Assigning Agents to Increase Network-Based Neighborhood Diversity

Zirou Qiu
University of Virginia
Charlottesville, VA, USA
zq5au@virginia.edu

Andrew Yuan
University of Virginia
Charlottesville, VA, USA
andrewyuan112@gmail.com

Chen Chen
University of Virginia
Charlottesville, VA, USA
chenannie45@gmail.com

Madhav V. Marathe
University of Virginia
Charlottesville, VA, USA
marathe@virginia.edu

S. S. Ravi
University of Virginia
Charlottesville, VA, USA
ssravi0@gmail.com

Daniel J. Rosenkrantz
University of Virginia
Charlottesville, VA, USA
drosenkrantz@gmail.com

Richard E. Stearns
University of Virginia
Charlottesville, VA, USA
thestearns2@gmail.com

Anil Vullikanti
University of Virginia
Charlottesville, VA, USA
vsakumar@virginia.edu

ABSTRACT

Motivated by real-world applications such as the allocation of public housing, we examine the problem of assigning a group of agents to vertices (e.g., spatial locations) of a network so that the *diversity level* is *maximized*. Specifically, agents are of two types (characterized by features), and we measure diversity by the number of agents who have at least one neighbor of a different type. This problem is known to be **NP**-hard, and we focus on developing approximation algorithms with provable performance guarantees. We first present a local-improvement algorithm for general graphs that provides an approximation factor of 1/2. For the special case where the sizes of agent subgroups are similar, we present a randomized approach based on semidefinite programming that yields an approximation factor better than 1/2. Further, we show that the problem can be solved efficiently when the underlying graph is treewidth-bounded and obtain a polynomial time approximation scheme (PTAS) for the problem on planar graphs. Lastly, we conduct experiments to evaluate the performance of the proposed algorithms on synthetic and real-world networks.

KEYWORDS

Agent assignment; diversity of allocation; combinatorial optimization; approximation algorithms; performance guarantee

ACM Reference Format:

Zirou Qiu, Andrew Yuan, Chen Chen, Madhav V. Marathe, S. S. Ravi, Daniel J. Rosenkrantz, Richard E. Stearns, and Anil Vullikanti. 2023. Assigning Agents to Increase Network-Based Neighborhood Diversity. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 9 pages.

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

1 INTRODUCTION

Many countries have public housing initiatives that offer low-income individuals secure and affordable residences. Housing options are typically allocated by government agencies that involve a process of *assigning applicants to vacant apartments* [14, 39]. Given that the applicants often come from a variety of demographic groups, the spatial distribution of public housing partially shapes the demographic structure of local communities [16, 35]. The promotion and cultivation of integrated communities is an objective of contemporary societies. It has been shown that integration can improve a country's financial performance, reduce the disparity between demographic groups, and advance social prosperity in general [9, 24, 36]. Conversely, segregated neighborhoods widen the socioeconomic divide in the population. As noted by many social scientists, residential segregation remains a persistent problem that directly contributes to the uneven distribution of resources and limited life chances for some groups (e.g., [33, 38, 40]).

In this work, we study the problem of promoting community integration (i.e., diversity) in the context of housing assignment. Indeed, public housing programs often take diversity into account. In Singapore, there are established policies to ensure that a certain ethnic quota must be satisfied for each project at the neighborhood level [10]. In the U.S., cities like Chicago and New York also place emphasis on the value of having integrated communities [8, 28]. Nevertheless, *formal computational methods* for improving the level of integration in the housing assignment process have received limited attention. Motivated by the above considerations, we investigate the problem of public housing allocation from an algorithmic perspective and *provide systematic approaches to design assignment strategies that enhance community integration.*

Formally, we model a housing project as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of vacant residences, and the edges in $\mathcal{E}$ represent proximity between residences. We are also given a set $\mathcal{A}$ of agents representing the applicants to be assigned to residences. Agents are partitioned into *two* demographic subgroups: type-1 and type-2. Without loss of generality, we assume that the number of type-1 agents does not exceed the number of type-2 agents. (We sometimes use the phrase "minority agents" for type-1 agents.) We also assume that the number of vacant residences (i.e., $|\mathcal{V}|$) equals the number of agents. Our goal is to construct an assignment (bijective mapping) $\mathcal{P}$ of residences to agents that maximizes the *the integration level* of the layout of agents on $\mathcal{G}$.

To quantify the integration level of a given assignment $\mathcal{P}$, we use the *index of integration* (IoA) metric proposed in [1]. This index is defined as the number of *integrated agents*, that is, agents with at least one neighbor of a different type in $\mathcal{G}$. An illustrative example is given in Fig. 1. We refer to the above assignment problem as Integration Maximization - Index of Agent Integration (IM-IoA). We note that this problem could also arise in other settings where integration is preferred, such as dormitory assignments for freshmen in universities [6].

The problem of maximizing IoA is known to be **NP**-hard [1]. Nevertheless, the authors of [1] did not address approximation questions for the problem, as their focus is on game theoretic aspects of IoA. In this work, we focus on developing approximation algorithms with provable performance guarantees for IM-IoA. Our main contributions are as follows.

- **Approximation for general instances**. We present a *local-improvement algorithm* that guarantees a factor 1/2 approximation. We further show that our analysis is tight by presenting an example that achieves this bound. While it is possible to derive an approximation for the problem using a general result in [7], the resulting performance guarantee is 0.356, which is weaker than our factor of 1/2.
- **Improved approximation for special instances**. For the case when the number of type-1 agents is a constant fraction α of the total number of agents, $0 < \alpha \leq 1/2$, we present a semidefinite programming (SDP) based randomized algorithm that yields approximation ratios in the range $[0.516, 0.649]$ for α in the range $[0.403, 0.5]$. For example, when $\alpha = 0.45$, the ratio is 0.578, and when $\alpha = 0.5$, the ratio is 0.649.
- **A polynomial time approximation scheme for planar graphs**. We present a *dynamic programming algorithm* that solves IM-IoA in polynomial time on graphs with bounded treewidth. Using this result in conjunction with a technique due to Baker [2], we obtain a *polynomial time approximation scheme* (PTAS) for the problem on planar graphs. For any fixed $\epsilon > 0$, the algorithm provides a performance guarantee of $1 - \epsilon$.
- **Empirical analysis**. We study the empirical performance of the proposed local-improvement algorithm against baseline methods on both synthetic and real-world networks. Overall, we observe that the empirical approximation ratio of the proposed algorithm is much higher than 1/2, which is our theoretical guarantee.

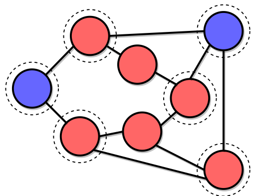


Figure 1: An example assignment of two type-1 agents (blue) and six type-2 agents (red) on a graph $\mathcal{G}$. Vertices with integrated agents are labeled by dashed circles. The index of integration for this assignment (i.e., the number of integrated agents) is 6.

2 RELATED WORK

Integration in public housing. Issues regarding segregation and the need for enhancing integration have been documented extensively in the social science literature (e.g., [12, 22, 25, 26]). In particular, many works on segregation in social networks (e.g., [17, 19]) stem from the pioneering models proposed by Schelling [34], where agents move between vertices to improve their utility values. While Schelling's framework allows the study of agent dynamics, Benabbou et al. [4] study integration in public housing allocation from a planning perspective. In particular, they formulate the setting as a weighted matching problem where the set of available houses is partitioned into blocks, and agents are assigned (by some central agency) to blocks to maximize a utility measure while satisfying some diversity constraints. They establish the **NP**-hardness of the problem and present an approximation algorithm based on a result of Stamoulis [37]. A number of other studies have also addressed integration in the context of public housing from a social science perspective (e.g., [18, 20, 23, 30]).

The problem formulations and the algorithmic techniques used in Benabbou et al. [4] and in our work are significantly different. First, Benabbou et al. [4] examine a weighted matching problem. Their model does not use any network structure for the residences, whereas our work approaches the problem from a graph theoretic standpoint, with the underlying network playing an important role in the formulation. Further, the integration index studied in our work is defined w.r.t the graph structure, whereas the measure used in [4] is based on constraints on the ethnicity quotas for blocks. More importantly, the goal of our work is to find an assignment that maximizes the integration level, whereas the goal in [4] is to maximize the overall utility of agents under a diversity constraint.

