UGP REPORT
LOCAL ENVY FREE ALLOCATIONS

Submitted by:
Jatin Jindal
Roll No: 160308
Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur

Under the supervision of:
Sunil Simon
Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur

September 22, 2019

1 Abstract

We considered the problem of fair allocation of resources among agents where each agent is assigned one item. While various notions of fairness like envy-freeness, proportionality and maximin share guarantee are well studied in the literature, we focus on the measure of envy-freeness. We consider the setting where the agents form a network and the notion of envy is based on agents' neighbourhood. It is known that even for sparse graphs, and for regular graphs of degree n-3 the problem of determining the existence of local envy-free allocations is NP-hard. We consider the setting where the preference ordering have restricted structure. We show that if agents' preferences are single peaked or local single peaked, then there is an efficient procedure to determine the existence of a local envy-free allocation.

2 Introduction

Fair division problems have attracted the attention of various researchers in the last decades. The objective in fair division is to allocate a set of resources to a set of agents in a way that leaves every agent satisfied, to the extent that is feasible. To model the preferences of the involved agents, the standard assumption is that every agent is associated with a valuation function on the set of resources. Under this setup, various notion of fair allocation has been proposed, including e.g., proportionality and envy-freeness, along with several variants, strengthenings and relaxations. Also, the resources are categorised into 2 categories: infinitely divisible and indivisible goods.

In this report, we consider the problem of allocating the indivisible items such that each agent gets only one object. In the presence of indivisible goods, we cannot guarantee existence anymore for most fairness concepts like envy-free, and it is even NP-hard to decide whether there exists an envy-free allocation. So, the recent part of the research is now shifted to restrict the notion of envy-free and the one most popular is local envy-free. There has been several extensions of this problem in different kinds of graph, but we will consider the well know extension in which agents are present in a connected network graph. We also assume that agents have a limited information, i.e, they can get the info for only those agents who are directly connected to them. Our aim in this report is to find the fair allocation for this setting. We will try to find the local envy free allocation where envy means that the agent prefers the object of some other agents over her own and the local envy free means that the agent prefers the object of some neighbour over his own object. Note that these assumptions doesn't make sense in a completely connected network, because then each must get his top preferred object. But suppose if this is not the case, an agent is connected to n-k neighbours, then an agent will get an object from his top-k preference.

But even in such a restricted concept of local envy-free, the hardness result of finding the local envy-free allocation still exists. Motivated by this hardness result, we tried impose further restriction, this time on the agents preferences. We assumed that agent preferences are single peaked [2]. The single peaked preferences is the well known assumption used in the voters preferences to decide the Condorcet winner. Single peaked ordering denotes the ordering over the candidates such that there is a top-preferred object available in the ordering for each agent, and each agent preferences strictly decreases as he goes away from his top preferred object in the ordering. For eg: Consider the agents preferences over worked hours, and thus over earned money, are assumed to be single-peaked. This means that each agent has a most preferred amount of hours (equivalently money) to work, and if it happens that he has to work more than this preferred amount he wishes to deviate as less as possible. Similarly, if it happens that he has to work less than this preferred amount he wishes to be as close as possible to his preferred amount. In this report, we showed the algorithm for finding the local envy free allocation in polynomial time when the agents preferences are restricted to single peaked. Further, we showed the positive results by weakening the single peak preference to local single peaked preferences.

3 Model

Let N denotes the set of agents, O denotes the set of objects. Each i has a preference relation $\succ_i$ over O. The profile of preference relations is denoted by $\succ$. G = (N,E) is an un-directed graph with vertex set N and edge E. Each edge in E represents a social relation between the corresponding agents. Let N_i denotes the neighbour of the agent i i.e. $N_i = \{j : (i,j) \in E\} \cup \{i\}$. So, the problem is thus described by the tuple $e =< N, O, \succ, G = (N, E) >$.

4 Local envy-free allocation(LEF)

The first problem we will look over here is to find the computational complexity for checking the existence of LEF.

Problem 1.[DEC LEF]: *To check whether there is a local envy-free allocation*

It turns out that DEC LEF is computationally hard even for sparse graphs and for the regular graphs with degree n-3. But for regular graphs with degree n-2, we can solve this problem in polynomial time. We will look over the proof sketch of each of these cases.

