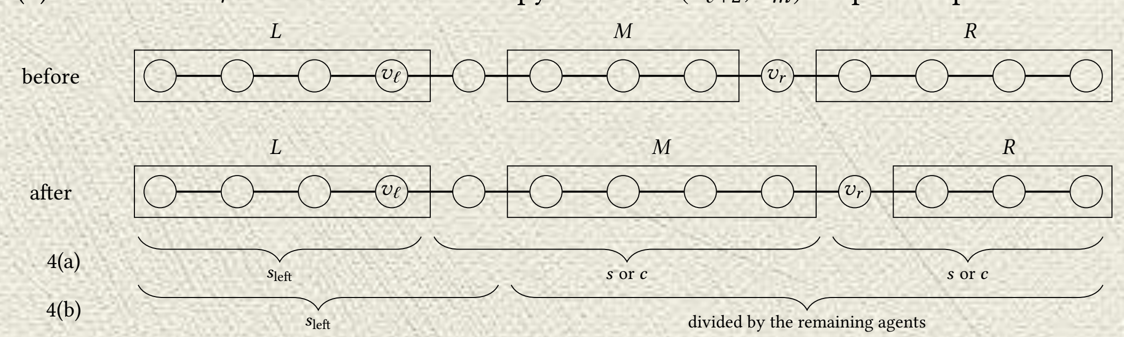
Step 4. If v_r is the median lumpy tie over $P(v_{\ell+2}, v_m)$, directly move to the following cases (a)–(d). If v_r is not the median lumpy tie over $P(v_{\ell+2}, v_m)$, set $r = r + 1$, $M = \{v_{\ell+2}, v_{\ell+3}, \ldots, v_{r-1}\}$, and $R = \{v_{r+1}, v_{r+2}, \ldots, v_m\}$; then, consider the following cases (a)–(d).

(a) If at least two agents shout, find a shouter s who did not shout at the previous step. If there is a shouter s_{left} who shouted at the previous step, s_{left} receives L; else, give L to an arbitrary shouter s_{left} distinct from s. The agent c distinct from s and s_{left} choose his preferred bundle among $\{v_{\ell+1}\} \cup M$ and $\{v_r\} \cup R$, breaking ties in favor of the former option. Agent s receives the other bundle.

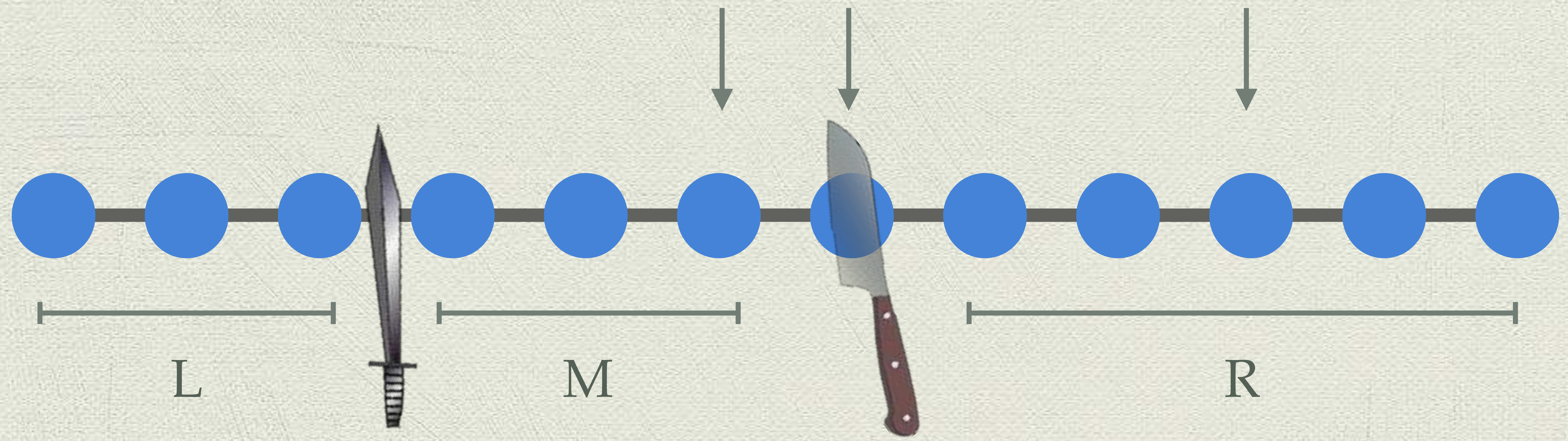
(b) If v_r is the median lumpy tie over $P(v_{\ell+2}, v_m)$ and only one agent s_{left} shouts, give $L \cup \{v_\ell\}$ to s_{left} and allocate the rest according to $\mathsf{Lumpy}(N \setminus \{s_{\text{left}}\}, v_r, P(v_{\ell+2}, v_m))$.

