Under Guidance of
Dr. Sunil Simon
IIT, KANPUR

ALGORITHMIC GAME THEORY (CS656A)

Paper Review : Efficient reallocation under additive and responsive preferences.

Harsh Agarwal
21111030
18th November 2021

OUTLINE

- Introduction
- Notations And Definitions
- Additive Utilities
- Ordinal Preference
- Conclusion
- Reference

INTRODUCTION

- Reallocation of resources in multi-agent system in a way that improves system's social welfare or results in a better outcome is the most basic and important concern.
- A well-known solution to the above-mentioned problem is finding a set of individually rational and Pareto Improvement assignments which converge to a Pareto Optimal solution.
- Finding Pareto Optimal assignment from scratch is trivial.
- So, we discuss ways to determine whether an initial assignment/ endowment is Pareto Optimal. If not, then we try to find a Pareto Optimal assignment which satisfies individual rationality.

INTRODUCTION

- Testing Pareto Optimality of an assignment is computationally difficult and so is computing an individually rational and Pareto Optimal assignment is also computationally difficult.

- For Cardinal Additive Utilities, a weak monotonicity assumption is made.

- For Ordinal Preference, a strong monotonicity assumption is made.

DEFINITIONS AND NOTATIONS

- Let set of agents be denoted by $N = \{1 \dots . n\}$
- Let set of objects be denoted by $O = \{1 \dots . m\}$
- An assignment $p = \big(p(1) \dots p(n)\big)$ is a partitioning of O into n subsets, where $p(i)$ denotes the set of objects assigned/endowed to agent i.
- For cardinal utilities, the utilitarian social welfare metric of an assignment p is defined as $SW(p) = \sum_{i \in N} u_i\big(p(i)\big)$
- An assignment $p \in \chi$ is said to be *individually rational* for an initial assignment $e \in \chi$ if $u_i\big(p(i)\big) \geq u_i\big(e(i)\big)$ holds for every agent i.
- p is said to be *Pareto Dominated* by q if (a) for every agent $i \in N$, $u_i\big(q(i)\big) \geq u_i\big(p(i)\big)$ is true and (b) for at least one agent $i \in N$, $u_i\big(q(i)\big) > u_i\big(p(i)\big)$ is valid.

ADDITIVE UTILITIES

1. Hard Cases
2. Constant No of agents and small utilities
3. Lexicographic Utilities
4. Two Utility Values
5. Conservative Pareto Optimality.

Hard Cases

Lemma 1: If there exists a polynomial-time algorithm to compute a *Pareto Optimal and individually rational* assignment, then there exists a polynomial-time algorithm to test *Pareto Optimality*.

Proof:

- Algorithm A which can compute individually rational and Pareto Optimal assignment in polynomial-time.
- p is Pareto Optimal, if $u_i\big(q(i)\big) = u_i\big(p(i)\big)$ for all $i \in N$
- p is not Pareto Optimal, if $u_i\big(q(i)\big) > u_i\big(p(i)\big)$ for at least one agent i.

Hard Cases

Theorem 1: Under additive preferences, testing *Pareto Optimality* of a given assignment is *weakly coNP-complete*, even for n =2 even if the induced ordinal preferences over individual objects are the same.

A Decision Problem *P* is said to belong to *coNP* complexity class if and only if its complement ($\overline{\mathrm{P}}$) is in the *NP* complexity class

Proof:

- Testing Pareto Optimality is in coNP, as it is possible to determine whether an assignment is Pareto Dominated in polynomial time.
- For completeness, we will reduce PARTITION problem into Testing Pareto Optimality problem

Hard Cases

- Instance of Partition Problem is defined as
  - A set of t elements $E = \{ e_1 \dots \dots e_t \}$
  - Integer weights w(e_j) for each element in E such that $\sum_{e_i \in E} w(e_i) = 2M$
- Problem is whether a balanced Partition S exists for E such that $\sum_{e_i \in S} w(e_i) = \sum_{e_i \in E \setminus S} w(e_i) = M$.

Hard Cases

- For Reduction
  - We assume a set of $t+1$ objects $\{g^{+}, g_{1}, \dots \dots, g_{t}\}$
  - Two agents {1,2}
  - Utility function:
    - For agent 1, $u_1(g^+) = M$ and $u_1(g_i) = w(e_i)$ for all $i \in \{1 \dots \dots t\}$
    - For agent 2, as $u_2(g^+) = M + \varepsilon$, with $0 < \varepsilon < 1$ and $u_2(g_i) = w(e_i)$ for all $i \in \{1 \dots \dots t\}$
  - Assignment p such that agent 1 gets gets g^+ and agent 2 gets all g_j objects is *Pareto Optimal* if and only if there is no balanced partition of E.

Hard Cases Example

Suppose $E = \{e_1, e_2, e_3\}$ and $M = 3$.

