Multi-item diffusion auctions

Haolin Liu

September 28, 2021

Generalized information diffusion mechanism (GIDM)

1. Find the optimal allocation in the whole social network
2. Construct diffusion critical tree for these optimal buyers
3. Allocate item from top to the bottom based on $SW_{-C_i^{\mathcal{K}}}$ (fix allocated items, fix optimal allocation for other branches)
4. $p_i = SW_{-D_i} - (SW_{-C_i^{\mathcal{K}}} - v_i')$ if i is winner
   $p_i = SW_{-D_i} - SW_{-C_i^{\mathcal{K}}}$ if i is not winner

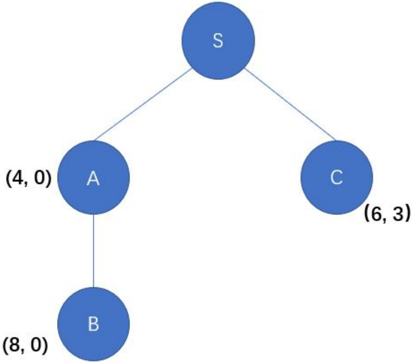
Zhao, D.; Li, B.; Xu, J.; Hao, D.; and Jennings, N. R. 2018. Selling Multiple Items via Social Networks. In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, 68–76.

Distance-based Network auction mechanism

1. Order every buyer based on their shortest distance to the seller, buyers with shorter distance has higher priority.
2. Initialize $K' = K$. Traverse all buyers based on their priority, if $v'_i \geq v^{K'}_{-D_i}$ (fix allocated items), allocate one item to i, $K' = K' - 1$.
3. $p_i = v^{K'}_{-C(i)}$ if i is winner
   $p_i = 0$ if i is not winner

Kawasaki, T.; Barrot, N.; Takanashi, S.; Todo, T.; and Yokoo, M. 2020. Strategy-Proof and Non-Wasteful Multi- Unit Auction via Social Network. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, 2062–2069.

A counter example for more general settings



Assume true value of C is $(6, 3)$, seller has two items.
If C reports truthfully, allocation is A and B, $u_c = 0$.
If C misreports to $(6, 5)$, allocation is B and C, $u_c = 6 - 4 = 2$.

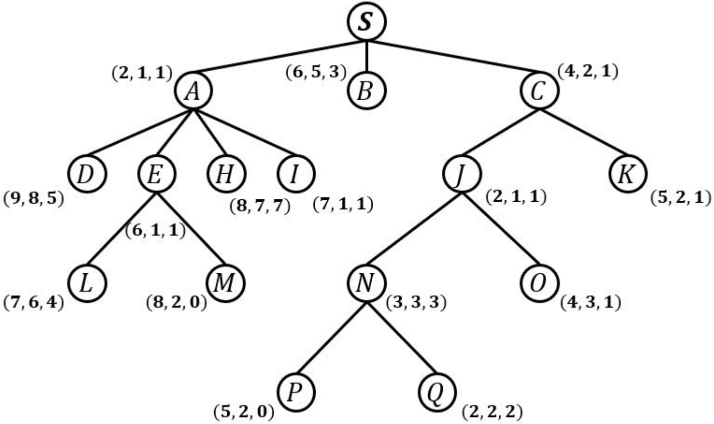
Misreport to $(6, 5)$ is equal to diffuse information to a buyer with value $(5, 0)$.

Layer-based diffusion mechanism (only use two layers)

First we design mechanism on trees, a simple mechanism for multi-unit auction with buyers with diminishing marginal value.