Integration indices. Various indices to measure the level of integration in a population are surveyed in [25]. However, most of those indices cannot be naturally extended to a network setting. The integration index IoA considered in our work was proposed by Agarwal et al. [1] in the context of the Schelling Game on networks, where agents can change locations to increase their utilities. Agarwal et al. explore several properties (e.g., the integration price of anarchy/stability) of the index from a game theoretic perspective. Further, they show that finding an assignment for which all agents are integrated (i.e., each agent has at least one neighbor of a different type) is NP-hard [1]. A further discussion of related work is provided in the full version of this manuscript [31].

3 PROBLEM DEFINITION

Graphs and agents. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph, where $\mathcal{V}$ is a set of vertices representing vacant residences, and $\mathcal{E}$ is a set of edges representing the proximity relationship between residences. Let $\mathcal{A}$ be the set of agents to be assigned to $\mathcal{V}$. The set $\mathcal{A}$ is divided into *two* subgroups. Formally, $\mathcal{A}$ is partitioned into two subsets $\mathcal{A}_1$ and $\mathcal{A}_2$; we refer to agents in $\mathcal{A}_i$ as type i agents, $i = 1, 2$. Let $k = |\mathcal{A}_1|$ denote the number of type-1 agents, so $n - k$ is the number of type-2 agents. Without loss of generality, let $k \leq n/2$, and we refer to $\mathcal{A}_1$ as the *minority subgroup*. Lastly, we assume that $|\mathcal{V}| = |\mathcal{A}|$; that is, the number of vertices is the same as the number of agents.

Assignment. An assignment is a mapping from vertices to agents. To simplify the proofs, we use an *equivalent definition* where an assignment is a mapping from vertices to agent types. In particular, an assignment $\mathcal{P} : \mathcal{V} \rightarrow \{1, 2\}$ is a function that assigns an *agent type* to each vertex in $\mathcal{V}$, such that k vertices are assigned type-1 and $n-k$ vertices are assigned type-2. In such an assignment, a type-i vertex is occupied by a type-i agent, $i = 1, 2$. We remark that the above definition of an assignment is mathematically equivalent to defining an assignment to be a mapping from $\mathcal{V}$ to $\mathcal{A}$.

The index of integration. We consider the integration index proposed in [1] and apply it to our context.

*Definition 3.1 (**Index of agent-integration** (IoA) [1]).* Given an assignment $\mathcal{P}$, an agent $x \in \mathcal{A}$ is **integrated** if x has at least one neighbor in $\mathcal{G}$ whose type is different from that of x. Let $\mathcal{A}'$ be the set of integrated agents under $\mathcal{P}$. The index of agent-integration of $\mathcal{P}$ is then defined as the number of integrated agents in $\mathcal{A}$:

$$\texttt{IoA}(\mathcal{P}) = |\mathcal{A}'| \tag{1}$$

Equivalently, a vertex $u \in \mathcal{V}$ is *integrated* under $\mathcal{P}$ if the agent assigned to u is integrated. Thus, we may also view the index as $\texttt{IoA}(\mathcal{P}) = |\mathcal{V}'|$ where $\mathcal{V}'$ is the set of integrated vertices under $\mathcal{P}$. These two definitions of IoA are mathematically equivalent.

The optimization problem. We now define the problem Integration Maximization-Index of Agent Integration (IM-IoA). Note that IM-IoA can be viewed as an optimization version of 2-*weak* coloring [27], where the number of vertices with each color is specified, and the number of properly colored vertices is maximized.

*Definition 3.2 (**IM-IoA**).* Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a set $\mathcal{A}$ of agents with k type-1 and $n-k$ type-2 agents, find an assignment $\mathcal{P}$ such that $\texttt{IoA}(\mathcal{P})$ is maximized.

4 APPROXIMATION FOR GENERAL GRAPHS

IM-IoA is known to be **NP**-hard [1]. In this section, we present a *local-improvement algorithm* for IM-IoA and show that the algorithm achieves a factor 1/2 approximation for general graphs. For convenience in presenting the proofs, we consider an *assignment* from the perspective of vertices rather than that of the agents. As stated earlier, these two definitions are equivalent.

The algorithm. Starting from a random assignment $\mathcal{P}$, in each iteration, we find a pair of vertices with different types such that swapping their types strictly increases the objective. Specifically, let u be a type-1 vertex, and v be a type-2 vertex. We swap the types of u and v if and only if the resulting new assignment $\mathcal{P}'$ yields a strictly higher IoA; that is, $\texttt{IoA}(\mathcal{P}) < \texttt{IoA}(\mathcal{P}')$. The algorithm terminates when no such swap can be made. The pseudocode is given in Algorithm (1).

4.1 Analysis of the algorithm

Given a problem instance of IM-IoA, let $\mathcal{P}$ be a saturated assignment[1] returned by Algorithm (1). Let $\mathcal{P}^*$ be an optimal assignment that achieves the maximum objective, denoted by OPT. We assume that $\mathcal{P} \neq \mathcal{P}^*$. In this section, we show that $\texttt{IoA}(\mathcal{P}) \geq 1/2 \cdot \texttt{IoA}(\mathcal{P}^*) = 1/2 \cdot \text{OPT}$, thereby establishing a 1/2 approximation. Due to the page limit, we sketch the proof here; the detailed proof appears in the full version [31].

Algorithm 1: Local-Improvement-IoA

Input : A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, k, where $k \leq |\mathcal{V}|/2$
Output: An assignment $\mathcal{P}$

```
P ← a random assignment, Updated ← True
while Updated do
    Updated ← False
    for x ∈ V_1(P) do
        for y ∈ V_2(P) do
            P′ ← the assignment where P′(x) = P(y) and
             P′(y) = P(x)
            if IoA(P′) > IoA(P) then
                P = P′, Updated ← True & break
return P
```

Under the assignment $\mathcal{P}$, we call a vertex v a *type*-1 (or *type*-2) vertex if $\mathcal{P}(v) = 1$ (or $\mathcal{P}(v) = 2$). Let $\mathcal{V}_1(\mathcal{P})$ and $\mathcal{V}_2(\mathcal{P})$ denote the set of type-1 and type-2 vertices under $\mathcal{P}$. Let $\mathcal{V}_1^{\mathsf{U}}(\mathcal{P})$ and $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$ denote the set of uncovered[2] type-1 and type-2 vertices under $\mathcal{P}$. For each vertex u, let $\mathcal{N}_u^{\mathsf{U}}(\mathcal{P})$ denote the set of neighbors of u that are uncovered under $\mathcal{P}$, and let $\Gamma_u(\mathcal{P})$ denote the set of different-type neighbors of u that are **uniquely** covered by u, i.e., $\Gamma_u(\mathcal{P})$ is the set of vertices v such that (i) v is a neighbor of u, (ii) the type of v is different from the type of u, and (iii) v has no other neighbor whose type is the same as u's type.

▷ Observation 4.1. *The index* $\texttt{IoA}(\mathcal{P}) = n - |\mathcal{V}_1^{\mathsf{U}}(\mathcal{P})| - |\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|$.

We now consider the following mutually exclusive and collectively exhaustive cases of $\mathcal{V}_1^{\mathsf{U}}(\mathcal{P})$ and $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$ under the saturated assignment $\mathcal{P}$. We start with a simple case where all the type-2 vertices under $\mathcal{P}$ are integrated.

Case 1: $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) = \emptyset$.
Under this case, all vertices in $\mathcal{V}_2(\mathcal{P})$ are integrated which gives $\texttt{IoA}(\mathcal{P}) \geq (1/2) \cdot \text{OPT}$. This case trivially implies that the algorithm provides a 1/2 approximation. We now look at the remaining case:
Case 2: $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) \neq \emptyset$.
Under this case, there exists at least one vertex in $\mathcal{V}_2(\mathcal{P})$ that is not integrated. We first show that $\mathcal{V}_1^{\mathsf{U}}(\mathcal{P})$ and $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$ cannot both be non-empty.

▷ Lemma 4.2. *For a saturated assignment* $\mathcal{P}$, *if* $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) \neq \emptyset$, *then* $\mathcal{V}_1^{\mathsf{U}}(\mathcal{P}) = \emptyset$.

Proof. (Sketch) Let $y \in \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$ be a vertex of type-2 that is not integrated (i.e., all neighbors of y are of type-2). For contradiction, suppose $\mathcal{V}_1^{\mathsf{U}}(\mathcal{P}) \neq \emptyset$. Now let $x \in \mathcal{V}_1^{\mathsf{U}}(\mathcal{P})$ be an non-integrated vertex of type-1 whose neighbors are all of type-1. Let $\mathcal{P}'$ denote the assignment where we switch the types between x and y, that is, $\mathcal{P}'(x) = \mathcal{P}(y) = 2$, $\mathcal{P}'(y) = \mathcal{P}(x) = 1$, while the types of all other vertices remain unchanged. One can verify that $\texttt{IoA}(\mathcal{P}') \geq$

[1] An assignment is *saturated* if no pairwise swap of types between a type-1 and a type-2 vertices can increase the objective.

