Proceedings of the International Workshop on Citizen-Centric Multiagent Systems 2023 (CMAS'23)

Co-located with the International Conference on Autonomous Agents and Multiagent Systems (AAMAS'23)

London, ExCeL Conference Centre

Sebastian Stein[1], Natalia Criado[2], Behrad Koohy[1], Kate Larson[3], Marija Slavkovik[4], and Vahid Yazdanpanah[1]

[1]University of Southampton
[2]Universitat Politecnica de Valencia
[3]University of Waterloo
[4]University of Bergen

30 May 2023

Large-scale AI systems promise to address important societal challenges, such as decarbonising our energy system, transitioning to on-demand mobility or responding effectively to disasters. However, citizen end users are often seen as peripheral to these systems, assumed to be passively providing data and consuming services. The goal of this workshop on citizen-centric multiagent systems (C-MAS) is to explore alternative approaches that treat citizen end users as first-class agents with diverse needs and preferences, thus enabling more trustworthy, fairer and potentially more widely accepted sociotechnical solutions to pressing societal challenges.

C-MAS 2023 will draw on the substantial body of work within multiagent systems on how to model, design and reason about complex systems of interacting self-interested agents, which may include citizen end users, service providers, governmental bodies and other stakeholders. It will also build on emerging techniques from human-centred AI to promote fairness and to enable explainability. This workshop will be relevant for researchers, both in industry and academia, whose research affects and involves citizens end non-expert users.

Further details are available at: https://sites.google.com/view/cmas23

Contents

1 Keynote: Socially Intelligent Civic Infrastructure and Challenges for AI Ethics

Keynote by Munindar Singh, North Carolina State University

Advances in technology are leading to a shift from traditional to smart civic infrastructure, one driven by data and seeking to optimize resource usage. I posit that this shift, though desirable, is not enough. I motivate the conception of socially intelligent civic infrastructure, one that adaptively deals with multiparty requirements, is user-centric, satisfices individual and societal objectives, and provides affordances for cooperation. In so doing, the envisioned infrastructure supports and benefits from users' social intelligence by revealing to themselves and others the externalities of their decisions and promoting prosocial attitudes (empathy) and behaviours (cooperation) between users. As envisioned, socially intelligent civic infrastructure would not only serve user needs but also shape their preferences toward societally desirable outcomes such as sustainability.

2 Multiagent Systems for Citizen-Centric Applications

2.1 Towards Reducing School Segregation by Intervening on Transportation Networks

Towards Reducing School Segregation by Intervening on Transportation Networks

Dimitris Michailidis, Mayesha Tasnim, Sennay Ghebreab, and Fernando P. Santos

Civic AI Lab, Socially Intelligent Artificial Systems
Informatics Institute, University of Amsterdam
{d.michailidis, m.tasnim, s.ghebreab, f.p.santos}@uva.nl

Abstract. Urban segregation is a complex phenomenon associated with different forms of social inequality. Segregation is reflected in parents' school preferences, especially in context of free school choice modes. Studies have shown that parents consider both distance and demographic composition when selecting schools for their children, potentially exacerbating levels of residential segregation. This raises the question of how intervening on transit networks — thereby affecting school accessibility to citizens belonging to different groups — can alleviate spatial segregation. In this work-in-progress paper, we propose a new agent-based model to explore this question. Conducting experiments in synthetic and real-life scenarios, we show that improving access to schools via transport network interventions can lead to a reduction in school segregation over time. The mathematical framework we propose provides the basis to simulate, in the future, how the dynamics of citizens preferences, school capacity and public transportation availability might contribute to patterns of residential segregation.

Keywords: Transportation Networks · One-sided Matching · Agent-based Simulations · Dynamic Preferences

1 Introduction

Urban segregation is a complex phenomenon that reverberates across multiple socio-economic contexts — from social mobility to educational opportunities. In the context of education, centralized school admissions systems such as *Deferred Acceptance* and *Random Serial Dictatorship* have been popularized across the world for their simplicity and fairness in student allocation [8, 2]. However, school segregation can emerge in such preference-based systems, reflecting (or even amplifying) existing residential segregation patterns [6]. There is evidence that parents do not send their children to schools in their residential neighborhoods; if they did, schools would be less segregated than how they currently are [12].