1. Divide buyers into different layers based on their shortest distance(d_i) to the seller
2. For every buyer in the first layer, remove their top $\mathcal{K}$ ranked children based on their report value for the first item. Besides, remove all buyers below the second layer. Compute allocation that maximizes social welfare in the remaining buyers (SW_{-R_1}), such allocation to buyers in the first layer is their final allocation.
3. Remove all buyers below the second layer, fix allocation in the first layer. Compute allocation that maximizes social welfare in the remaining buyers (SW_{-R_2}), such allocation to buyers in the second layer is their final allocation.
4. $p_i = SW_{-D_i} - (SW_{-R_{d_i}} - v_i(\pi_i))$ where $D_i = R_{d_i} \cup C_i$

An example



There are three items.
Allocate two items to B, one item to D.
$p_B = 6 + 4 + 2 - (6 + 5 + 6 - 6 - 5) = 6$,
$p_D = 6 + 5 + 8 - (6 + 5 + 9 - 9) = 8$.
A gets a reward.
$p_A = 6+5+4-(6+5+6)$
$=$ -2

Why it works?

- Remove all potential winners, if a potential winner misreports and is still a potential winner, then the set of buyers who are not potential winners will not change.
- If a buyer pretend to be a potential winner by misreporting, his utility will not increase.
- There is no potential contribution buyers.

Finding a reasonable potential winner set can extend this method to combinatorial auctions. But it is challenging.

Challenge to utilize full social network

There are more misreport cases.

- Buyers in the second layer may misreport to affect allocation of the first layer. In that case, items may "flow into" deeper layer and buyers in the second layer will get more rewards.
- Buyers in the third layer may misreport to affect allocation of the first layer.
- ......

A reasonable mechanism should remove all potential winners and potential contribution buyers in deeper layers.

One method to remove buyers

When computing allocation of buyers in layer q:

- For every buyer $i \in L_q$, in C_i:
  1. Remove buyers with top $\mathcal{K}$ largest value of the first item. ($C_i^{\mathcal{K}}$)
  2. Remove buyers who have children.($C_i^{\mathcal{P}}$) Assume all $|C_i^{\mathcal{P}}| \leq \mu$.
  3. Remove buyers with top x largest value of the first item in $C_i \setminus (C_i^{\mathcal{K}} \cup C_i^{\mathcal{P}})$. ($H_i^x$)

  where $x = \mu - |C_i^{\mathcal{P}} \setminus C_i^{\mathcal{K}}|$.
  Let $C_i^{\mathcal{A}} = C_i^{\mathcal{K}} \cup C_i^{\mathcal{P}} \cup H_i^x$
- Remove all buyers in and below $q+2$. ($\bigcup_{k \geq q+2} L_k$)

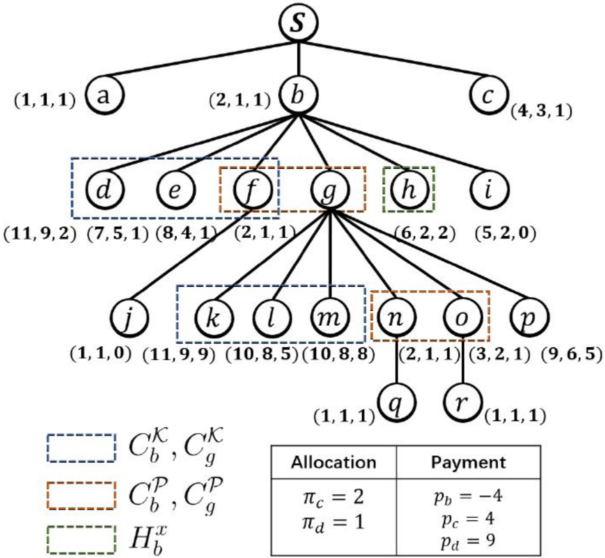
Let $R_q = \bigcup_{i \in L_q} C_i^{\mathcal{A}} \cup \bigcup_{k \geq q+2} L_k$

Layer-based diffusion mechanism on trees (LDM-Tree)

For buyers in layer $q = 1, ..., q^{max}$:

- Remove buyer set R_q and fix allocation of upper layers.
- Find optimal allocation π^q in the remaining buyers, compute SW_{-R_q}. For buyer $i \in L_q$, π_i^q is the final allocation.
- For every buyer $i \in L_q$, compute SW_{-D_i}(where $D_i = R_q \cup C_i$), $p_i = SW_{-D_i} - (SW_{-R_q} - v_i(\pi_i^q))$