[2] A vertex is "covered" if it is integrated and "uncovered" otherwise.

$\texttt{IoA}(\mathcal{P})+2$, that is, switching the types of x and y increases the index $\texttt{IoA}$ by at least 2. This implies the existence of an improvement move from $\mathcal{P}$, which contradicts the fact that $\mathcal{P}$ is saturated. It follows that $\mathcal{V}_1^{\text{U}}(\mathcal{P}) = \varnothing$. ∎

Lemma 4.2 implies that under case 2 (i.e., $\mathcal{V}_2^{\text{U}}(\mathcal{P}) \neq \varnothing$), we have $\mathcal{V}_1^{\text{U}}(\mathcal{P}) = \varnothing$. We now consider the following two mutually exclusive and collectively exhaustive subcases under Case 2 and show that the approximation factor under each subcase is 1/2.

<u>**Subcase 2.1:**</u> $\mathcal{V}_2^{\text{U}}(\mathcal{P}) \neq \varnothing$, and $\Gamma_x(\mathcal{P}) \neq \varnothing, \quad \forall x \in \mathcal{V}_1(\mathcal{P})$, that is, for each type-1 vertex $x \in \mathcal{V}_1(\mathcal{P})$, there is at least one type-2 neighbor of x that is *uniquely* covered ("made integrated") by x.

Suppose $\mathcal{P} \neq \mathcal{P}^*$, that is, for some vertices $x \in \mathcal{V}$, $\mathcal{P}(x) \neq \mathcal{P}^*(x)$. Let $\tilde{\mathcal{V}}_{2-1} = \{v \in \mathcal{V} \ : \ \mathcal{P}(v) = 2, \mathcal{P}^*(v) = 1\}$ be the set of vertices that are type-2 under $\mathcal{P}$, but are type-1 under $\mathcal{P}^*$. Analogously, let $\tilde{\mathcal{V}}_{1-2} = \{v \in \mathcal{V} \ : \ \mathcal{P}(v) = 1, \mathcal{P}^*(v) = 2\}$ be the set of vertices of type-1 under $\mathcal{P}$, but are of type-2 under $\mathcal{P}^*$. Observe that $|\tilde{\mathcal{V}}_{2-1}| = |\tilde{\mathcal{V}}_{1-2}|$. We may view $\mathcal{P}^*$ as the result of a transformation from $\mathcal{P}$ under pairwise swaps of types between $\tilde{\mathcal{V}}_{2-1}$ and $\tilde{\mathcal{V}}_{1-2}$. An example is given in Figure (2). We present a key lemma that bounds the difference between the objective values of $\mathcal{P}$ and $\mathcal{P}^*$.

▷ Lemma 4.3 (Subcase 2.1). *Let $\mathcal{P}$ be a saturated assignment under subcase 2.1, and let $\mathcal{P}^*$ be an optimal assignment. We have*

$$\texttt{IoA}(\mathcal{P}^*) - \texttt{IoA}(\mathcal{P}) \leq \sum_{y \in \tilde{\mathcal{V}}_{2-1} \setminus \mathcal{V}_2^{\text{U}}(\mathcal{P})} |(\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| + \sum_{y \in \tilde{\mathcal{V}}_{2-1} \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})} \Big(|(\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| + 1\Big). \tag{2}$$

Proof. (Sketch) Since $\mathcal{P}$ is saturated, Lemma (4.2) implies that all type-1 vertices under $\mathcal{P}$ are integrated. Thus, the difference $\texttt{IoA}(\mathcal{P}^*) - \texttt{IoA}(\mathcal{P})$ is at most the number of type-2 vertices that are integrated under $\mathcal{P}^*$ but are *not* integrated under $\mathcal{P}$.

Let $f : \tilde{\mathcal{V}}_{1-2} \to \tilde{\mathcal{V}}_{2-1}$ be an arbitrary bijective mapping. We may regard $\mathcal{P}^*$ as a result of the transformation from $\mathcal{P}$ via pairwise swaps of types between vertices specified by f (i.e., the type of $x \in \tilde{\mathcal{V}}_{1-2}$ is swapped with the type of $f(x) \in \tilde{\mathcal{V}}_{2-1}$). Observe that only vertices in $\mathcal{V}_2^{\text{U}}(\mathcal{P})$ that are adjacent to $\tilde{\mathcal{V}}_{2-1}$ (or within $\tilde{\mathcal{V}}_{2-1}$) under $\mathcal{P}$ can be newly integrated under $\mathcal{P}^*$ after swapping $\tilde{\mathcal{V}}_{1-2}$ with $\tilde{\mathcal{V}}_{2-1}$. (By the definition of $\mathcal{V}_2^{\text{U}}(\mathcal{P})$, vertices in $\tilde{\mathcal{V}}_{1-2}$ have no neighbors in $\mathcal{V}_2^{\text{U}}(\mathcal{P})$.) It follows that for each vertex $y \in \tilde{\mathcal{V}}_{2-1}$, at most $|(\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})|$ of its neighbors can become newly integrated after transforming from $\mathcal{P}$ to $\mathcal{P}^*$. Further, if also $y \in \tilde{\mathcal{V}}_{2-1} \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})$, y itself could also be newly integrated after the swap. We then have

$$\begin{aligned}\texttt{IoA}(\mathcal{P}^*) - \texttt{IoA}(\mathcal{P}) &\leq \Big|\bigcup_{y \in \tilde{\mathcal{V}}_{2-1}} \mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})\Big| + |\tilde{\mathcal{V}}_{2-1} \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| \\ &\leq \sum_{y \in \tilde{\mathcal{V}}_{2-1} \setminus \mathcal{V}_2^{\text{U}}(\mathcal{P})} |(\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| \qquad (3) \\ &\quad + \sum_{y \in \tilde{\mathcal{V}}_{2-1} \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})} \Big(|(\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| + 1\Big)\end{aligned}$$

where the last inequality follows from the union bound. ∎

We now proceed to show that the difference between $\texttt{IoA}(\mathcal{P})$ and $\texttt{IoA}(\mathcal{P}^*)$ established in Lemma (4.3) is *at most* $\texttt{IoA}(\mathcal{P})$, thereby establishing $\texttt{IoA}(\mathcal{P}) \geq \frac{1}{2} \cdot \texttt{IoA}(\mathcal{P}^*)$. Recall that for each vertex $x \in \mathcal{V}$, $\Gamma_x(\mathcal{P})$ is the set of neighbors of x whose types are different from x, and are uniquely covered by x under $\mathcal{P}$. By the definition of Subcase 2.1, $\Gamma_x(\mathcal{P})$ is not empty for all $x \in \mathcal{V}_1(\mathcal{P})$. We first argue that for any $y \in \mathcal{V}_2^{\text{U}}(\mathcal{P})$ and any $x \in \mathcal{V}_1(\mathcal{P})$, we have $|\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| \leq |\Gamma_x(\mathcal{P})|$.

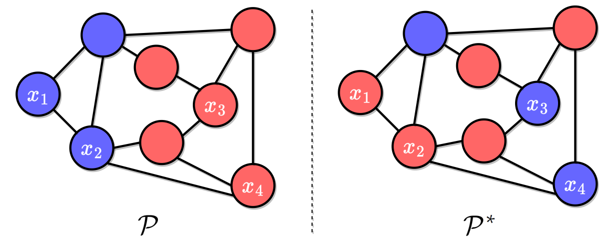


Figure 2: Two assignments $\mathcal{P}$ and $\mathcal{P}^*$ where type-1 and type-2 vertices are highlighted in blue and red, respectively. In this case, $\tilde{\mathcal{V}}_{2-1} = \{x_3, x_4\}$ and $\tilde{\mathcal{V}}_{1-2} = \{x_1, x_2\}$. We may then transform $\mathcal{P}$ into $\mathcal{P}^*$ by swapping types between the pair (x_1, x_3) and between (x_2, x_4). Note that this example is *only to demonstrate how $\tilde{\mathcal{V}}_{2-1}$ and $\tilde{\mathcal{V}}_{1-2}$ are defined,* as $\mathcal{P}$ cannot be a saturated assignment returned by the algorithm.