Theorem 1. *In the sparse graphs where each agents has only one neighbor in G, DEC-LEF is NP Hard*

Proof. The reduction is from 3SAT. We are given a set of clauses $C = \{c_1, c_2, .., c_m\}$ defined over a set of variables $X = \{x_1, x_2, ..x_p\}$. Each clause is disjunctive and consists of 3 literals. The problem is to find out a truth assignment which satisfies all the clauses? So given an instance I = (C,X) of 3SAT, we will create an instance of DEC-LEF as follows:

First of all we have 2 agents for each clause c_j which are connected and whose ordering is of this form:

$$q_j > \text{(the 3 objects related to the literals of } c_j) > \text{rest}$$
$$\text{(the 3 objects related to the literals of } c_j) > q_j > \text{rest}$$

where "rest" means the remaining objects. The only envy free allocation will be to assign one agent q_j and assign the other, one of the 3 objects related to the literals of c_j. This is basically equivalent to finding the literal which must be true in each clause.

Note that right now we have the created the object for each literal and one for its negation form in each clause. Now we want to ensure its consistency i.e., each positive literal has the same value and each negative literal has the same value and both these values are different.

To satisfy these conditions, we created the agents ordering as follows: For each literal x_i :

$$
\begin{aligned}
X_{i1} :& u_i^1 > t_{i1} > \bar{u}_i^1 > t_{i2} > rest_i^1 \\
X'_{i1} :& t_{i1} > u_i^1 > t_{i2} > \bar{u}_i^1 > rest_i^1 \\
\\
X_{i2} :& u_i^2 > t_{i2} > \bar{u}_i^2 > t_{i3} > rest_i^2 \\
X'_{i2} :& t_{i2} > u_i^2 > t_{i3} > \bar{u}_i^2 > rest_i^2 \\
& . \\
& . \\
X_{im} :& u_i^m > t_{im} > \bar{u}_i^m > t_{i1} > rest_i^m \\
X'_{im} :& t_{im} > u_i^m > t_{i1} > \bar{u}_i^m > rest_i^m
\end{aligned}
$$

Note that this cyclic structure forces that either the agent are allocated all the objects u_i, t_i or all are allocated $\bar{u}_i, t_i$. This means that agents corresponding to clauses(which are defined at top of proof) cannot get both pos and neg literal, because otherwise there will be no envy free allocation of the X_i agents. Note that agent X_{ij} is connected to only agent X'_{ij}

Now, the only thing left is to assign the remaining objects(which are the left out literals) to the remaining agents. This can be done easily. Create the dummy objects equal to the number of remaining objects. Create the agents ordering as follows:
Dummy Objects > Remaining Literal Objects > Rest($t_i s, q_i s$)
Remaining Literal Objects > Dummy Objects > Rest($t_i s, q_i s$)

So, each agent will get one dummy objects and one remaining literal object. Hence, eventually all the objects will be allocated and the local envy free allocation exists iff the problem is SAT.

□

Theorem 2. *Dec LEF is NP-complete in regular graphs of degree n-3.*

Proof. Here, we will only show the top 3-preference of the agent since an agent can be local envy-free only if he gets an object from his top-3 preference.

We will consider the complement of this graph i.e., we will consider $\bar{G}$ where each node will have 2 and the graph will be a collection of cycles.

We will prove its hardness by reducing it from (3,B2)-SAT [25] which is a restriction of 3SAT where each literal appears exactly twice in the clauses, and therefore, each variable appears exactly four times.

For each clause c_i we introduce dummy clause-objects d_i^1 and d_i^2 , as well as a cycle in G containing clause-agents K_i^1, K_i^2, K_i^3 . The preferences of clause-agent K_i^j are:

$K_i^j : d_i^1 > d_i^2 > l(i,j) > ...$

where l(i,j) is the literal object corresponding to the i^{th} literal of c_j. This corresponds to the condition that at least one literal will be true.

Now we want the literal value to be consistent for each 2 cases and only one of the pos and neg literal should be satisfied.