An example



Allocation	Payment
$\pi_c = 2$ $\pi_d = 1$	$p_b = -4$ $p_c = 4$ $p_d = 9$

How to generalize this mechanism into graphs

Algorithm 2: Layer-based Transformation

Input: A graph $\mathcal{G}$;
Output: A tree $\mathcal{T}^{trans}$;

```
Initialize T^trans = G;
Run breadth first search on G to get layers L_1, ..., L_{l^max};
for l = 1, 2, ..., l^max in T^trans do
    Remove all edges between the nodes in layer L_l;
    if l >= 2 then
        for all i ∈ L_l do
            Find all edges (i, j) where j ∈ L_{l-1}. Randomly
            remain one edge and remove the others.
        end for
    end if
end for
Return T^trans.
```

Algorithm 3

Input: A report profile $\hat{\theta}$ and a diffusion mechanism $\mathcal{M}$ for trees;
Output: $\pi(\hat{\theta})$ and $p(\hat{\theta})$;

- Construct the graph $\mathcal{G}(\hat{\theta})$;
- Run algorithm 2 and the return is $\mathcal{T}^{trans}(\hat{\theta})$;
- Run $\mathcal{M}$ on $\mathcal{T}^{trans}(\hat{\theta})$;
- Return $\pi_i(\hat{\theta})$ and $p_i(\hat{\theta})$ for each buyer i.

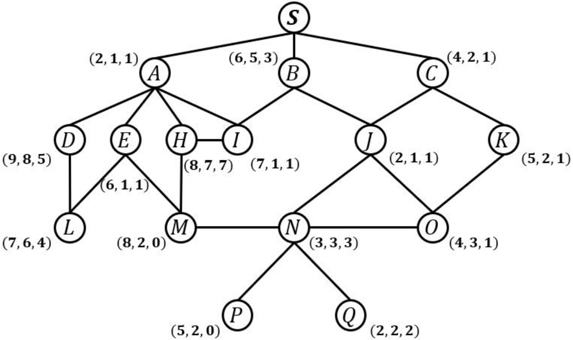
How to generalize this mechanism into graphs

Theorem

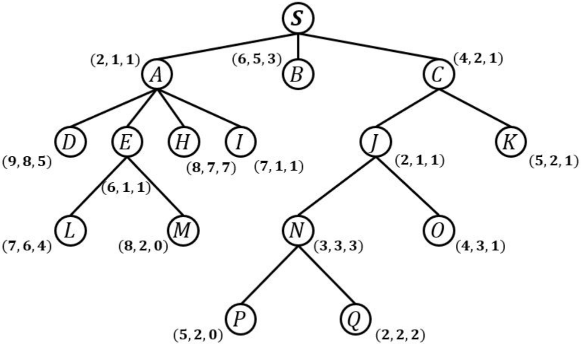
If the input mechanism $\mathcal{M}$ for algorithm 3 is IC and for buyers i, j in the same layer, u_i is non-increasing when j has more children, then algorithm 3 is IC.

This theorem simplifies some mechanism design problems on graphs.
LDM-Tree satisfies this theorem. Thus if $\mathcal{M}$ is LDM-Tree, algorithm 3 is IC.

An example



S
A (2,1,1)
B (6,5,3)
C (4,2,1)
D (9,8,5)
E (6,1,1)
H (8,7,7)
I (7,1,1)
J (2,1,1)
K (5,2,1)
L (7,6,4)
M (8,2,0)
N (3,3,3)
O (4,3,1)
P (5,2,0)
Q (2,2,2)



S
A (2,1,1)
B (6,5,3)
C (4,2,1)
D (9,8,5)
E (6,1,1)
H (8,7,7)
I (7,1,1)
J (2,1,1)
K (5,2,1)
L (7,6,4)
M (8,2,0)
N (3,3,3)
O (4,3,1)
P (5,2,0)
Q (2,2,2)