Although parents might prefer schools outside their neighborhoods, distance and commuting time are important factors for attending a school [6]. With the exception of high-income households, most do not tend to move house and thus

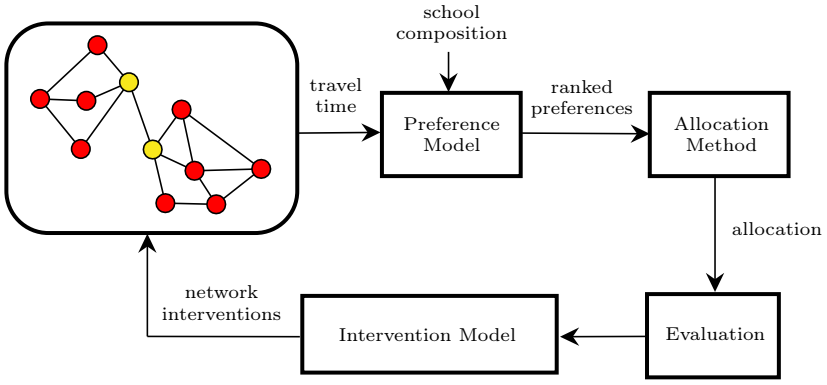

Fig. 1. Proposed agent-based model to study the impact of transport network interventions on school segregation. We consider an environment where citizens, schools, and a transportation graph are distributed in space (Section 2.1). At each round, agents A generate preferences for schools F, using a preference model (section 2.2). Agents are assigned to schools via an allocation method (Section 2.3), which is evaluated on segregation (Section 2.4). An intervention model creates edge-based interventions to the transportation network, aiming to improve segregation (Section 2.5).

their choice is limited by their location [3]. Intervening on public transportation networks can thereby affect segregation, by allowing citizens from different societal groups to attend a wider set of schools. This raises a natural question: *Can transportation networks be designed, or extended, to efficiently reduce school segregation?*

Here we resort to agent-based modeling (ABM) to explore the previous question. Prior studies focused on the complexity of residential and school segregation via ABMs [15, 6], and preference models based on both school composition and distance have been explored [6, 14]. However, these works do not study the effect of strategically increasing accessibility to specific schools. Graph-based interventions have been utilized before to reduce accessibility inequality [10], but not to tackle school segregation. We assess whether graph-based transportation interventions can be used to reduce disparities in group composition within schools, under a centralized admission system.

We test transport network intervention strategies based on greedy optimization of classic graph centrality measures such as *closeness*, *betweenness*, and *degree* centrality. We conduct experiments in a synthetic and a real-life environment in the city of Amsterdam and show that targeted interventions can lead to a significant reduction in segregation over time.

2 Methods

2.1 Environment: Citizens, Transportation, and Schools

We model the environment as an undirected graph $\mathbb{G} = (V, E)$, where $V = \{v_1, ..., v_{n_v}\}$ are nodes, one for each census tract in the city, and $E = \{e_{i,j}\}, i, j \in V, i \neq j$ are edges that represent transportation connections between nodes. For the sake of simplicity, the edges are unweighted, but the model can be used with weighted edges too (e.g., representing transportation times). We define the shortest path between i and j as $t_{i,j},\ i, j \in V$.

We define a set of N agents (citizens), $A = \{a_1, ..., a_N\}$. An agent is characterized by its residence node $v_a \in V$. Each agent is belongs to a group $g \in G$, defined based on characteristics such as ethnicity, income, or other socio-economic status. Finally, each agent has a homophily attribute, $h_i \in [0, 1]$, defining a preference for an optimal fraction of agents from the same group attending a school [6, 11]. Note that agents are abstract entities that represent students in a city.

We define schools $f \in F$, which are located in nodes $v_f \in V$. Each school is associated with a capacity (maximum number of allowed agents) $s_f \in [0, N]$ and a group composition (fraction of assigned agents from each group) $c_{g,f} \in [0, 1],\ g \in G$. Note that $\sum_g c_{g,f} = 1,\ \forall f \in F$.