(i) Let weights be {1,2,3} then there exists a balanced partition $\{e_1, e_2\}$ and $\{e_3\}$ such that the sum of their weights is 3. So, if we consider above mentioned initial assignment that agent 1 gets g^+ and its utility is 3 and agent 2 gets $\{g_1, g_2, g_3\}$ and its utility is 6. A possible exchange is agent 1 gives g^+ in exchange for $\{g_3\}$ then utility of agent 1 remains the same but utility of agent 2 becomes $1 + 2 + 3 + \varepsilon = 6 + \varepsilon$, thus increasing and therefore above-mentioned assignment is not *Pareto Optimal*.

(ii) Let weights be {2,2,2} then there does not exist a balanced partition. So, if we consider above mentioned initial assignment that agent 1 gets g^+ and its utility is 3 and agent 2 gets $\{g_1, g_2, g_3\}$ and its utility is 6. We can see that for agent 1 to be *individually rational*, it needs to take 2 objects from agent 2 in exchange for g^+, which decreases the utility of agent 2 and is therefore not *individually rational,* and therefore above-mentioned assignment is *Pareto Optimal* assignment.

Constant number of agents and small utilities

Lemma 2: If there is a constant number of agents and the utilities are all integers, then the set of all vectors of utilities that correspond to an assignment can be computed in pseudo-polynomial-time

Proof:

Consider the below algorithm where 0^k $denotes$ $0, \dots \dots, 0$ k $times.$

1. L = $\{(0^n)\}$
2. for k = 1 to m do
    a. $L' = \{l + (0^{i-1}, u_i(o_j), 0^{n-i}) | i \in N ; l \in L \}$
    b. L = L'
3. End for
4. return L

Constant number of agents and small utilities

- Above algorithm runs in pseudo-polynomial-time as its complexity is $O(W^2 . n . m)$ where W is maximal welfare possible. No of utility vector in L can never exceed $(W+1)^2$
- Using mathematical induction on k, it can be proven that $l = (v_1, \dots \dots, v_n)$ can be achieved by assigning objects $o_1, \dots, o_k$ to the agents if and only if $l \in L$ after $o_1, \dots, o_k$ objects have been considered.
- $l' \in L$ after k+1th iteration if l' is obtained from l by adding $u_i(o_k)$ to the utility of some agent i.

Constant number of agents and small utilities

Theorem 2: If there is a constant number of agents and the utilities are all integers, then there exists a pseudo-polynomial-time algorithm to compute a *Pareto Optimal* and *individually rational* assignment.

Proof:

- We use algorithm described in **Lemma 2,** to calculate all possible utility vectors and now we also store partial assignment of objects to agents for each utility vector.
- At end of algorithm, we get a list of all possible utility vectors and corresponding assignments for each utility vector.
- Complexity of this process is $O(|L|^2)$ time and remaining vectors in L are *Pareto Optimal*.

Lexicographic Utilities

- A utility function is Lexicographic if for each agent $i \in N$ and each object $o \in O$, $u_i(o) > \sum_{o \succ_i o'} u_i(o')$ with the condition that $\sum_{o \in \emptyset} u_i(o) = 0$, which implies that $u_i(o) > 0$ for each o.
- Example: an agent with utilities (11,6,3,1)
- To test *Pareto Optimality* of an assignment p, we construct a graph called envy graph of p.
  - The vertices of this graph are one vertex for each object $o \in O$.
  - For each vertex associated with object o, set of edges are (o, o') for any object $o' \in O - \{o\}$ such that $o' \succcurlyeq_i o$, where i is the agent to whom object o is allotted in assignment p

Lexicographic Utilities

Theorem 3: An assignment p is not *Pareto Optimal* with respect to *Lexicographic* utilities if and only if there exists a cycle in $G(p)$ (envy graph of p) which contains at least one edge corresponding to a strict preference.

Proof:

- For right to left direction, let's assume there exists a cycle C that contains at least one edge corresponding to a strict preference. Then, the exchange of objects along the cycle by agents owning these objects corresponds to a *Pareto Improvement* and thus p is not *Pareto Optimal*.

Lexicographic Utilities

- For left to right direction, assume that p is not *Pareto Optimal* and let q_1 be an assignment that *Pareto Dominates* p.

- A sequence of agents $i_1, i_2, \dots, i_k$ and a sequence of objects $o_1, o_2, \dots, o_{k+1}$ such that each agent i_j gets object o_{j+1} in q_1 in exchange for the loss of o_j.

- Since we have a finite set of objects, there must exist $k \, and \, k'$ such that the sequence $o_k \rightarrow o_{k+1} \rightarrow \cdots \rightarrow o_{k'}$ forms a cycle.

- If there does not exist $l \in [k, k'-1]$ such that $o_{l+1} \succ_{i_l} o_l$ then we consider assignment q_2 derived from q_1 by reassigning every object o_{l+1} to agent i_l for $l \in [k, k'-1]$.

- There must exist some finite value t, for which there exists a $l \in [k, k'-1]$ such that $o_{l+1} \succ_i o_l$ for the cycle $o_k \rightarrow o_{k+1} \rightarrow \cdots \rightarrow o_{k'}$ founded in q_t.

Lexicographic Utilities

- After a finite number of steps, we would have $q_t(i) = p(i)$ for all $i \in N$.