▷ Lemma 4.4 (Subcase 2.1). *Given a saturated assignment $\mathcal{P}$, for any $y \in \mathcal{V}_2^{\text{U}}(\mathcal{P})$ and any $x \in \mathcal{V}_1(\mathcal{P})$, we have*

$$|\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| \leq |\Gamma_x(\mathcal{P})|.$$

Proof. (Sketch) Given that y is not integrated under $\mathcal{P}$, x and y cannot be adjacent. Since $\mathcal{P}$ is a saturated assignment, if the types of x and y are to be swapped, the number of newly integrated vertices would be at most the number of newly non-integrated vertices. Further, one can verify that the number of vertices that are newly integrated is at least $|\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| + 1$, and the number of vertices that are newly non-integrated is at most $|\Gamma_x(\mathcal{P})| + 1$. Since $\mathcal{P}$ is saturated, it follows that $|\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| \leq |\Gamma_x(\mathcal{P})|$. This concludes the proof. ∎

We now establish the next lemma, which bounds the size of $\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})$ for $y \in \mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\text{U}}(\mathcal{P})$ and $x \in \mathcal{V}_1(\mathcal{P})$.

▷ Lemma 4.5 (Subcase 2.1). *Given a saturated assignment $\mathcal{P}$, for any $y \in \mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\text{U}}(\mathcal{P})$ and any $x \in \mathcal{V}_1(\mathcal{P})$, we have*

$$|\mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P})| \leq |\Gamma_x(\mathcal{P})| + 1$$

Proof. (Sketch) We partition $\mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\text{U}}(\mathcal{P})$ into two subsets $\mathcal{B}$ and $\mathcal{C}$, as follows. Subset $\mathcal{B}$ is the set of integrated type-2 vertices whose neighbors are all integrated under $\mathcal{P}$, i.e., $\mathcal{B} = \{y \in \mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\text{U}}(\mathcal{P}) \ : \ \mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P}) = \varnothing\}$. Subset $\mathcal{C}$, the complement of $\mathcal{B}$, is the set of integrated type-2 vertices with at least one non-integrated neighbor under $\mathcal{P}$, i.e., $\mathcal{C} = \{y \in \mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\text{U}}(\mathcal{P}) \ : \ \mathcal{N}(y) \cap \mathcal{V}_2^{\text{U}}(\mathcal{P}) \neq \varnothing\}$. The lemma clearly holds if $y \in \mathcal{B}$. Further, we show that for the case when $y \in \mathcal{C}$, no type-1 neighbors of y is uniquely covered by y under $\mathcal{P}$ (i.e., $\Gamma_y(\mathcal{P}) = \varnothing$). Further, suppose $y \in \mathcal{C}$, consider an objective non-increasing move from $\mathcal{P}$ where we swap the types between x and y. If y is a neighbor of x under

$\mathcal{P}$, one can verify that the the maximum loss is $|\Gamma_x(\mathcal{P})|$ and the minimum gain is $|\mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|$. Thus

$$|\mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| \leq |\Gamma_x(\mathcal{P})|. \tag{4}$$

On the other hand, if y is **not** a neighbor of x under $\mathcal{P}$, one can verify that the maximum loss is $|\Gamma_x(\mathcal{P})| + 1$ and the minimum gain is $|\mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|$. Thus

$$|\mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| \leq |\Gamma_x(\mathcal{P})| + 1. \tag{5}$$

This concludes the proof. ∎

We are now ready to establish $\texttt{IoA}(\mathcal{P}) \geq \frac{1}{2} \cdot \texttt{IoA}(\mathcal{P}^*)$ under Subcase 2.1.

▷ Lemma 4.6 (Subcase 2.1). *Suppose $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) \neq \emptyset$ and $\Gamma_x(\mathcal{P}) \neq \emptyset, \forall x \in \mathcal{V}_1(\mathcal{P})$, we have $\texttt{IoA}(\mathcal{P}) \geq \frac{1}{2} \cdot \texttt{IoA}(\mathcal{P}^*)$ where $\mathcal{P}^*$ is an optimal assignment that gives the maximum objective.*

Proof. (Sketch) Note that $\tilde{\mathcal{V}}_{2-1}$ is a subset of $\mathcal{V}_2(\mathcal{P})$. Further, observe that $\Gamma_x(\mathcal{P})$ are pairwise disjoint for different vertices $x \in \mathcal{V}_1(\mathcal{P})$. Now, by Lemma (4.3) and (4.5), We have

$$\begin{aligned} \texttt{IoA}(\mathcal{P}^*) - \texttt{IoA}(\mathcal{P}) &\leq \left(\sum_{y \in \tilde{\mathcal{V}}_{2-1}} |\Gamma_{f^{-1}(y)}(\mathcal{P})| \right) + |\tilde{\mathcal{V}}_{2-1}| \\ &\leq |\mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| + |\mathcal{V}_1(\mathcal{P})| \quad (6) \\ &\leq \texttt{IoA}(\mathcal{P}) \end{aligned}$$

where Inequality (6) follows from $|\tilde{\mathcal{V}}_{2-1}| = |\tilde{\mathcal{V}}_{1-2}| \leq |\mathcal{V}_1(\mathcal{P})|$ and $\left(\sum_{y \in \tilde{\mathcal{V}}_{2-1}} |\Gamma_{f^{-1}(y)}(\mathcal{P})|\right) \leq |\mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|$. ∎

We have shown that if $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) \neq \emptyset$ and $\Gamma_x(\mathcal{P}) \neq \emptyset, \forall x \in \mathcal{V}_1(\mathcal{P})$, the algorithm gives a 1/2 approximation. We proceed to the final subcase.

__**Subcase 2.2:**__ $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) \neq \emptyset$, and $\Gamma_x(\mathcal{P}) = \emptyset, \exists x \in \mathcal{V}_1(\mathcal{P})$, that is, there exists at least one type-1 vertex $x \in \mathcal{V}_1(\mathcal{P})$ such that for each type-2 neighbor y of x, y is adjacent to at least one type-1 vertex *other than* x.

▷ Lemma 4.7 (Subcase 2.2). *Under subcase 2.2, for each non-integrated type-2 vertex $y \in \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$, all type-2 neighbors of y are integrated (i.e., $\mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) = \emptyset$) under $\mathcal{P}$. That is, the vertices in $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$ form an independent set of $\mathcal{G}$.*

Proof. (Sketch) Given such a $x \in \mathcal{V}_1(\mathcal{P})$ defined in Subcase 2.2, for contradiction, suppose there exists a non-integrated type-2 vertex $y \in \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$ such that at least one type-2 neighbor, denoted by $y' \in \mathcal{N}(y)$, of y is not integrated under $\mathcal{P}$. (Note that all neighbors of y are of type-2 since y is not integrated.) Now consider a new assignment $\mathcal{P}'$ where we switch the types between x and y. One can verify that $\texttt{IoA}(\mathcal{P}') \geq \texttt{IoA}(\mathcal{P}) + 1$, that is, after the switch, the index $\texttt{IoA}$ would increase by at least 1. This implies the existence of an improvement move from $\mathcal{P}$, which contradicts $\mathcal{P}$ being a saturated assignment. Thus, no such a non-integrated type-2 vertex y' of y can exist. ∎

Observe that $\texttt{IoA}(\mathcal{P}) = (n - |\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|)$. With Lemma (4.7) in place, we now argue that the size of $\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})$ cannot be too large.

▷ Lemma 4.8 (Subcase 2.2). *Under Subcase 2.2, $|\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| \leq \frac{n}{2}$*

Proof. (Sketch) Let $\mathcal{Y} := \{y \in \mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) \ : \ \mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P}) \neq \emptyset\}$ be the set of type-2 *integrated* vertices which have at least one non-integrated type-2 neighbor. We first note that $\Gamma_y(\mathcal{P})$ (if not empty) are mutually disjoint for different $y \in \mathcal{Y}$. It follows that $\texttt{IoA}(\mathcal{P}) \geq |\mathcal{Y}| + \sum_{y \in \mathcal{Y}} |\Gamma_y(\mathcal{P})|$. Suppose we switch the types between such a vertex x and a vertex $y \in \mathcal{Y}$, and let $\mathcal{P}'$ denote the resulting new assignment. One can verify that the maximum loss of objective after the swap is $|\Gamma_y(\mathcal{P})| + 1$, whereas the minimum gain is $|\mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|$. Since $\mathcal{P}$ is a saturated assignment returned by the algorithm, we must have $\texttt{IoA}(\mathcal{P}) \geq \texttt{IoA}(\mathcal{P}')$. Therefore, $|\mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| \leq |\Gamma_y(\mathcal{P})| + 1, \ \forall y \in \mathcal{Y}$. Overall, we have that

$$\begin{aligned} |\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| &= |\bigcup_{y \in \mathcal{Y}} \mathcal{N}(y) \cap \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| && (7) \\ &\leq |\mathcal{V}_1(\mathcal{P})| + |\mathcal{Y}| && (8) \\ &\leq |\mathcal{V}_1(\mathcal{P})| + |\mathcal{V}_2(\mathcal{P}) \setminus \mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| && (9) \\ &= n - |\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|. && (10) \end{aligned}$$