2.2 Preference Model

At every round, each agent $a_i \in A$ creates a preference list $P_i \subseteq F$, over schools. Each school appears once. The preference list is based on a utility function $U_{if}, f \in F$, and schools are sorted in descending order. We adopt the widely used Cobb-Douglas utility function, based on a function of school composition $C : c_{g,f} \to \mathbb{R}$ and travel time from the agent's residence to the school $t_{i,f}$ [6, 14]

$$U_{i,f} = c_{g,f}^{\alpha}\ t_{i,f}^{(1-\alpha)}, \tag{1}$$

where g denotes the group that agent a_i belongs to and $0 \leq \alpha \leq 1$ is a parameter that controls the weight of the group composition over the travel time. Travel time is normalized by the maximum value and is calculated as follows [6]:

$$t'_{i,f} = \begin{cases} \frac{t_{max,i} - t_{i,f}}{t_{maxi} - t_{min,i}}, & \text{if } t_{i,f} \leq t_{max,i} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

For the school composition, we use a single-peaked utility function, that is maximized when the number of agents of the same group in a school $x_{g,f}$ is equal to the homophily attribute h_i [6, 14]. Values above h_i incur a constant penalty M:

$$C(x_{g,f}, h_i, M) = \begin{cases} \frac{x_{g,f}}{h_i}, & \text{if } x_{g,f} \leq h_i \\ M + \frac{(1-x_{g,f})(1-M)}{1-h_i}, & \text{if } x_{g,f} > h_i \end{cases} \tag{3}$$

M controls the level of dissatisfaction when the fraction of similar agents exceeds the optimal h_i. With this formulation interventions in the transportation network are performed to reduce the travel time $t_{i,f}$ of agents to school, with the goal of increasing utility towards more segregated schools.

2.3 Allocation Method

Once the preference lists P have been generated at each simulation round for all $a_i \in A$, they are then provided as input to an allocation method R. R is defined as a function $R : P \rightarrow F$ which takes as input a preference list p_i for agent a_i and capacity s_f for all $f \in F$ and assigns a school $f_i \in p_i$. Random Serial Dictatorship (RSD) is a popular mechanism for one-sided matching between schools and students [2]. In RSD a lottery number is first uniformly drawn for each student. The students are then serially allocated to the top-preferred school with remaining capacity in increasing order of the lottery. For our simulations we implement RSD and perform allocations at every round; schools have, overall, capacity to allocate all students, i.e., $\sum s_f \geq N$. Additionally, for each student the preference model from Section 2.2 provides a ranking for all schools, and RSD can allocate all students. The allocation result is then aggregated for evaluation.

2.4 Allocation Evaluation

After each simulation round, the allocation of agents to schools is evaluated on segregation. To measure segregation, we use the Dissimilarity Index (DI), a measure that captures the differences in the proportions of agents from two groups assigned to a school [7]. DI has been widely used in assessing segregation, as it takes into account the total number of agents from each group, making it suitable to use even when one group is a minority [1]. DI is defined as follows:

$$DI = \frac{1}{2} \sum_{f=1}^{|F|} |\frac{g_{1,f}}{G_1} - \frac{g_{2,f}}{G_2}|, \quad DI \in [0, 1] \tag{4}$$

Where $g_{j,f}$ is the number of agents of group j in school f; G_j is the number of agents in group j. Segregation is minimum (maximum) when $DI = 0$ ($DI = 1$).

2.5 Intervention Model

We explore the impact that intervening on public transport networks has on school choices. By improving transportation, we aim to elevate the rank of schools composed of majority groups in the preference lists of minority groups, increasing their accessibility to popular (yet distant) schools. Transport interventions are performed in the form of graph augmentations, by creating a new edge set $E' : \mathbb{G}, B \rightarrow \mathbb{G}'$ to the spatial graph, under a budget B [10]. It follows that $\mathbb{G}' = (V, E \cup E')$. Interventions can be seen as a proxy to the creation/expansion of

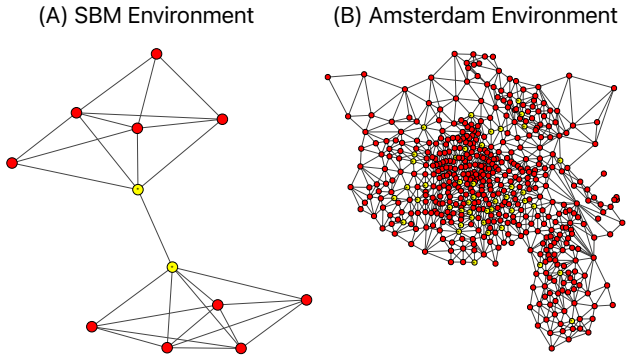

Fig. 2. We study synthetic (left image) and real (right image) environments. Nodes represent neighborhoods and yellow nodes (marked with *) indicate nodes with schools.

public transportation lines in a real city, such as bus, metro or tram. We constrain the total number of interventions to a budget B, reflecting resource limitations.