For each variable x_i , we introduce dummy variable-objects q_i^1 and q_i^2 and literal-objects u_i^1 , u_i^2 , $\bar{u}_i^1$ and $\bar{u}_i^2$ corresponding to its first and second occurrence as an unnegated and negated literal, respectively, as well as a cycle in $\bar{G}$ containing literal-agents $X_i^1, \bar{X}_i^1, X_i^2 and \bar{X}_i^2$, connected in this order. Preferences are as follows:

$$
\begin{aligned}
X_i^1 &: q_i^1 > q_i^2 > u_i^1 > ..\\
\bar{X}_i^1 &: q_i^1 > q_i^2 > \bar{u}_i^1 > ..\\
X_i^2 &: q_i^2 > q_i^1 > u_i^2 > ..\\
\bar{X}_i^2 &: q_i^2 > q_i^1 > \bar{u}_i^2 > ..
\end{aligned}
$$

Note also that in any LEF allocation, either q_i^1 and q_i^2 are allocated to agents X_i^1 and X_i^2 , either q_i^1 and q_i^2 are allocated to agents $\bar{X}_i^2$ and $\bar{X}_i^1$. The first case can be interpreted as assigning false to x_i , and the later case as assigning true to x_i.

Now, to distribute the remaining the literal objects, for each literal we created the cyclic ordering using the dummy objects as follows for each $i \leq 4m - 1$ because before this 4m-1 objects were not allocated.

$$L_i^1 : t_i^1 > t_i^2 > h_{i-1}$$
$$L_i^2 : t_i^1 > t_i^2 > l(i)$$
$$L_i^3 : t_i^1 > t_i^2 > h_i$$

where h_0, h_m starts for h_1, h_{4m-1} resp.

□

Theorem 3. *Dec-LEF in graphs of minimum degree n-2 is solvable in polynomial time.*

Proof. Note that $\bar{G}$ is a matching in that case. In order to simplify notations, we denote by $\phi(i)$ the neighbor of agent i in G. We will reduce this problem to 2-SAT which is solvable in linear time. x_{ij} is true iff object j is assigned to agent i. o_i^j denotes the object at the position j in the preference relation of agent i.

Consider the following formula ψ:

$$\bigwedge_{i \in N} (x_{io_i^1} \cup x_{io_i^2}) \wedge \bigwedge_{i<l, 1 \le j \le n} (\neg x_{ij} \cup \neg x_{lj}) \wedge \bigwedge_{i \in N} (x_{io_i^1} \cup x_{\phi(i)o_i^1})$$

The first part of formula ψ expresses that each agent must obtain an object within her top-2. The second term denotes that every agent is owned by at most one agent and $|N| = |O|$. For local envy free, the last term is added. So, the top object of each agent i must be assigned to agent i or the edge in $\bar{G}$. Hence, formula ψ exactly translates the constraints of an LEF allocation. □

The above 3 theorems showed that for unrestricted profiles it is computationally hard to check the existence of LEF. Now, we will consider the setting in which agents preferences are restricted to single peaked.

Single Peaked: A profile is single peaked if $\exists \rhd$ over O such that the profile in single peaked over $\rhd$.

Single Peaked over $\rhd$: An ordering $\rhd$ over the objects O is single peaked over N if $\exists o^* \in O$ such that $\forall i \in N$, we have
(1) $a \rhd b \rhd o^* \rightarrow b \succ_i a$
(2) $o^* \rhd b \rhd a \rightarrow b \succ_i a$
(3) o^* is the most preferred object for agent i.

4.1 Local envy free allocation with single peak preferences

Theorem 4. *Given a regular graph G = (N,E) with each vertex degree = n-3 and the preference profile is* ***single peaked****, then there is an algorithm which can return the* ***local envy allocation****(if exists), otherwise return no such allocation in $O(n)$ time where $n = |N|$.*

Observations:
(1) Since the degree of each vertex is n-3, then an agent is local envy free only if he gets an object from his top 3 preferences. Therefore, we will focus only on the top 3 preferred objects of each agent.
(2) Since the profile is single peaked, lets say over $\triangleright$. Then the top 3 preferred objects of each agent forms an interval in the $\triangleright$.