It immediately follows that $|\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| \leq \frac{n}{2}$. ∎

Lastly, Since $\texttt{IoA}(\mathcal{P}) = n - |\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})|$, by Lemma (4.8), we have $\texttt{IoA}(\mathcal{P}) = n - |\mathcal{V}_2^{\mathsf{U}}(\mathcal{P})| \geq \frac{1}{2} \cdot n \geq \frac{1}{2} \cdot \texttt{IoA}(\mathcal{P}^*)$, thereby establishing a 1/2 approximation for Subcase 2.2. Overall, we have shown that a saturated assignment $\mathcal{P}$ returned by Algorithm (1) gives a 1/2-approximation for IM-IoA. Thus:

▷ Theorem 4.9. *Algorithm (1) gives a $\frac{1}{2}$-approximation for IM-*IoA.

Tightness of Analysis. We present a class of problem instances where the approximation ratio of the solution produced by Algorithm (1) can be made arbitrarily close to 1/2. Therefore, the ratio 1/2 in the statement of Theorem (4.9) cannot be improved, so *our analysis is tight.* Due to the page limit, the proof appears in the full version [31].

▷ Proposition 4.10. *For every $\epsilon > 0$, there exists a problem instance of IM-*IoA *for which there is a saturated assignment $\mathcal{P}$ such that* $\texttt{IoA}(\mathcal{P}) \leq (\frac{1}{2} + \epsilon) \cdot$ *OPT.*

5 SUBGROUPS WITH SIMILAR SIZES

In this section, we study the problem instances where the number of type-1 agents is a constant fraction of the total number of agents, that is, $k = \alpha \cdot n$ for some constant $0 \leq \alpha \leq 1/2$. We refer to this problem as αn-IM-IoA. For example, $\alpha = 1/2$ represents the *bisection* constraint. We first show that αn-IM-IoA remains computationally intractable. See the full version [31] for the proof.

▷ Theorem 5.1. *The problem αn-IM-*IoA *is **NP**-hard.*

5.1 A semidefinite programming approach

We now present an approximation algorithm for αn-IM-IoA based on semidefinite programming (SDP) relaxation [15]. The overall scheme is inspired by the work of Frieze and Jerrum [13] on the Max-Bisection problem. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, each vertex $i \in \mathcal{V}$ has a binary variable $x_i \in \{-1, 1\}$ such that $x_i = -1$ if i is of type-1, and $x_i = 1$ if i is of type-2. First, a valid quadratic program (QP) for the problem is as follows (see the full version [31] for the proof): maximize $\sum_{i \in \mathcal{V}} \max_{j \in \mathcal{N}(i)} \{\frac{1-x_i x_j}{2}\}$ s.t. $\sum_{i<j} x_i x_j = \frac{(1-2\alpha)^2 \cdot n^2 - n}{2}$. It can be verified that the following SDP is a relaxation of the QP:

$$\text{SDP}: \quad \text{maximize} \quad \sum_{i \in \mathcal{V}} \max_{j \in \mathcal{N}(i)} \{\frac{1 - \vec{y}_i \cdot \vec{y}_j}{2}\}$$
$$\text{s.t.} \quad \sum_{i<j} \vec{y}_i \cdot \vec{y}_j \le \frac{(1-2\alpha)^2 \cdot n^2 - n}{2}$$
$$\vec{y}_i \cdot \vec{y}_i = 1, \quad \forall i \in V$$

Main idea of the algorithm and analysis. Our algorithm involves two steps. We elaborate on these steps and the analysis below.

(1) The SDP solution $\vec{y}_i, i = 1, \ldots, n$ is not a feasible integral solution. So we "round" it to get a partition $(\mathcal{V}_1, \mathcal{V}_2)$ using the *hyperplane rounding* [15] approach. We show that the expected number of integrated vertices is $\Omega(OPT_{SDP})$, where OPT_{SDP} is the value of the SDP solution.

(2) $\{\mathcal{V}_1, \mathcal{V}_2\}$ need not be a $(\alpha n, (1-\alpha)n)$-partition, so we fix it by moving $|\mathcal{V}_1| - \alpha n$ vertices from $\mathcal{V}_1$ to the other side. In particular, we present a strategy that picks a vertex to remove from $\mathcal{V}_1$ at each step, which minimizes the decrease in `IoA`. Overall, to achieve the overall guarantees, we increase the probability of success by running the rounding and size adjustment step multiple times, and taking the best solution.

First step: Round the SDP. Let $\{\vec{y}_1, ..., \vec{y}_n\}$ be an optimal solution to the SDP; let OPT_{SDP} be the objective value of the SDP. We round the SDP solution to a partition $\{\mathcal{V}_1, \mathcal{V}_2\}$ of the vertex set such that vertices in $\mathcal{V}_i$ are of type-i, $i = 1, 2$ by applying Goemans and Williamson's *hyperplane rounding method* [15]. In particular, we draw a random hyperplane thought the origin with a normal vector r, and then $\mathcal{V}_1 = \{i : \vec{y}_i \cdot r \ge 0\}$ and $\mathcal{V}_2 = \{i : \vec{y}_i \cdot r < 0\}$.

Consider an assignment $\mathcal{P}$ generated by the above rounding method (i.e., vertices in $\mathcal{V}_i$ are assigned to type-i). Let $f(\mathcal{V}_1) : 2^{\mathcal{V}} \to \mathbb{N}$ be the number of integrated vertices under $\mathcal{P}$. We establish the following lemma. The detailed proof appears in the full version [31].

▷ Lemma 5.2. *$\mathbb{E}[f(\mathcal{V}_1)] \ge \alpha_{GW} \cdot OPT_{SDP}$, where $\alpha_{GW} \ge 0.878567$.*

Proof. (Sketch) We first establish that $\Pr[i \text{ is integrated}] \ge \max_{j \in \mathcal{N}(i)} \{\frac{\arccos(\vec{y}_i \cdot \vec{y}_j)}{\pi}\}$ for any vertex i. Further, as shown in [15], $\arccos(z)/\pi \ge \alpha_{GW} \cdot (1-z)/2$ for any real $z \in [-1, 1]$. Thus,

$$\mathbb{E}[f(\mathcal{V}_1)] \ge \sum_{i \in \mathcal{V}} \max_{j \in \mathcal{N}(i)} \{\frac{\arccos(\vec{y}_i \cdot \vec{y}_j)}{\pi}\} \tag{11}$$
$$\ge \alpha_{GW} \cdot \sum_{i \in \mathcal{V}} \max_{j \in \mathcal{N}(i)} \{\frac{1 - \vec{y}_i \cdot \vec{y}_j}{2}\} \tag{12}$$
$$\ge \alpha_{GW} \cdot \text{OPT}_{SDP} \tag{13}$$

This concludes the proof. ∎

Second step: Fix the size. In the previous step, we have shown that given a partition $\{\mathcal{V}_1, \mathcal{V}_2\}$ resulting from hyperplane rounding, if all vertices in $\mathcal{V}_1$ are of type-1, and all vertices in $\mathcal{V}_2$ are of type-2, then the expected number of integrated vertices is at least α_{GW} of the optimal. However, the partition is not necessarily an $(\alpha n, (1-\alpha)n)$-partition. Thus, we present an algorithm to move vertices from one subset to another such that (i) the resulting new partition is an $(\alpha n, (1-\alpha)n)$-partition, and (ii) the objective does not decrease "too much" after the moving process.

Algorithm 2: `Fix-the-Size`. Without loss of generality, suppose $|\mathcal{V}_1| \ge \alpha n$. Overall, our algorithm consists of $T = |\mathcal{V}_1| - \alpha n$ iterations, and in each each iteration, we move a vertex $i \in \mathcal{V}_1$ to $\mathcal{V}_2$. Specifically, let $\mathcal{V}_1^{(t)}$ be the subset at the tth iteration, with $\mathcal{V}_1^{(0)} = \mathcal{V}_1$. To obtain $\mathcal{V}_1^{(t+1)}$, we choose $i \in \mathcal{V}_1^{(t)}$ to be a vertex that maximizes $f(\mathcal{V}_1^{(t)} \setminus \{i\}) - f(\mathcal{V}_1^{(t)})$, and the move i to the other subset. Lemma 5.3 below establishes the performance of Algorithm (2); a detailed proof appears in the full version [31].