The goal of interventions is to find the best set of edges E' to add to the graph, such that total segregation is reduced. Segregation depends on the allocation method (section 2.3), which has a random element to it. Therefore, optimizing directly for the dissimilarity index is not possible. We look for targeted interventions that increase accessibility to certain schools for certain groups, aiming to affect the agent's preferences in such a way that segregation is reduced.

We test two classes of greedy interventions: 1) **Centrality** and 2) **Group-based Centrality Optimization**. We identify the schools that have the lowest network centrality measure (*closeness*, *betweenness* or *degree*) [4] with respect to any group and then add the intervention that leads to the maximum increase in that node's corresponding 1) centrality or 2) group-based centrality.

3 Experimental Setup

We perform experiments on two graph environments: a real-life city environment based on Amsterdam neighborhoods, demographic and transportation data; and a synthethic environment based on the stochastic block model (SBM) [9], which allows us to have full control over the level of modularity and segregation in a hypothetical city. For more details please refer to Appendix B and Fig. 2.

4 Preliminary Results

In Figure 3, we present the 95% confidence interval of the Dissimilarity Index on each simulation round, over a total of 50 rounds. Our preliminary experiments

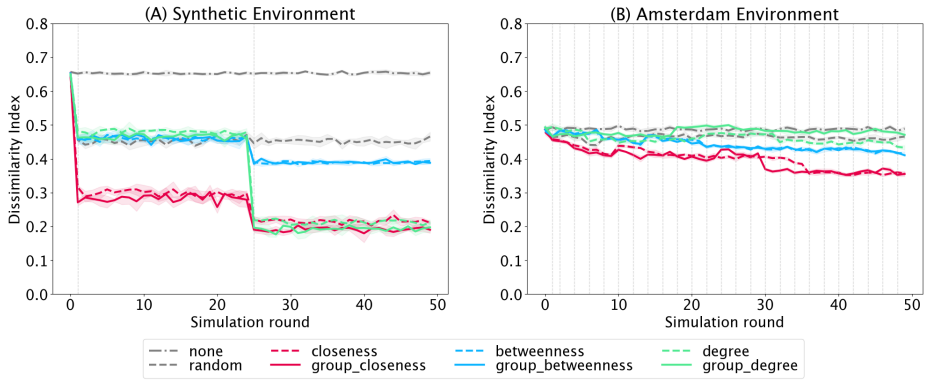

Fig. 3. We show that targeted interventions in the network can significantly decrease segregation over time. Strategies based on closeness perform best over other centrality measures. Vertical dashed lines indicate rounds with graph interventions.

show that, under the settings outlined above, all targeted network intervention strategies proposed in Section 2.5 lead to a significant reduction of segregation over time, when compared to a no-intervetion scenario (none) or random interventions. Specifically, we observe that greedy interventions aimed at increasing the closeness of the least-accessible nodes lead to the highest reduction of segregation over time. We also observe that degree-based interventions can have similar effects to closeness, but only in small networks, like SBM. This is because, when the number of nodes is low, increasing a the degree of a node also increases its closeness to other nodes. A betweenness-based strategy reduces segregation and outperforms degree-based ones in a bigger environment, like that of Amsterdam. Finally, there are seemingly not big differences between centrality and group-based centrality strategies, but depending on the budget, group-based closeness can outperform its classic counterpart.

5 Conclusion and Future Work

In this work-in-progress paper, we used an agent-based simulation model to study the impact of transport network interventions on school segregation, under the prevalence of a centralized school choice algorithm. We have demonstrated in both a synthetic and a real-life environment that, by affecting citizens preferences for particular schools, targeted transportation interventions can ultimately reduce school segregation over time. In the future, we plan to further experiment with the parameters of the preference model, to assess the sensitivity of network interventions to different types of agent school preferences. We plan to further experiment with group-based interventions, aiming at identifying the contexts where they become more efficient than centrality-based interventions.