(c) If v_r is the median lumpy tie over $P(v_{\ell+2}, v_m)$ but no agent shouts, go to Step 2.

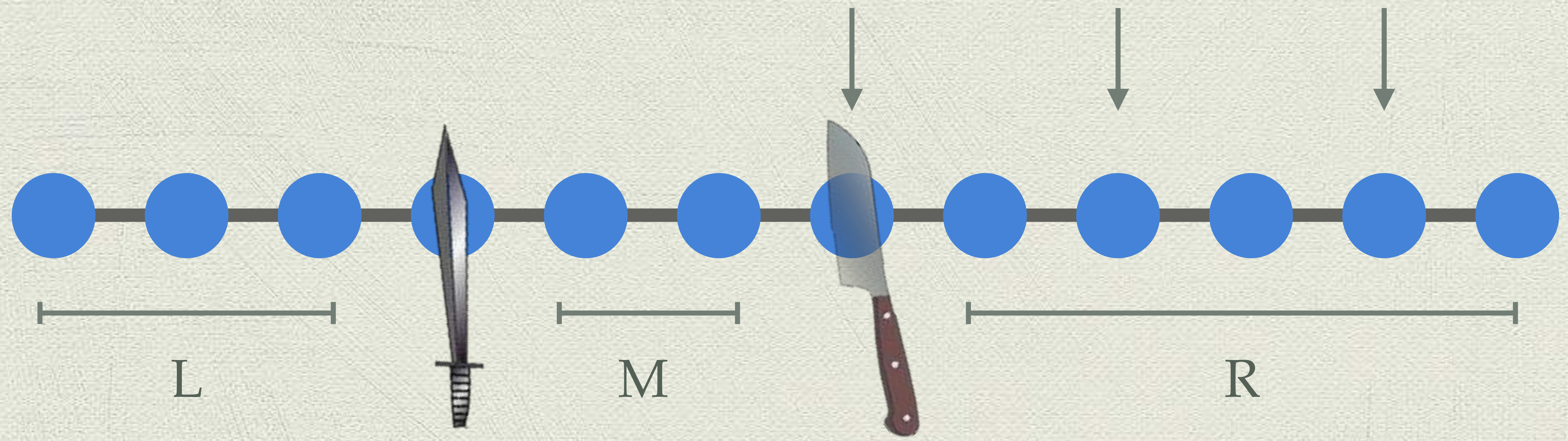
(d) Otherwise v_r is not the median lumpy tie over $P(v_{\ell+2}, v_m)$: Repeat Step 4.