- It is evident that envy graph can be constructed in linear time for any assignment p and search for a cycle containing at least one strict preference edge in $G(p)$ can be found out in linear time by applying a graph traversal algorithm for each strict preference edge in $G(p)$. Thus, the complexity of testing *Pareto Optimality* of an assignment p is in linear time for *Lexicographic* utilities.

Two Utility Values

- Each agent uses only two utility values to show their preference over objects.

- A utility function profile u is bivalued if there exist only two values $\alpha > \beta \geq 0$ such that for every agent i and every object o, $u_i(o) \in \{\alpha, \beta\}$.

- For each agent, the set of objects O is divided into two subsets $E_i^1 = \{o \in O, u_i(o) = \alpha\}$ and $E_i^2 = \{o \in O, u_i(o) = \beta\}$.

- So, for an assignment q, $q^+(i) = q(i) \cap E_i^1$ and $q^-(i) = q(i) \cap E_i^2$

Two Utility Values

Lemma 3: If an assignment p is *Pareto Dominated* by an assignment q then $|\bigcup_{i\in N} q^+(i)| > |\bigcup_{i\in N} p^+(i)|$

Proof:

- Done using contradiction
- Assume $|\bigcup_{i\in N} q^+(i)| \leq |\bigcup_{i\in N} p^+(i)|$ holds if assignment p is *Pareto Dominated* by an assignment q
- Social welfare for assignment q is $SW(q) = |\bigcup_{i\in N} q^+(i)|\alpha + |\bigcup_{i\in N} q^-(i)|\beta = |\bigcup_{i\in N} q^+(i)|(\alpha - \beta) + |O|\beta.$
- Similarly, $SW(p) = |\bigcup_{i\in N} p^+(i)|(\alpha - \beta) + |O|\beta.$
- $SW(p) \geq SW(q)$, but this contradicts the assumption that q *Pareto Dominates* p.

Conservative Pareto Optimality

An assignment p is *Conservatively Pareto Optimal* if there does not exist another assignment q that *Pareto Dominates* p and $|q(i)| = |p(i)|$ for all $i \in N$. It is applicable in many scenarios where the number of objects initially endowed to each agent needs to be conserved.

Lemma 4: There exists a polynomial-time algorithm to test *Pareto Optimality if and only if there exists a polynomial-time algorithm to test Conservative Pareto Optimality.*

Proof:

- For left to right direction, for each agent and object, we update the utility function from $u_i(o)$ to $u_i(o) + C$, where $C = m * \max_{i \in N, o \in O} u_i(o)$.
- In any *Pareto Improvement,* no agent will get lesser number of objects than it received in the original endowment.

Conservative Pareto Optimality

- By simply modifying the utility function as mentioned above, a polynomial-time algorithm to test *Pareto Optimality* can be used to test *Conservative Pareto Optimality*.

- For the right to left direction, let's say there are n agents, m objects, u utility matrix, and an assignment p.

- Define $(n-1)m$ dummy objects that each agent values at 0.

- A new assignment q is derived from p by giving $m - |p(i)|$ objects to agent i.

- Then q is *Conservative Pareto Optimal* for the modified instance if p is *Pareto Optimal* for the original instance.

Ordinal Preference

- We consider agents have additive cardinal utilities but only their ordinal preferences over objects are known by the central authority.
- Assumed that $u_i(o) > 0$ for $i \in N$ and $o \in O$.
- An assignment p is *Possibly Pareto Optimal* with respect to preference profile $\succcurlyeq$ if there exists $u \in U(\succcurlyeq)$ such that p is *Pareto Optimal* for u
- An assignment p is *Necessarily Pareto Optimal* with respect to preference profile $\succcurlyeq$ if for all $u \in U(\succcurlyeq)$ p is *Pareto Optimal* for u.
- *Necessary Pareto Optimality* implies *Possible Pareto Optimality.*
- At least one *Necessarily Pareto Optimal* assignment exists in which all objects are given to one agent.

Ordinal Preference

Theorem 4: A assignment is (1) *Possibly Pareto Optimal* if and only if (2) there exists no cycle in $G(p)$ which contains at least one edge corresponding to a strict preference if and only if (3) it is *Pareto Optimal* under *Lexicographic* utilities.

Proof:

- Already proved (2) $\Leftrightarrow$ (3) in ***Theorem 3***.
- We can also infer (3) $\Rightarrow$ (1) using the definition of *Possibly Pareto Optimal*
- To show (1) $\Rightarrow$ (2), Suppose p is not *Pareto Optimal* with respect to *Lexicographic* utilities, then by ***Theorem 3***, $G(p)$ contains a cycle which contains at least one edge corresponding to a strict preference.
- Assignment q is a *Pareto Improvement* over p with respect to all utilities consistent with the ordinal preferences.
- p is not *Possibly Pareto Optimal*.

CONCLUSION

From a computational point of view, *Pareto Optimality* in resource reallocation under additive utilities and ordinal preferences was studied and the paper came up with various *characterization theorems* and *polynomial-time algorithms* to solve problems of testing *Pareto Optimality* under various conditions.

REFERENCE

- **Haris Aziz, Péter Biró, Jérôme Lang, Julien Lesca, Jérôme Monnot, Efficient reallocation under additive and responsive preferences.**

THANK YOU