Acknowledgements This research was supported by the Innovation Center for AI (ICAI, The Netherlands) and the City of Amsterdam.

References

1. Abbasi, S., Ko, J., Min, J.: Measuring destination-based segregation through mobility patterns: Application of transport card data. Journal of Transport Geography **92**, 103025 (Apr 2021). https://doi.org/10.1016/j.jtrangeo.2021.103025
2. Abdulkadiroğlu, A., Sönmez, T.: Random serial dictatorship and the core from random endowments in house allocation problems. Econometrica **66**(3), 689–701 (1998)
3. Boterman, W.R.: Socio-spatial strategies of school selection in a free parental choice context. Transactions of the Institute of British Geographers **46**(4), 882–899 (2021). https://doi.org/10.1111/tran.12454
4. Chen, D., Lü, L., Shang, M.S., Zhang, Y.C., Zhou, T.: Identifying influential nodes in complex networks. Physica a: Statistical mechanics and its applications **391**(4), 1777–1787 (2012)
5. Crescenzi, P., D'angelo, G., Severini, L., Velaj, Y.: Greedily Improving Our Own Closeness Centrality in a Network. ACM Transactions on Knowledge Discovery from Data **11**(1), 9:1–9:32 (Jul 2016). https://doi.org/10.1145/2953882
6. Dignum, E., Athieniti, E., Boterman, W., Flache, A., Lees, M.: Mechanisms for increased school segregation relative to residential segregation: a model-based analysis. Computers, Environment and Urban Systems **93**, 101772 (Apr 2022). https://doi.org/10.1016/j.compenvurbsys.2022.101772
7. Duncan, O.D., Duncan, B.: A Methodological Analysis of Segregation Indexes. American Sociological Review **20**(2), 210–217 (1955). https://doi.org/10.2307/2088328, publisher: [American Sociological Association, Sage Publications, Inc.]
8. Erdil, A., Ergin, H.: What's the matter with tie-breaking? improving efficiency in school choice. American Economic Review **98**(3), 669–689 (2008)
9. Holland, P.W., Laskey, K.B., Leinhardt, S.: Stochastic blockmodels: First steps. Social Networks **5**(2), 109–137 (Jun 1983). https://doi.org/10.1016/0378-8733(83)90021-7
10. Ramachandran, G.S., Brugere, I., Varshney, L.R., Xiong, C.: GAEA: Graph Augmentation for Equitable Access via Reinforcement Learning. arXiv:2012.03900 [cs] (Apr 2021), arXiv: 2012.03900
11. Schelling, T.C.: Dynamic models of segregation. The Journal of Mathematical Sociology **1**(2), 143–186 (Jul 1971). https://doi.org/10.1080/0022250X.1971.9989794, publisher: Routledge _eprint: https://doi.org/10.1080/0022250X.1971.9989794
12. Sissing, S., Boterman, W.R.: Maintaining the legitimacy of school choice in the segregated schooling environment of Amsterdam. Comparative Education **59**(1), 118–135 (Jan 2023). https://doi.org/10.1080/03050068.2022.2094580
13. Sousa, S., Nicosia, V.: Quantifying ethnic segregation in cities through random walks. Nature Communications **13**(1), 5809 (Oct 2022). https://doi.org/10.1038/s41467-022-33344-3, number: 1 Publisher: Nature Publishing Group
14. Stoica, V.I., Flache, A.: From Schelling to Schools: A Comparison of a Model of Residential Segregation with a Model of School Segregation. Journal of Artificial Societies and Social Simulation **17**(1), 5 (2014)

15. Zuccotti, C.V., Lorenz, J., Paolillo, R., Rodríguez Sánchez, A., Serka, S.: Exploring the dynamics of neighbourhood ethnic segregation with agent-based modelling: an empirical application to Bradford, UK. Journal of Ethnic and Migration Studies **49**(2), 554–575 (Jan 2023). https://doi.org/10.1080/1369183X.2022.2100554