Notations: Without loss of generality, assume that $\triangleright$ is $o_1, o_2, o_3, ..o_n$, and agents are single peaked over this $\triangleright$. Let $r_i(j)$ denotes the j^{th} preferred(ranked) object by agent i. So, $r_i(1), r_i(2), r_i(3)$ denotes the first, second and third preferred object of the agent i respectively. Let p_i denotes the position of the leftmost positioned object among the $r_i(1), r_i(2), r_i(3)$ objects in the ordering $\triangleright$. So, objects at $p_i, p_i + 1, p_i + 2$ will denotes his top-3 preferred objects and we will call it the 3-size window for agent i. Let $a_1, a_2, a_3, ...a_n$ denotes the set of agents. We define $A(i)$, the agents at position i, as the set of agents such that $\{a_j | p_j = i\}$. Therefore, the number of agents at position i is $|A(i)|$. We use stage i for the allocation when agents in $A(1) \cup A(2) \cup A(3).. \cup A(i)$ gets the envy free allocation.

Inference:

- if $|A(i)| > 3$, then there doesn't exist any envy free allocation. Since, all the agents in A(i) needs the objects from o_i, o_{i+1}, o_{i+2}, and by pigeon hole principal atleast one agent will not receive the object.
- If we find the envy free allocation for the agents in $A(1) \cup A(2) \cup A(3) \cup ... \cup A(i)$, then the agents in $A(1) \cup A(2) \cup ..A(i-2)$ are always local envy-free for any extension of this allocation since all the other agents($A(i+1), A(i+2), ..A(n)$ gets the object of lower preference than their own.

- For any i, the agent$A(i)$ can't be envy to other agent for the objects at position $1, 2, ..., i-1$. Hence, agents in $A(i+1) \cup A(i+2).. \cup A(n)$ will care only for the objects o_i on-wards.
- From previous 2 statements, given the allocation of agents $A(i-1), A(i)$ all the possible local envy free allocation of $A(1) \cup A(2) \cup A(3) \cup ... \cup A(i)$ comes in the same equivalence class. Hence, no of equivalence class is less than max number of possible different ways of allocation to agents $A(i-1) \cup A(i)$. Hence, number of equivalent classes is < 6 at each stage. This is the crux of our incremental algorithm that number of equivalent classes at each stage is O(1) i.e independent of n.

Algorithm:

- **Base Step:** Start with all possible local envy free allocation of agents $A(1) \cup A(2)$. Hence, we are initially at stage2.
- **Time complexity to shift from stage i to stage i+1:** Given an equivalent classes in stage i, time complexity to find the next stage equivalent classes for this case is equivalent to the number of ways of allocating objects to agents A(i+1) $<$ 3!. Hence time complexity to find the next stage equivalent class is $< 3! * 6 = 36$.
- We have to repeat the previous step from stage 2 and upto stage n. Overall time complexity is the time to reach the stage n. Hence, time complexity of the algorithm $< 36n = O(n)$.

Note: This algorithm is the foundation for its extensions to regular graph of degree n-k. Now we will show case to case analysis on the regular graph of degree n-3 with some optimization.

Now, we will show case to case analysis for the regular graph of degree n-3

Proof. Before beginning this, we will define the term configuration used in this algorithm. The configuration consists of 2 rows in which first row represent the remaining objects. $*$ means that this object is already allocated. The second row represents the remaining agents who has not been allocated any object so far. So, the final configuration is basically empty since all the agents are allocated the objects. Now, we will show the algorithm to find the action at a particular configuration.

Repeat these steps until you reach with stage = n. Suppose you are at stage i-1. Now consider the configurations:

Case 1: If the configuration at stage i-1 looks like this, i.e, agents $A(1) \cup A(2) \cup .. \cup A(i-1)$ are assigned the objects $o_1, o_2, o_3, ..o_{i-1}$.

o_i	o_{i+1}	....	o_n
$A(i)$	$A(i+1)$	....	$A(n)$

1.1. if $|A(i)| == 0$: report error, since object o_i will not be assigned to any agent.

1.2. if $|A(i)| == 1$: The only possible local envy free allocation(if exists) for object o_i is to assign the object to agent $A(i)$. The configuration changes to:

o_{i+1}	o_{i+2}	....	o_n
$A(i+1)$	$A(i+2)$	....	$A(n)$

1.3. if $|A(i)| == 2$: There will be 2 possibilities for object o_1. Try both these cases. Then the problem in each case reduces to configuration of the form in case 2.2. Run algo 1:

*	o_{i+1}	o_{i+2}	....	o_n
$\|A(i)\| = 1$	$A(i+1)$	$A(i+2)$	....	$A(n)$

1.4. if $|A(i)| == 3$: Try all the 6 possibilities of allocating the objects. Then, the configuration becomes equivalent to case 3.