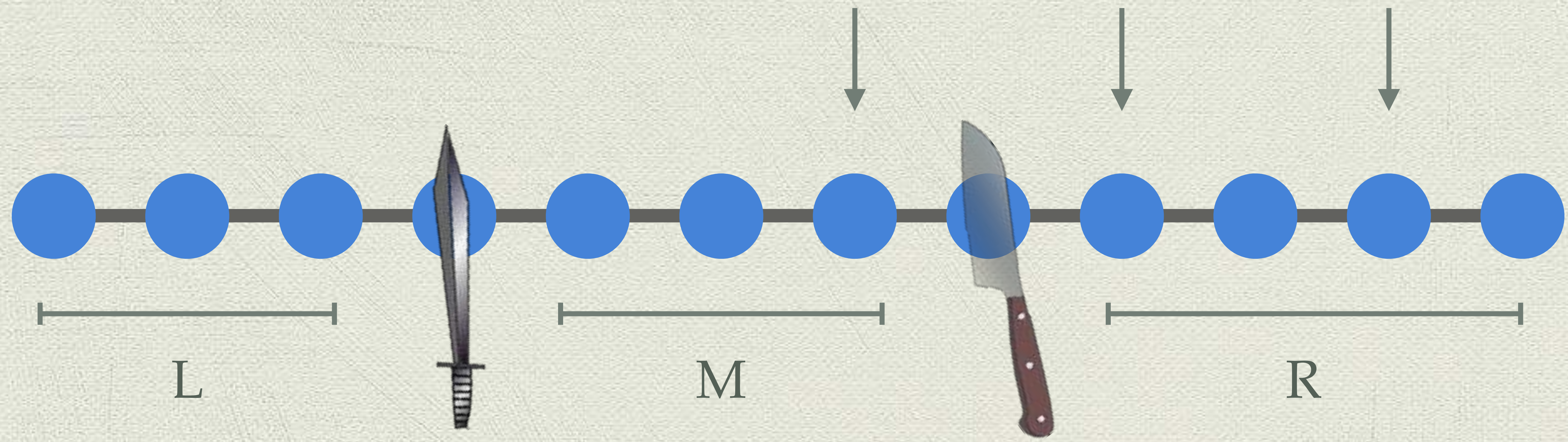
Stromquist's Moving-Knife



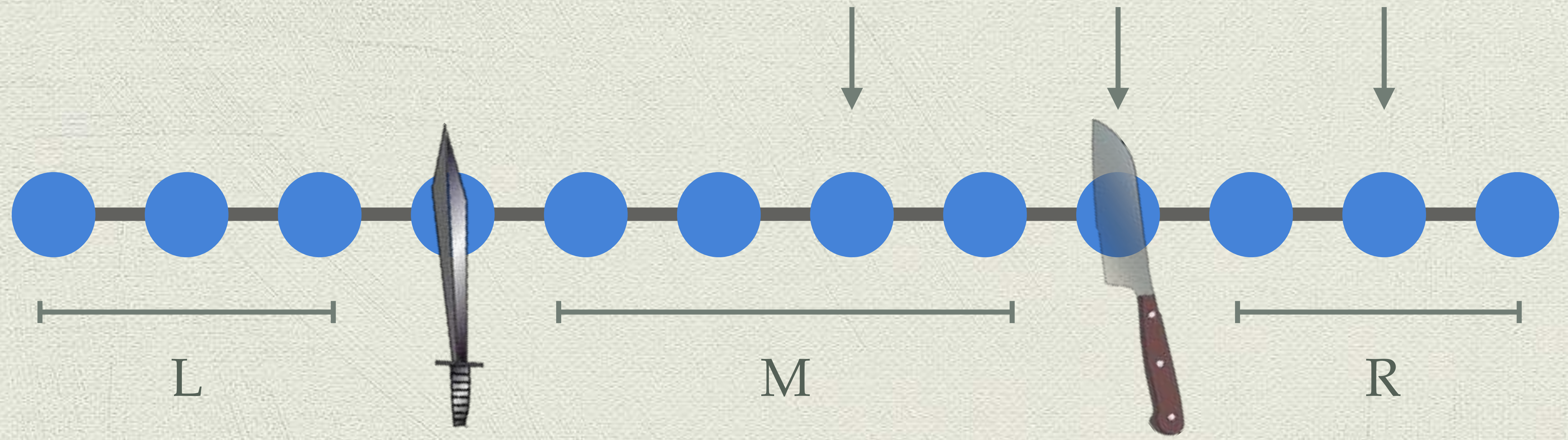
Stromquist's Moving-Knife



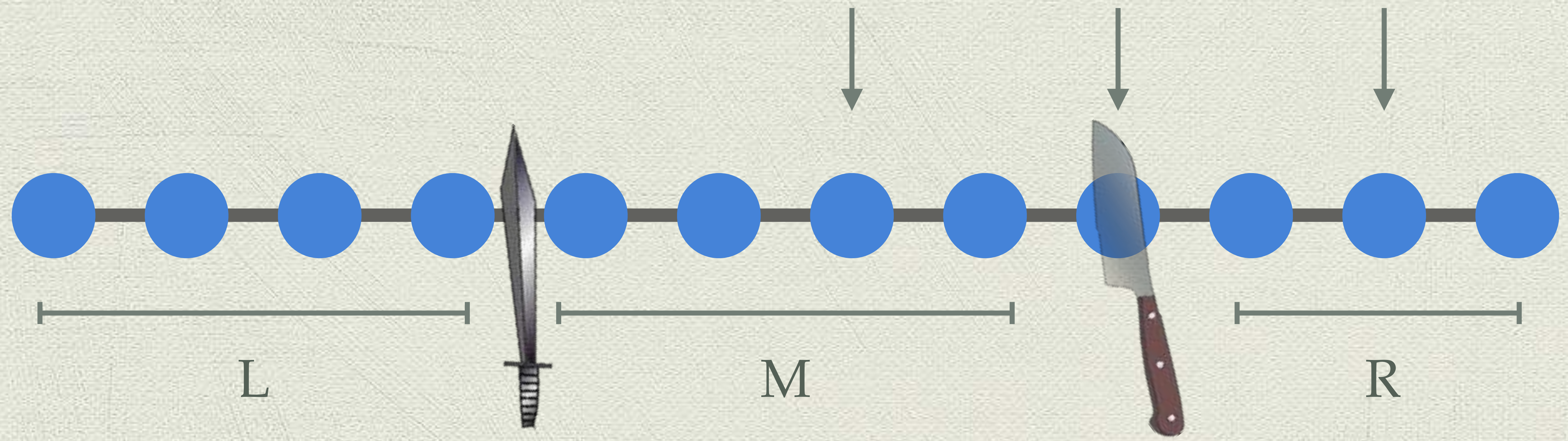
Stromquist's Moving-Knife



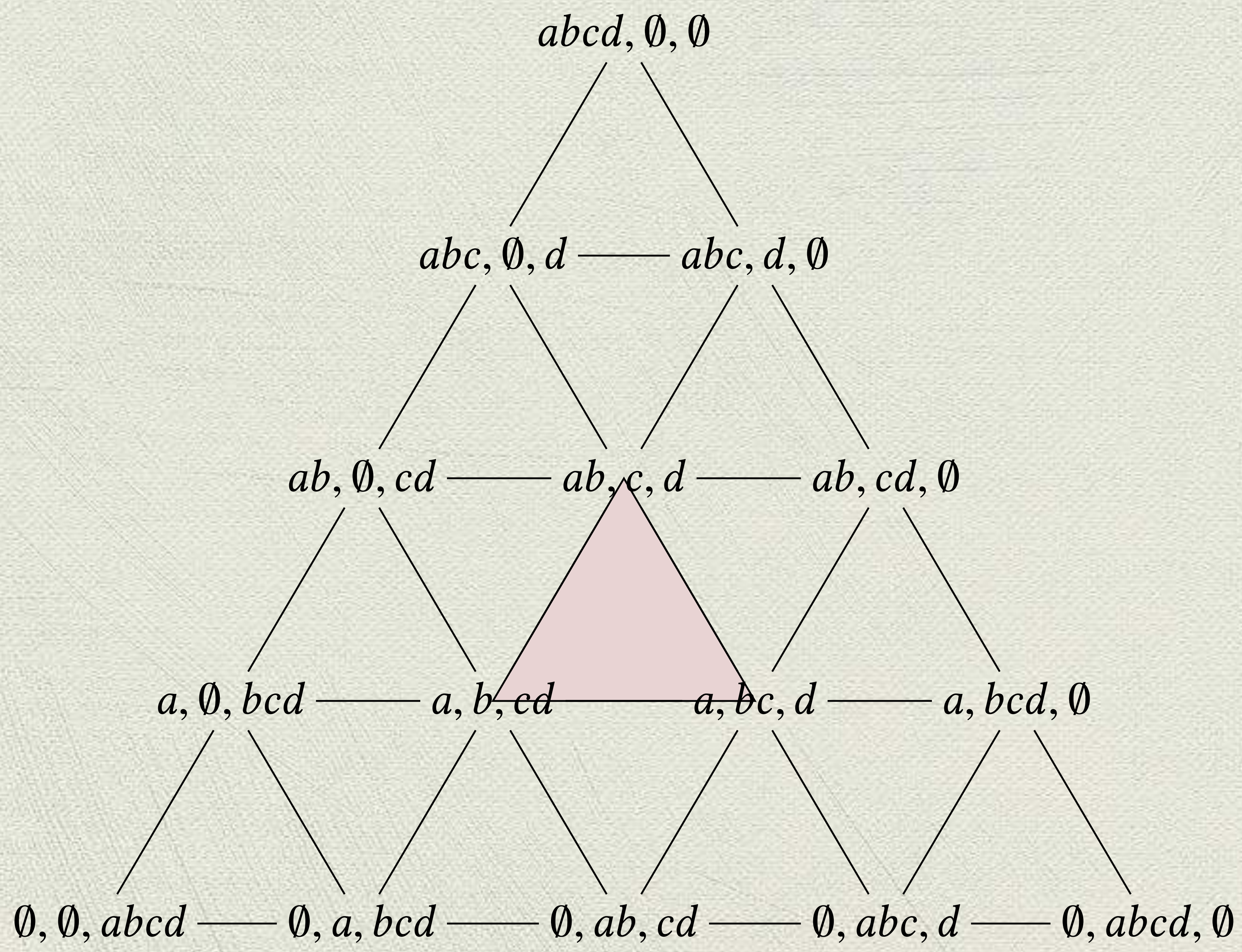
Stromquist's Moving-Knife



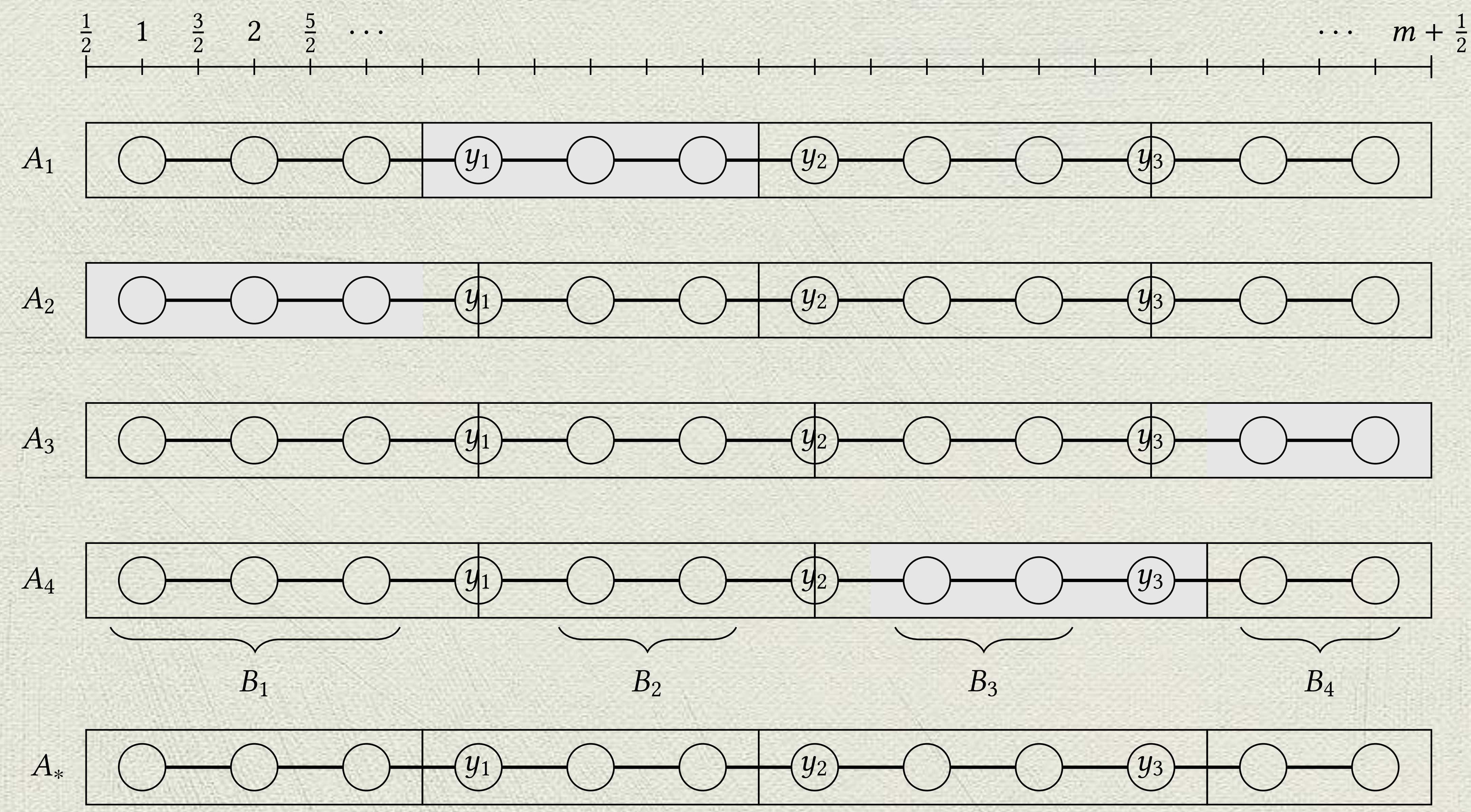
Stromquist's Moving-Knife



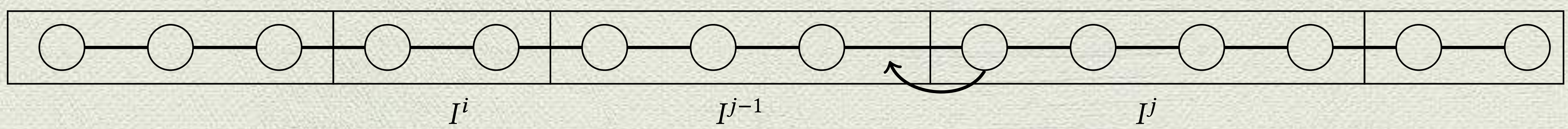
Sperner's Lemma: Simplex



Sperner's Lemma



Identical Valuations



Algorithm 1 LEXIMIN-TO-EF1