Appendix

A Intervention Methods

Section 2.5 introduces the design choice to test two classes of greedy algorithms in the intervention model of the ABM. The algorithms and their usage as intervention methods are discussed below:

A.1 Greedy Centrality Optimization

Making a school more accessible is a non-trivial optimization problem, especially for large graphs [5]. We use a greedy algorithm to approximate the optimal set of interventions to apply to the graph with respect to accessibility. This translates to increasing a school node centrality $\mathbb{C}$ with respect to the other nodes. We evaluate strategies based on the classic graph measures of *closeness* ($\mathbb{C}_C$), *betweenness* ($\mathbb{C}_B$), and *degree* ($\mathbb{C}_D$) centrality.

At every intervention step, we find the school that has the lowest centrality measure with respect to any group and then add the intervention that leads to the maximum increase in this node's centrality. The process is described in Algorithm 1.

Algorithm 1 Greedy Centrality Optimization

Input $\mathbb{G} = (V, E)$

$E' \leftarrow \{\}$
for $b = 1, 2, ...B$ **do**
  $v_{g_{min}} \leftarrow argmin\{\mathbb{C}(v, g) \mid v \in V,\ g \in G\}$
  $\mathbb{C}_{max} = 0$
  $e_{max} \leftarrow null$
  for $u \in V,\ u \neq V$ **do**
    $e \leftarrow (u, v)$
    Compute $\mathbb{C}(v_{g_{min}}, E \cup E' \cup e)$
    if $\mathbb{C}(v_{g_{min}}, E \cup E' \cup e) > \mathbb{C}_{max}$ **then**
      $\mathbb{C}_{max} = \mathbb{C}(v_{g_{min}}, E \cup E' \cup e)$
      $e_{max} \leftarrow e$
    end if
  end for
  $E' \leftarrow E' \cup e_{max}$
end for
Output $\mathbb{G}' = (V, E \cup E')$

A.2 Group-based Centrality

Classic centrality measures fail to capture group dynamics in a graph. In segregated environments like cities, central areas can exhibit high closeness centrality, despite having low accessibility to specific groups. Examples of this phenomenon include cities where low-income households concentrate in the outskirts, while high-income households are situated closer to the center. To account for this disparity in measurement, we introduce group-based extensions of the classic centrality measures $\mathbb{C}^g,\ g \in G$, that take into account the distribution of groups within nodes. These are namely group-based closeness $\mathbb{C}^g_C$, betweenness $\mathbb{C}^g_B$ and degree $\mathbb{C}^g_D$. Let $D_g,\ g \in G$ be the distribution of group g on all nodes V in the network such that $\sum_g D_g = 1$.

Group-based Closeness Centrality Group-based closeness $\mathbb{C}^g_C$ of a node $v \in V$ is defined as the reciprocal of the sum of travel times from all other nodes u, weighted by the fraction of agents of group g in u, $p(g|u)$.

$$\mathbb{C}^g_C(v) = \sum_u \frac{1}{t(u,v)\ p(g|u)} \tag{5}$$

Where $t(u,v)$ is the travel time between nodes u and v.

Group-based Betweenness Centrality Group-based betweenness $\mathbb{C}^g_B$ of a node $v \in V$ is defined as the number of shortest paths σ from all nodes $o \in V$ to all nodes $d \in V, o \neq d$, that pass through v, weighted by the fraction of agents of group g in d. $p(g|d)$.

$$\mathbb{C}^g_B(v) = \sum_{o \neq v \neq d} \frac{\sigma_{t_{o,d}(v)}}{\sigma_{t_{o,d}}} p(g|d) \tag{6}$$

Group-based Degree Centrality Group-based degree $\mathbb{C}^g_D$ of a node $v \in V$ is defined as the total number of edges connected to a node $E_v = e_{u,v}$, $u \in V, u \neq v$, weighted by the fraction of agents of group g in u, $p(g|u)$.

$$\mathbb{C}^g_D(v) = \sum_{u \in V, u \neq v, e_{u,v} \in E} p(g|u) \tag{7}$$

Optimizing for group-based centrality measures leads to interventions that target schools where specific groups are underrepresented, instead of arbitrarily increasing the centrality of a school.