*	*	o_{i+2}	....	o_n
$A(i+1)$	$A(i+2)$	$A(i+2)$	....	$A(n)$

Therefore, time complexity for case(1) per shift is O(1).

Case 2: The current configuration looks like this,

$*$	o_{i+1}	....	o_n
$A(i)$	$A(i+1)$	....	$A(n)$

2.1. if $|A(i)| == 0$: Then the problem reduces to case 1

o_{i+1}	....	o_n
$A(i+1)$	....	$A(n)$

2.2. if $|A(i)| == 1$: run algo 1. The current configuration looks like this:

$*$	o_{i+1}	....	o_n
$\lvert A(i)\rvert = 1$	$A(i+1)$	....	$A(n)$

2.3. if $|A(i)| == 2$: Agents at $A(i)$ should be allocated objects o_{i+1}, o_{i+2}. Then, we reach the configuration of case 3.

$*$	$*$	o_{i+3}	....	o_n
$A(i+1)$	$A(i+2)$	A(i+3)	...	$A(n)$

2.4. For any other value of $|A(i)|$ there is no local envy free allocation.

Therefore, time complexity for case(2) per shift is O(1).

Case 3: The current configuration looks like this,

$*$	$*$	o_{i+2}	....	o_n
$A(i)$	$A(i+1)$	$A(i+2)$	....	$A(n)$

3.1. if $|A(i)| == 0$: Then the problem reduces to case 2:

$*$	o_{i+2}	....	o_n
$A(i+1)$	$A(i+2)$	....	$A(n)$

3.2. if $|A(i)| == 1$: The problem size reduces by 1 because agent at a_1 should get object o_{i+2} and the remaining problem looks like this:

$*$	$*$	o_{i+3}	....	o_n
$A(i+1)$	$A(i+2)$	$A(i+3)$	....	$A(n)$

3.3. For any other value of a_1 there is no local envy free allocation. Therefore, time complexity for case(3) per shift is O(1).

Note that the total number of possible configurations $O(n)$ by our algorithm. If we used the bottom up dynamic programming algorithm by using the above steps, we can solve this problem in O(n). Overall time complexity is hence O(n).

Algo 1: The initial configuration(input) looks like this:

$*$	o_{i+1}	o_{i+2}	....	o_n
$\lvert A(i)\rvert = 1$	$A(i+1)$	$A(i+2)$	....	$A(n)$

Case 1.1: The prefix of the second row satisfies the regex(1 1*0). If we allocate the object o_{i+1} *to* $A(i)$ then the problem is reduced by size 1. If o_{i+2} is allocated to $A(i)$ then o_{i+1} must be allocated to $A(i+1)$. In this case the problem reduces by size 2. Using the DP algorithm, and storing the result at each step we can solve this problem in O(n) time. Finally, we end in case 1 configuration.

Case 1.2: Similar to above case, just the other thing is that regex is of form(11*2). The final configuration will look like case 3.

Case 1.3: Reports the error for all the remaining cases.
The amortized cost for the algorithm per shift is O(1).

□

Theorem 5. *Given a regular graph G = (N,E) with each vertex degree = n-k and the preference profile is single peaked, then there is an algorithm which can return the local envy free allocation(if exists), otherwise return no, i.e, no such allocation exists in $O(n)$ time where $n = |N|$.*

Observations:
(1) Since the degree of each vertex is n-k, then an agent is local envy free only if he gets an object from his top k preferences. Therefore, we will focus only on the top k preferred objects of each agent.

(2) Since the profile is single peaked, lets say over $\triangleright$. Then the top k preferred objects of each agent forms an interval in the $\triangleright$.