▷ Lemma 5.3. *We have $\frac{f(\mathcal{V}_1^{(T)})}{|\mathcal{V}_1^{(T)}|} \ge \frac{f(\mathcal{V}_1)}{|\mathcal{V}_1|}$ where $\mathcal{V}_1^{(T)}$, with $T = |\mathcal{V}_1| - \alpha n$, is returned by Algorithm (2).*

The final algorithm. We have defined the two steps (i.e., (i) *round the SDP* and (ii) *fix the sizes of the two subsets*) needed to obtain a feasible solution for the problem. Let $\epsilon \ge 0$ be a small constant, and let $L = \lceil \log_a(\frac{1}{\epsilon}) \rceil$ where $a = [(1+\beta) - (1-\epsilon)2\alpha_{GW}]/(1 + \beta - 2\alpha_{GW})$, $\beta = 1/(4(\alpha - \alpha^2))$. Note that L is a constant w.r.t. n. The final algorithm consists of L iterations, where each iteration performs the two steps defined above. This gives us L feasible solutions. The algorithm then outputs a solution with the highest objective among the L feasible solutions.

▷ Theorem 5.4. *The final algorithm gives a factor*

$$\frac{\alpha\left((1-\epsilon) \cdot 2\alpha_{GW} - \frac{\gamma - \gamma^2}{\alpha - \alpha^2}\right)}{\gamma} \cdot (1-\epsilon)$$

approximation with high probability, where $\alpha_{GW} \ge 0.878567$, $\epsilon \ge 0$ is an arbitrarily small positive constant, $\alpha = k/n$ is the fraction of minority agents in the group, and $\gamma = \sqrt{\alpha(1-\alpha)(1-\epsilon) \cdot 2\alpha_{GW}}$.

See the full version [31] for detailed proof. For small enough ϵ, say $\epsilon = 10^{-3}$, the approximation ratio is greater than $1/2$ for α in range $[0.403, 0.5]$. For example, $\alpha = 0.45$ gives a ratio of 0.5781, and $\alpha = 0.5$ gives a ratio of 0.6492.

6 TREE-WIDTH BOUNDED GRAPHS AND PLANAR GRAPHS

In this section, we show that IM-`IoA` can be solved in polynomial time on treewidth bounded graphs. Using this result, we obtain a *polynomial time approximation scheme* (PTAS) for the problem on planar graphs.

6.1 A dynamic programming algorithm for treewidth bounded graphs

The concept *treewidth* was introduced in the seminal work of Robertson and Seymour [32]. Many **NP**-hard graph problems are known to be solvable in polynomial time when the underlying graphs have bounded treewidth. In this section, we present a polynomial-time dynamic programming algorithm for IM-`IoA` for the class of treewidth bounded graphs. We refer readers to the full version [31] for the definition of a tree decomposition and treewidth.

Dynamic programming setup. Given an instance of IM-`IoA` with graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and the number k of minority agents, let

$\mathcal{T} = (\mathcal{I}, \mathcal{F})$ be a tree decomposition of $\mathcal{G}$ with treewidth σ. For each $\mathcal{X}_i \in \mathcal{I}$, let $\mathcal{Y}_i$ be the set of vertices in the bags in the subtree rooted at $\mathcal{X}_i$. Let $\mathcal{G}[\mathcal{Y}_i]$ denote the subgraph of $\mathcal{G}$ induced on $\mathcal{Y}_i$. For each bag $\mathcal{X}_i$, we define an array H_i to keep track of the optimal objectives in $\mathcal{G}[\mathcal{Y}_i]$. In particular, let $H_i(S, S', \gamma)$ be the optimal objective value for $\mathcal{G}[\mathcal{Y}_i]$ such that (i) vertices in the subset $S \subseteq \mathcal{X}_i$ are of type-1 and vertices in $\mathcal{X}_i \setminus S$ are of type-2; (ii) vertices in $S' \subseteq \mathcal{X}_i$ are to be *treated as integrated*; and (iii) $\mathcal{G}[\mathcal{Y}_i]$ has a total of γ type-1 vertices and $|\mathcal{Y}_i| - \gamma$ type-2 vertices. For space reasons, the update scheme for H_i for each bag $\mathcal{X}_i$ and the proof of correctness appear in the full version [31].

▷ Theorem 6.1. *IM-*IoA *can be solved in polynomial time on treewidth bounded graphs.*

6.2 PTAS for planar graphs

Using the proof presented in [1], it is easy to verify that IM-IoA remains hard on planar graphs. Given a planar graph $\mathcal{G}$ and for any fixed $\epsilon > 0$, based on the technique introduced in [2], we present a *polynomial time approximation scheme* that achieves a $(1 - \epsilon)$ approximation for IM-IoA.

PTAS Outline. Let $q = 2 \cdot \lceil 1/\epsilon \rceil$. We start with a plane embedding of $\mathcal{G}$, which partitions the set of vertices into ℓ layers for some integer $\ell \le n$. Let $\mathcal{V}_i$ be the set of vertices in the ith layer, $i = 1, ..., \ell$. For each $r = 1, ..., q$, observe that we may partition the vertex set into $t + 1$ subsets, where $t = \lceil (\ell - r)/q \rceil$, such that (i) the first subset $\mathcal{W}_{(1,r)}$ consists of the first r layers, (ii) the last subset $\mathcal{W}_{(t+1,r)}$ consists of the last $((l - r) \mod q)$ layers, and (iii) each ith subset $\mathcal{W}_{(i,r)}$ in the middle contains q layers in sequential order. Let $\mathcal{W}_r = \{\mathcal{W}_{(1,r)}, ..., \mathcal{W}_{(t+1,r)}\}$ be such a partition. Let $\mathcal{G}_{(i,r)}$ be the subgraph induced on $\mathcal{W}_{(i,r)}$, $i = 1., , , t + 1$. It is known that each $\mathcal{G}_{(i,r)}$ is a q-outerplanar graph with treewidth $O(q)$ [5], which is bounded. Let $\mathcal{G}_r = \bigcup_i \mathcal{G}_{(i,r)}$. By Theorem (6.1), we can solve the problem optimally on each $\mathcal{G}_r$, $r = 1, ..., q$, in polynomial time. The algorithm then returns the solution with the largest objective over all $r = 1, ..., q$. Using the fact that q is fixed, one can verify that the running time of the overall scheme is polynomial in n.

▷ Theorem 6.2. *The PTAS algorithm gives a factor $(1 - \epsilon)$ approximation on planar graphs for any fixed $\epsilon > 0$.*

Proof. (Sketch) Let $q = 2 \cdot \lceil 1/\epsilon \rceil$. We show that the algorithm gives a $1 - 2/q \ge 1 - \epsilon$ approximation. Let $\mathcal{P}^*$ be an assignment of agents on $\mathcal{G}$ that gives the maximum number of integrated agents. Fix an integer $r \in [1 .. q]$, and let $\mathcal{W}_r = \{\mathcal{W}_{(1,r)}, ..., \mathcal{W}_{(t+1,r)}\}$ be a partition of the vertex set as described above. Let $\mathcal{P}_r$ be an assignment on $\mathcal{G}_r$ that is obtained from the proposed algorithm. We now look at the assignments $\mathcal{P}_r$ and $\mathcal{P}^*$, restricted to vertices in $\mathcal{W}_r$. Specifically, let $\mathcal{P}_{(i,r)}$ and $\mathcal{P}^*_{(i,r)}$ be the assignment of agents restricted to the subset $W_{(i,r)}$ under $\mathcal{P}_r$ and $\mathcal{P}^*$, respectively. Further, let $\texttt{IoA}(\mathcal{P}_{(i,r)})$ be the number of integrated agents in $\mathcal{G}_{(i,r)}$ under $\mathcal{P}_r$, and $\texttt{IoA}(\mathcal{P}^*_{(i,r)})$ be the number of integrated agents in $\mathcal{G}_{(i,r)}$ under $\mathcal{P}^*$.