Input: a path $P = (v_1, v_2, \ldots, v_m)$, and identical valuations
Output: an EF1 connected allocation of P

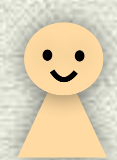
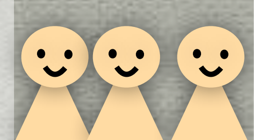
```
Let A = (I^1, ..., I^n) be a leximin allocation of P
Fix an agent i with minimum utility in A, i.e., u(I^i) ⩽ u(I^j) for all j ∈ [n]
for j = 1, ..., i − 1 do
    if i envies I^j even up to one good, i.e., u(I^i) < u^−(I^j) then
        repeatedly delete the right-most item of I^j and add it to I^{j+1} until u(I^i) ⩾ u^−(I^j)
    end if
end for
for j = n, ..., i + 1 do
    if i envies I^j even up to one good, i.e., u(I^i) < u^−(I^j) then
        repeatedly delete the left-most item of I^j and add it to I^{j−1} until u(I^i) ⩾ u^−(I^j)
    end if
end for
return A
```

EF1 + MMS

- All of our existence arguments produce allocations that are also MMS!
- Plausible guess: EF1 => MMS?
- No: consider 3—1—1—1—3, and the EF1 allocation (3—1, 1, 1—3).
- Can show: EF1 => 1/3-MMS.
- (Without connectivity, EF1 => 1/n-MMS [EC-16])

PO + EF1

There are instances where no EF1 allocation is Pareto-optimal:

	0	0	0	1	1	0	0	0
	1	1	1	0	0	1	1	1

$\in \Sigma_2^p$

On paths, NP-hard to decide whether a PO + EF1 allocation exists

there is bigger example where 1's form intervals

Future Directions

- Same thing for chores
- Existence for $n > 4$
- Complexity of finding EF1/EF2
- Restricted utility classes
- Local envy-freeness: only envy bundles next to yours