Notations: Without loss of generality, assume that $\triangleright$ is $o_1, o_2, o_3, ..o_n$, and agents are single peaked over this $\triangleright$. Let $r_i(j)$ denotes the j^{th} preferred(ranked) object by agent i. So, $r_i(1), r_i(2), r_i(3)$ denotes the first, second and third preferred object of the agent i respectively. Let p_i denotes the position of the leftmost positioned object among the $r_i(1), r_i(2), r_i(3), ..., r_ik$ objects in the ordering $\triangleright$. So, objects at $p_i, p_i + 1, p_i + 2, ..., p_i + k - 1$ will denotes the position of top-k preferred objects and we will call it the k-size window for agent i. Let $a_1, a_2, a_3, ...a_n$ denotes the agents. We define $A(i)$ agents at position i, as the set of agents such that $\{a_j | p_j = i\}$. Therefore, the number of agents at position i is $|A(i)|$. We denote stage i as the point when agents in $A(1) \cup A(2) \cup A(3).. \cup A(i)$ gets the envy free allocation.

Inference:

- if $|A(i)| > k$, then there doesn't exist any envy free allocation. Since, all the agents in A(i) needs the objects from $o_i, o_{i+1}, o_{i+2}, ..o_{i+k-1}$, and by pigeon hole principal atleast one agent will not receive the object.
- If we find the envy free allocation for the agents in $A(1) \cup A(2) \cup A(3) \cup ... \cup A(i)$, then the agents in $A(1) \cup A(2) \cup ..A(i-k+1)$ are always local envy-free because all the other agents get the object of lower preference than their own.
- The agents $A(i)$ can't be envy to other agent for the objects at position $1, 2, ..., i-1$. Hence, agents in $A(i+1) \cup A(i+2).. \cup A(n)$ will care only for the objects o_i on-wards.
- From previous 2 arguments, we can say that given the allocation of agents $A(i-k+2), A(i-k+3), ..A(i-1), A(i)$ all the possible local envy free allocation of $A(1) \cup A(2) \cup A(3) \cup ... \cup A(i)$ comes in the same equivalence class. Hence, no of equivalence class $< k^{k-1}$ at each stage. This is the crux of our incremental algorithm that number of equivalent class is independent of n.

Algorithm:

- **Base Step:** Start with all possible local envy free allocation of agents $A(1) \cup A(2)$. So, we starts with stage-2 of allocation.

- **Time complexity to shift from stage i to stage i+1:** Given the equivalent classes at stage i, we need to find the equivalent classes at stage i+1. For a given stage-i equivalent class, to find the stage i+1 equivalence class, we just need to try all the ways of assigning objects to agents in A(i+1) which is $< k!$. Hence time complexity to shift stage $= < k^{k-1} * k!$.

- Total time complexity is the time to reach the stage-n starting from stage-2 and repeated using the previous step. Hence, overall time complexity $< nk^{k-1}k!$.

Corollary: The same algorithm will also works for graphs with minimum degree n-k with same time complexity.

In the remaining part, we will try to relax the restriction preference from Single Peaked to Local Single Peaked.

Locally single peaked: The problem $e =< N, O, \succ, G = (N, E) >$ is locally single peaked if $\forall i \in N$, there exists an ordering $\rhd_i$ over the objects for which the set N_i is single peaked.

Locally single peaked with fixed ordering: The problem e is locally single peaked with fixed ordering if there exists an ordering $\rhd$ over the objects O such that $\forall i \in N$, the set N_i is single peaked with respect to $\rhd$.

4.2 Local Envy Free Allocation with Local Single Peaked Preference

Theorem 6. *Given a regular graph G = (N,E) with each vertex degree = n-k and the preference profile is local-single peaked, then there is an algorithm which can return the local envy free allocation(if exists), otherwise return no such allocation in $O(n)$ time where $n = |N|$.*

Observations:

(1) Since the degree of each vertex is n-k, then an agent is local envy free only if he gets an object from his top k preferences. Therefore, we will focus only on the top k preferred objects of each agent.

(2) Consider an agent x Since the profile is local single peaked, the agents N_x(Neighbourhood of x) will be single peaked with respect to some ordering, lets say over $\triangleright$. Then the top k preferred objects of each agent in the N_x forms an interval in the $\triangleright$ ordering.