Define $\Delta_r = \texttt{IoA}(\mathcal{P}^*) - \sum_{i=1}^{t+1} \texttt{IoA}(\mathcal{P}^*_{(i,r)})$. Integrated vertices that are left uncounted can only exist on the two adjacent layers between each pair of subgraphs $\mathcal{G}_{(i,r)}$ and $\mathcal{G}_{(i+1,r)}$, $i = 1, ...t$. Let $\mathcal{V}^*$ be the set of integrated vertices under $\mathcal{P}^*$. We then have, $\Delta_r \le \sum_{j=0}^{t} (\mathcal{V}^* \cap \mathcal{V}_{j \cdot q+r}) + (\mathcal{V}^* \cap \mathcal{V}_{j \cdot q+r+1})$. It follows that $\min_{r=1,...,q}\{\Delta_r\} \le \frac{2}{q} \cdot \texttt{IoA}(\mathcal{P}^*)$. One can then verify that $\texttt{IoA}(\mathcal{P}_{r^*}) \ge (1 - \frac{2}{q}) \cdot \texttt{IoA}(\mathcal{P}^*)$ where $r^* = \arg\min_{r=1,...,q}\{\Delta_r\}$. Lastly, let $\hat{\mathcal{P}}$ be an assignment returned by the algorithm, $\hat{\mathcal{P}} = \arg\max_r \texttt{IoA}(\mathcal{P}_r)$. It follows that $\texttt{IoA}(\hat{\mathcal{P}}) \ge (1 - \frac{2}{q}) \cdot \texttt{IoA}(\mathcal{P}^*)$. ■

7 EXPERIMENTAL EVALUATION

We evaluate the empirical performance of the proposed local improvement algorithm for IM-IoA under several scenarios. Our results demonstrate the high effectiveness of the algorithm on both synthetic and real-world networks.

7.1 Experimental setup

Networks. We select networks based on their sizes and application domains, as shown in Table 1. Specifically, Gnp and Power-law are synthetic networks generated using the Erdős-Rènyi [11] and Barabási-Albert [3] models, respectively. City is a synthetic network of a residential area in Charlottesville, obtained from the Biocomplexity Institute at the University of Virginia. In this network, vertices are houses, and any pair of houses within 100 yards are considered as neighbors. Arena and Google+ are mined social networks obtained from a public repository [21].

Network	Type	n	m	*Max deg*
Gnp	Random	1,000	4,975	36
Power-law	Random	1,000	5,015	355
City	Residential	7,444	238,802	165
Arena	Social	10,680	24,316	205
Google+	Social	23,613	39,182	2,761

Table 1: List of networks

Algorithms. We evaluate the performance of Local-Improvement algorithm using the following baselines: (1) Greedy: Initially, all vertices are occupied by type-2 agents; then iteratively k of these are replaced by type-1 agents in a greedy manner. Specifically, in each iteration, a replacement that causes the largest increase in the objective value is chosen. (2) Random: a random subset of k vertices are chosen for type-1 agents, and the remaining vertices are assigned to type-2 agents.

Evaluation metrics. We use two metrics to quantify the performance: (i) the *integration ratio* $\mu = obj/n$ (i.e., the fraction of integrated agents) and (ii) the *empirical approximation ratio* $\gamma = obj/OPT$ where OPT is the optimal value. The value OPT is computed by solving an integer linear program (ILP) using Gurobi [29].

Reproducibility. The source code and selected datasets are at https://github.com/bridgelessqiu/Integration_Max.

7.2 Experimental results

We present an overview of the results under the following experimental scenarios.

Empirical ratio across networks. We first study the empirical approximation ratio γ of the algorithms on different networks. For the three large networks, namely City, Arena and Google+, the

ILP solver didn't terminate even though it was run for 24 hours. Therefore, we restricted our focus to *smaller subgraphs* of these networks. For each subgraph, we fixed the number k of minority agents to be 10% of n, where n is the number of vertices in the network. The empirical ratio for each algorithm is then averaged over 100 repetitions.

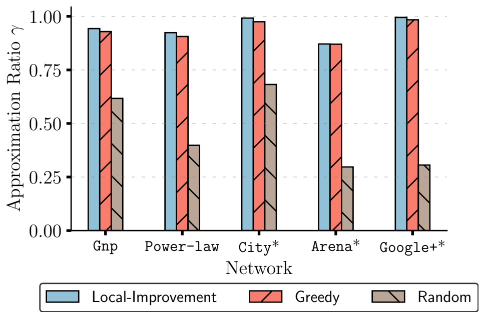
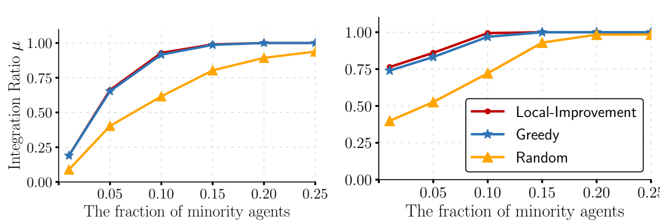


Figure 3: The empirical approximation ratio γ for algorithms. The number of vertices and edges (n, m) for each subgraph are as follows. `City*`: $(1607, 50112)$, `Arena*`: $(1981, 9132)$, `Google+*`: $(2000, 5042)$.

Representative results for the empirical ratio are shown in Fig. 3. Overall, we observe that the performance of Local-Improvement and Greedy are close to the optimal value, with Local-Improvement outperforming Greedy by a small margin. Specifically, the empirical ratio of Local-Improvement is greater than 0.85 on all tested instances. As one would expect, the empirical ratio of Random is much lower than its counterparts. Overall, we note that the empirical ratio of Local-Improvement is much higher than its theoretical guarantee of 1/2. Recall from Section 4 that there are instances where Local-Improvement produces solutions that are of 1/2 of the optimal value. Our experimental findings indicate such worst-case instances did not occur in these experiments. We also note that empirically Greedy is comparable to Local Improvement. However, no known performance guarantee for Greedy has been established. In contrast, as shown in Section 4, Local Improvement provides a guarantee of 1/2.



(a) `Gnp` network **(b) `City` network**

Figure 4: The change of the fraction of integrated agents as the fraction of minority agents increases. The networks are `Gnp` and `City` shown in Table (1).

Variations on the number of minority agents. Next, we study the integration ratio μ (i.e., the fraction of integrated agents) obtained by the algorithms under the scenario where the fraction of minority agents (k) increases from 0.01 to 0.25. The representative results for `Gnp` and `City` networks are shown in Fig. 4. Overall, we observe that as the fraction of minority agents increases, the integration ratio μ grows monotonically for all algorithms. Similar results are observed for all the chosen networks. Despite the monotonicity observed in the experiments, we remark that the objective value that an algorithm can obtain is general non-monotone as k increases. (A simple example is a star where the objective is maximized for $k = 1$ when the type-1 agent is placed at the center. It is easy to verify that as k increases, the optimal objective decreases.)

Change of objective as local improvement proceeds. Lastly, we study the increase in the objective value as the number of swaps used in Local-Improvement is increased. Results are shown in Fig. 5 for gnp networks with 1000 nodes and average degrees varying from 10 to 30. Overall, we observe a linear relationship between the objective value and the number of swaps.

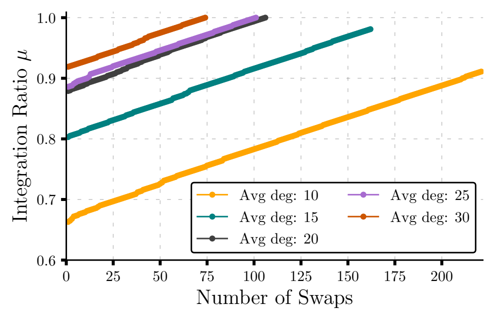


Figure 5: The change in the number of integrated agents as Local-Improvement proceeds. The underlying `gnp` networks have $1,000$ vertices; the average degree varies from 10 to 30.

8 CONCLUSIONS

We considered an optimization problem that arises in the context of assigning agents to the nodes of a network to maximize the integration level. Since the general problem is **NP**-hard, we presented approximation algorithms with provable performance guarantees for several versions of the problem. Our work suggests several directions for further research. First, it is of interest to investigate approximation algorithms with better performance guarantees for the general problem. One possible approach is to consider local improvement algorithms that, instead of swapping just one pair of vertices to increase the number of integrated vertices, swap up to j pairs, for some fixed $j \geq 2$ in each iteration. One can also study the problem under network-based extensions of other integration indices proposed in the social science literature [25]. Another direction is the scenario where the total number of agents is less than the number of vertices (so that some vertices remain unoccupied by agents). In addition, one can also study the variant where there are agents of three or more types, and the notion of integration is defined by requiring the neighborhood of an agent to include a certain number of agents of the other types. Overall, this topic offers a variety of interesting new problems for future research.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for reading the paper carefully and providing valuable feedback. We also thank the Biocomplexity Institute at the University of Virginia for providing the dataset used in our experimental study. All the authors are also affiliated with the Biocomplexity Institute and Initiative at the University of Virginia. This work is partially supported by University of Virginia Strategic Investment Fund award number SIF160, VDH Grant PV-BII VDH COVID-19 Modeling Program VDH-21-501-0135 and NSF Grants OAC-1916805 (CINES), CCF-1918656 (Expeditions), DTRA subcontract/ARA S-D00189-15-TO-01-UVA and NSF IIS-1908530.