Diffusion auction design for other settings

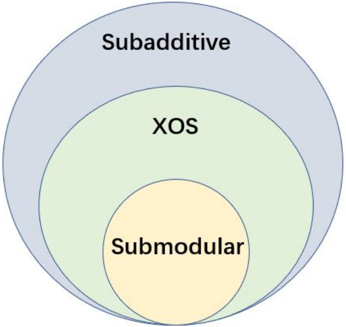
1. Multiple homogeneous items with unit-demand buyers:
   Every buyer has a single value, there is a sufficient and necessary condition for IC mechanisms.
   - $\pi_i(v_i^1, r_i) \geq \pi_i(v_i^2, r_i)$ where $v_i^1 \geq v_i^2$
   - If $v_i^{critical}(r_i') \neq \infty$, then $v_i^{critical}(r_i) \neq \infty$ where $r_i' \subseteq r_i$.
   - $\tilde{x}_i$ and $\bar{x}_i$ are bid-independent.
   - $\tilde{x}_i(r_i) - \bar{x}_i(r_i) = v_i^{critical}(r_i)$ for $r_i \supseteq r_i^{critical}$
   - $\tilde{x}_i(r_i)$ is diffusion-monotonic for $r_i \supseteq r_i^{critical}$, $\bar{x}_i(r_i)$ is diffusion-monotonic

Diffusion auction design for other settings

2. Combinatorial auction with single-minded buyers:
   - Known single-minded buyers.
     The same sufficient and necessary condition as above.
   - Unknown single-minded buyers.
     There is one more dimension, buyers may misreport bundle they want. However, buyers' value space is still one-dimensional, similar condition can be constructed.

Diffusion auction design for other settings

3. Combinatorial auction with quasi-linear utility buyers:
   - Submodular: $v(S \cup T) + v(S \cap T) \leq v(S) + v(T)$
   - XOS: There are t additive value functions $\{a_1, \cdots, a_t\}$ such that $v(S) = \max_{r \in [t]} a_r(S)$
   - Subadditive: $v(S \cup T) \leq v(S) + v(T)$



Some simple mechanisms for combinatorial auction

1. Sell items separately:
   - For homogeneous items, not IC when running IDM multiple times.
   - For heterogeneous items, IC when running IDM multiple times.

Some simple mechanisms for combinatorial auction

2. Fix price auction:
   - Sort buyers. If $d_i > d_j$, $i \succ j$. If $d_i = d_j$ and $|r_i| > |r_j|$, $i \succ j$. (with random tie-breaking)
   - Set a fixed price for each item.
   - Buyers come to select the remaining items according to their priority.(demand oracle)

IC, IR, WBB, polynomial time.
How to set fixed price?

Some simple mechanisms for combinatorial auction

3. "Max" mechanism:
   - Sort all buyers according to their distance.
   - Initialize $M' = M$. For the sorted buyers, if $v_i' \geq v^{max}_{N\backslash K_i \cup \{i\}}$, allocate S_i to i where S_i is the bundle in M' that can maximize utility of i, update $M' = M \backslash S_i$.
     The payment of i is $\tilde{x}_i(r_i) = v^{max}_{N\backslash K_i \cup C(i) \cup \{i\}}$
   - For the buyers who do not get any item: $\bar{x}_i(r_i) = v^{max}_{N\backslash K_i \cup C(i) \cup \{i\}} - v^{max}_{N\backslash K_i \cup \{i\}}$

Theorem

If K_i satisfies the following properties, then above mechanism is IC:

1. If $N' \subseteq N$, then $N' \backslash K_i \cup \{i\} \subseteq N \backslash K_i \cup \{i\}$
2. For buyer i, j, if $j \in K_i$ when diffusing information to r_j', then $j \in K_i$ when diffusing information to r_j where $r_j' \subseteq r_j$
3. For $j \in K_i$, $d_j > d_i$