A Global Multi-Sided Market with Ascending-Price Mechanism

Rica Gonen &
Erel Segal-Halevi

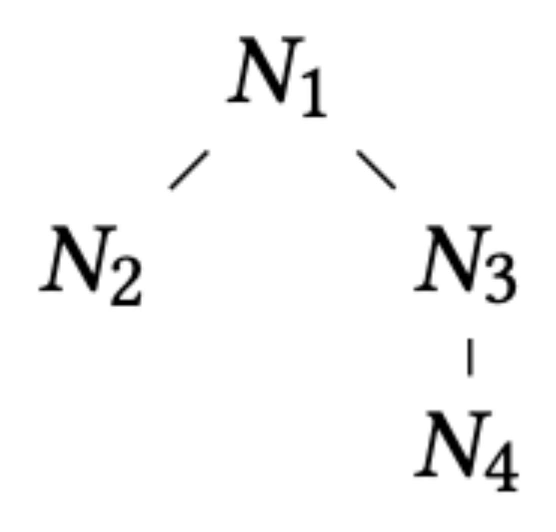
We consider ***multi-sided*** markets, with traders of different categories.

Each deal should follow a "recipe" specifying the participating traders' categories.

Our mechanism allows multiple recipes in a ***tree structure***.

It is ***truthful*** and ***strongly-budget-balanced***.

It attains ***asymptotically-optimal gain-from-trade***.



Algorithm 1 Ascending prices mechanism — recipe-tree.

Input: A market N, a set of categories G and a recipe-tree R.
Output: Strongly-budget-balanced trade.

1. *Initialization:* Let $M_g := N_g$ for each $g \in G$.
   Determine initial price-vector $\mathbf{p}$:
   For each non-leaf g, set $p_g := -V$;
   For each leaf g, set $p_g := -V \cdot (\text{MaxDepth} - \text{Depth}(g) + 1)$;
2. Using Algorithm 2, select a set $G^* \subseteq G$ of categories.
3. For each $g^* \in G^*$, ask each agent in $i \in M_{g*}$ whether $v_i > p_{g*}$.
   (a) If an agent $i \in M_{g*}$ answers "no", then — remove i from M_{g*} and go back to step 2.
   (b) If all agents in M_{g*} for all $g^* \in G^*$ answer "yes", then — for all $g^* \in G^*$, let $p_{g*} := p_{g*} + 1$.
   (c) If after the increase $\sum_{g \in G} p_g \cdot r_g = 0$ for some $\mathbf{r} \in R$, then go on to step 4; else go back to step 3.
4. Determine final trade using Algorithm 3.

Algorithm 2 Given a recipe-tree, find a set of prices to increase.

Input: A set of categories G,
a set of remaining traders M_g for all $g \in G$,
and a recipe-tree R based on a tree T.
Output: A subset of G denoting categories whose price should be increased.

0. *Initialization*: For each category $g \in G$, let $m_g := |M_g|$ = the number of agents of N_g who are in the market.
1. Let g_0 be the root category. Let $c_{g_0} := \sum_{g' \text{ is a child of } g_0} m_{g'}$.
2. If $m_{g_0} > c_{g_0}$ [or g_0 has no children at all], then return the singleton $\{g_0\}$.
3. Else ($c_{g_0} \geq m_{g_0}$), for each child g' of g_0:
   Recursively run Algorithm 2 on the sub-tree rooted at g';
   Denote the outcome by $I_{g'}$.
   Return $\bigcup_{g' \text{ is a child of } g_0} I_{g'}$.

Category	Agents' values
N_1: buyers	**17, 14, 13, 9,** 6, 2
N_2: sellers	**-4, -5,** -8, -10
N_3: A-producers	**-1, -3,** -5
N_4: B-producers	**-1, -4,** -6

Recipe-tree:
$R = \{(1,1,0,0), (1,0,1,1)\}$

Category counts	G^*	Price-increase stops when	New prices	Price-sum
6, 4, 3, 3	2, 4	B-producer −6 exits	$-V, -V-6, -V, -6$	$-2V-6$
6, 4, 3, 2	2, 3	A-producer −5 exits	$-V, -11, -5, -6$	$-V-11$
6, 4, 2, 2	2, 4	seller −10 exits	$-V, -10, -5, -5$	$-V-10$
6, 3, 2, 2	1	buyer 2 exits	2, −10, −5, −5	−8
5, 3, 2, 2	2, 4	B-producer −4 exits	2, −9, −5, −4	−7
5, 3, 2, 1	2, 3	seller −8 exits	2, −8, −4, −4	−6
5, 2, 2, 1	1	buyer 6 exits	6, −8, −4, −4	−2
4, 2, 2, 1	2, 3	A-producer −3 exits	6, −7, −3, −4	−1
4, 2, 1, 1	1	price-sum crosses zero	7, −7, −3, −4	0