REFERENCES

[1] Aishwarya Agarwal, Edith Elkind, Jiarui Gan, and Alexandros Voudouris. 2020. Swap stability in Schelling games on graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI Press, (online), 1758–1765.

[2] Brenda S Baker. 1994. Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM (JACM)* 41, 1 (1994), 153–180.

[3] Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. *Science* 286, 5439 (1999), 509–512.

[4] Nawal Benabbou, Mithun Chakraborty, Xuan-Vinh Ho, Jakub Sliwinski, and Yair Zick. 2018. Diversity constraints in public housing allocation. In *17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2018)*. IFAAMAS, Hosted by University of Liverpool Computer Science Department, 973–981.

[5] Hans L Bodlaender. 1998. A partial k-arboretum of graphs with bounded treewidth. *Theoretical computer science* 209, 1-2 (1998), 1–45.

[6] Johanne Boisjoly, Greg J. Duncan, Dan M Levy, Michael Kremer, and Jacque Eccles. 2006. Empathy or Antipathy? The Impact of Diversity. *The American Economic Review* 96, 5 (2006), 1890–1905.

[7] Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. 2014. Submodular maximization with cardinality constraints. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, Philadelphia, PA, 1433–1452.

[8] NYU Furman Center. 2019. How NYCHA Preserves Diversity in New York's Changing Neighborhoods. https://furmancenter.org/files/NYCHA_Diversity_Brief_Final-04-30-2019.pdf.

[9] Kendra S Cheruvelil, Patricia A Soranno, Kathleen C Weathers, Paul C Hanson, Simon J Goring, Christopher T Filstrup, and Emily K Read. 2014. Creating and maintaining high-performing collaborative research teams: the importance of diversity and interpersonal skills. *Frontiers in Ecology and the Environment* 12, 1 (2014), 31–38.

[10] Parliament Debates. 1989. Official Report.(1989, February 16). *Better racial mix in HDB housing estates* 52 (1989), 650–651.

[11] Paul Erdős and Alfréd Rényi. 1959. On random graphs I. *Publicationes Mathematicae* 6, 1 (1959), 290–297.

[12] Glenn Firebaugh and Francesco Acciai. 2016. For blacks in America, the gap in neighborhood poverty has declined faster than segregation. *Proceedings of the National Academy of Sciences* 113, 47 (2016), 13372–13377.

[13] Alan Frieze and Mark Jerrum. 1997. Improved approximation algorithms for max-k-cut and max bisection. *Algorithmica* 18, 1 (1997), 67–81.

[14] Michael R Glass and Anna E Salvador. 2018. Remaking Singapore's heartland: sustaining public housing through home and neighbourhood upgrade programmes. *International Journal of Housing Policy* 18, 3 (2018), 479–490.

[15] Michel X Goemans and David P Williamson. 1995. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)* 42, 6 (1995), 1115–1145.

[16] L. E. Gomez and Patrick Bernet. 2019. Diversity improves performance and outcomes. *Journal of the National Medical Association* 111, 4 (2019), 383–392.

[17] Olivier Brandouy Gretha, Philippe Mathieu Cristal, and Nicolas Mauhe. 2018. Segregation in social networks: a simple Schelling-like model. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, Los Alamitos, CA, 95–98.

[18] Ravit Hananel. 2017. From central to marginal: The trajectory of Israel's public-housing policy. *Urban Studies* 54, 11 (2017), 2432–2447.

[19] Adam Douglas Henry, Paweł Prałat, and Cun-Quan Zhang. 2011. Emergence of segregation in evolving social networks. *Proceedings of the National Academy of Sciences* 108, 21 (2011), 8605–8610.

[20] Rachel Garshick Kleit. 2001. Neighborhood relations in suburban scattered-site and clustered public housing. *Journal of Urban affairs* 23, 3-4 (2001), 409–430.

[21] Jure Leskovec and Andrej Krevl. 2014. SNAP Datasets: Stanford Large Network Dataset Collection. http://snap.stanford.edu/data.

[22] Stanley Lieberson. 1980. An Asymmetrical Approach to Measuring Residential Segregation. Research Paper No. 115, The Centre for Urban and Community Studies, University of Toronto.

[23] Jens Ludwig et al. 2012. Long-term effects of neighborhood environments on low-income families: a summary of results from the Moving to Opportunity experiment. Paris, France: Laboratoire Interdisciplinaire d'évaluation des politiques publiques. , 32 pages.

[24] Jyoti D Mahadeo, Teerooven Soobaroyen, and Vanisha Oogarah Hanuman. 2012. Board composition and financial performance: Uncovering the effects of diversity in an emerging economy. *Journal of business ethics* 105, 3 (2012), 375–388.

[25] Douglas S. Massey and Nancy A. Denton. 1988. The Dimensions of Residential Segregation. *Social Forces* 67, 2 (12 1988), 281–315.

[26] Barrie S. Morgan. 1983. A Distance-Decay Based Interaction Index to Measure Residential Segregation. *Area* 15, 3 (1983), 211–217. http://www.jstor.org/stable/20001935

[27] Moni Naor and Larry Stockmeyer. 1993. What can be computed locally?. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*. ACM, New York, NY, 184–193.

[28] City of Chicago. 2019. One Chicago Housing Strategy for a Thriving City Five-Year Housing Plan. https://www.chicago.gov/content/dam/city/depts/dcd/Housing%20Programs/20733_37_5_Year_Plan_Report_final_WEB_C.pdf.

[29] Gurobi Optimization. 2021. Reference Manual. https://www.gurobi.com/documentation/9.1/refman/index.html.

[30] Craig E Pollack, Harold D Green Jr, David P Kennedy, Beth Ann Griffin, Alene Kennedy-Hendricks, Susan Burkhauser, and Heather Schwartz. 2014. The impact of public housing on social networks: a natural experiment. *American Journal of Public Health* 104, 9 (2014), 1642–1649.

[31] Zirou Qiu, Andrew Yuan, Chen Chen, Madhav V Marathe, S.S. Ravi, Daniel J Rosenkrantz, Richard E Stearns, and Anil Vullikanti. 2022. Assigning Agents to Increase Network-Based Neighborhood Diversity - Full Version. https://arxiv.org/abs/2301.02876.

[32] Neil Robertson and Paul D. Seymour. 1986. Graph minors. II. Algorithmic aspects of tree-width. *Journal of algorithms* 7, 3 (1986), 309–322.

[33] Eva Rosen, Philip M.E. Garboden, and Jennifer E Cossyleon. 2021. Racial discrimination in housing: how landlords use algorithms and home visits to screen tenants. *American Sociological Review* 86, 5 (2021), 787–822.

[34] Thomas C Schelling. 1971. Dynamic models of segregation. *Journal of mathematical sociology* 1, 2 (1971), 143–186.

[35] Hans Skifter Andersen, Roger Andersson, Terje Wessel, and Katja Vilkama. 2016. The impact of housing policies and housing markets on ethnic spatial segregation: Comparing the capital cities of four Nordic welfare states. *International Journal of Housing Policy* 16, 1 (2016), 1–30.

[36] Daryl G Smith and Natalie B Schonfeld. 2000. The benefits of diversity what the research tells us. *About campus* 5, 5 (2000), 16–23.

[37] Georgios Stamoulis. 2014. Approximation algorithms for bounded color matchings via convex decompositions. In *International Symposium on Mathematical Foundations of Computer Science*. Springer, Berlin-Heidelberg, Germany, 625–636.

[38] Tiit Tammaru, Szymon Marcin´ Czak, Raivo Aunap, Maarten van Ham, and Heleen Janssen. 2020. Relationship between income inequality and residential segregation of socioeconomic groups. *Regional Studies* 54, 4 (2020), 450–461.

[39] Neil Thakral. 2016. The public-housing allocation problem. Technical report, Harvard University.

[40] Maarten Van Ham, Masaya Uesugi, Tiit Tammaru, David Manley, and Heleen Janssen. 2020. Changing occupational structures and residential segregation in New York, London and Tokyo. *Nature human behaviour* 4, 11 (2020), 1124–1134.