Notations: Without loss of generality, assume that $\triangleright$ is $o_1, o_2, o_3, ..o_n$, and agents in the neighbourhood(N_x) are single peaked over this $\triangleright$. So, n-k agents are single peaked with respect to this ordering. Let $r_i(j)$ denotes the j^{th} preferred(ranked) object by agent i. So, $r_i(1), r_i(2), r_i(3)$ denotes the first, second and third preferred object of the agent i respectively. Let p_i denotes the position of the leftmost positioned object among the $r_i(1), r_i(2), r_i(3), ..., r_i k$ objects in the ordering $\triangleright$. So, objects at $p_i, p_i + 1, p_i + 2, ..., p_i + k - 1$ will denotes the position of top-k preferred objects and we will call it the k-size window for agent i. Let $a_1, a_2, a_3, ...a_n$ denotes the agents. We define $A(i)$ agents at position i, as the set of agents such that $\{a_j \notin N_x | p_j = i\}$. Therefore, the number of agents at position i is $|A(i)|$. We denote stage i as the point when agents in $A(1) \cup A(2) \cup A(3).. \cup A(i)$ gets the envy free allocation.

Inference:

- if $|A(i)| > k$, then there doesn't exist any envy free allocation. Since, all the agents in A(i) needs the objects from $o_i, o_{i+1}, o_{i+2}, ..o_{i+k-1}$, and by pigeon hole principal at least one agent will not receive the object.
- n-k agents are single peaked with respect to that ordering. For the remaining k agents we will consider every possible allocation. Since, each agent should get an object from his top k preference, there total ways of assignment for them $= O(k^k)$.
- Now, for each possible assignment in the previous step, run this:
  If we find the envy free allocation for the agents in $A(1) \cup A(2) \cup A(3) \cup ... \cup A(i)$, then the agents in $A(1) \cup A(2) \cup ..A(i-k+1)$ are always local envy-free since other agents get the objects of lower preference than their own. The agents $A(i)$ can't be envy to other agent for the objects at position $1, 2, ..., i-1$. Hence, agents in $A(i+1) \cup A(i+2).. \cup A(n)$ will care only for the objects o_i on-wards. Hence, given the allocation of agents $A(i-k+2), A(i-k+3), ..A(i-1), A(i)$ all the possible local envy free allocation of $A(1) \cup A(2) \cup A(3) \cup ... \cup A(i)$ comes in the

same equivalence class. Hence, no of equivalence class $< k^{k-1}$ at each stage. This is the crux of our incremental algorithm that the size of the equivalent class is O(1).

Algorithm:

- **Base Step:** First of all give any random allocation for the agents not in N_x by assigning each agent the object in his top-k preference.
- Start with all possible local envy free allocation of agents $A(1) \cup A(2)$. So, we will be initially at stage-2.
- **Time complexity to shift from stage i to stage i+1:** Given the equivalent classes at stage i, we need to find the equivalent classes at stage i+1. Given an equivalent class in stage i, the extended equivalent class in stage i+1 corresponding to that can be found out by assigning objects to agents A(i+1) which is $< k!$. Hence total time to reach next stage is $< k^{k-1} * k!$.
- We need to reach stage n starting from stage-2 for each possible allocation in base step. If there is no local envy free allocation for all allocations in base step, then the algorithm returns: no envy free allocation possible. Otherwise it returns one of them satisfying it. Hence, overall time complexity $< nk^{k-1}k!$.

Corollary 1: The same algorithm will also works for graphs with minimum degree n-k and with local single peaked preference.

Corollary 2: If we careful look at the need of local-single peaked preference, you will see that we had used this only in the observation 2 and only for the one agent. This means that local-single peaked preference is even more restrictive. Any n-k agents which are single peaked will suffice for our results.

5 Conclusion

Hence we find that by restricting the preference domain to single peak or local single peak, we can change the hardness result of finding the local envy free allocation to polynomial time. Now, we are focusing to explore the more properties of local single peak preferences and its extensions.

References

[1] Beynier, Aurélie and Chevaleyre, Yann and Gourvès, Laurent and Lesca, Julien and Maudet, Nicolas and Wilczynski, Anaëlle `"Local Envy-Freeness in House Allocation Problems"` AAMAS 2018: 292–300

[2] Rediet Abebe, Jon M. Kleinberg and David C. Parkes `"Fair Division via Social Comparison"` CoRR 2016

[3] Edith Elkind and Martin Lackner and Dominik Peters *"CHAPTER 10 Structured Preferences"*, 2017

[4] Endriss, Ulle *"Trends in Computational Social Choice"*, 2017