B Simulation Environments

We perform experiments on two graph environments, a synthetic stochastic block model (SBM) [9] and a real city environment based on Amsterdam, Netherlands.

SBM Environment The SBM graph is specifically generated to form clusters of communities, where nodes are densely connected with other nodes in their community and scarcely connected with nodes outside of it. We generated an SBM graph of $n_v = 12$ nodes and $n_e = 27$ edges; nodes clustered in 2 communities, which represent the majority group of their respective nodes. The parameters are chosen speficially to create a highly segregated graph, in which we aim to study the impact of the proposed intervention strategies. In-community edge probability is set to 0.7 and out-community probability is set to 0.01. In Figure 2 (A) we show the realization of the SBM graph we used for the simulation.

Further, we generated a population of $N = 1000$ agents and sampled both their residence node and their group membership, from a total of 2 groups. Group samples are chosen in such a way that each group, within their respective community has a majority of ≥ 0.8 and outside of their community a minority of ≤ 0.2. Since agents do not start at random nodes and there is no moving action in the model, we assume that the optimal fraction of similar agents is equal to the fraction of the majority group of each node. Formally, the homophily parameter of an agent i in a node v is set to $h_{i,v} = \max\{c_{g,v}\},\ g \in G$, where $c_{g,v}$ is the composition of group g in node v.

Finally, we place two schools on the graph, located in the two most connected nodes of the SBM graph. The initial group composition of each school is set to be equal to the group composition of the node it is located in.

Amsterdam Environment To model the real-life environment of Amsterdam, we create a graph where census tracts are converted to nodes, which are connected with their neighboring tracts via an unweighted edge. This graph structure has recently been used to quantify segregation because it provides a scale-free and generalizable method [13]. In total, the graph consists of $n_v = 517$ nodes and $n_e = 1611$ edges. In Figure 2 (B) we show the graph used for the Amsterdam experiments.

Similar to SBM, we generate a population of $N = 7000$ agents. However, in this environment, agents are generated to represent the real-life population of Amsterdam and are split in groups of western (W) and non-western (NW) ethnic background. More details on the population can be found in Table B. Here the homophily parameter is set in the same way as in the SBM environment.

We use the publicly available Amsterdam secondary school dataset provided by DUO[1] which contains 47 secondary schools and their locations. We combine this information with the admissions dataset collected by OSVO[2] which provides

[1] Education Executive Agency: http://duo.nl
[2] The association of school boards in Amsterdam: https://www.verenigingosvo.nl/

the capacities for each school based on the admission results of the previous year.

Table 1. Parameters used in running the experiments.

	SBM	**Amsterdam**
Groups	g0, g2	W, NW
Total Population	1000	7000
Group Populations	524, 476	4547, 2453
Group Populations (%)	52%, 48%	65%, 35%
No. of Nodes	12	517
No. of Edges	27	1611
α, M	0.2, 0.6	0.2, 0.6
Budget (B)	1	1
Simulation Rounds	50	50
Allocation Rounds	5	5
Interventions	2	25

B.1 Simulation Parameters

For the experiments shown in this work, we follow the setup of Dignum et al. and set the relative weight of the composition in the preference model to $\alpha = 0.2$ and the constant $M = 0.6$ for both environments. All experiments are run over 50 simulation rounds, with 5 random serial dictatorship allocations at each environment. We perform 2 intervention rounds in the SBM environment with a budget of $B = 1$ each, while in Amsterdam, we perform a total of 25 intervention rounds, also with $B = 1$. Parameters including total number of intervention rounds and budget B are determined beforehand. Other parameters and their values used in the experimental studies are listed in Table 1.

At every simulation step of the agent-based model agents submit preferences and are allocated to schools. However, interventions are applied to the transport network in intervals. We evaluate the performance of the intervention strategies against a *null* baseline, where no interventions are being done, and against a *random* baseline, where interventions are performed randomly.

2.2 A Task Delegation Model for Delegation Chains

A Task Delegation Model for Delegation Chains*

Jeferson José Baqueta[0000−0002−6646−6888] and
Cesar A. Tacla[0000−0002−8244−8970]

Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial,
Universidade Tecnológica Federal do Paraná (UTFPR), Brazil
jefersonbaqueta@gmail.com, tacla@utfpr.edu.br

Abstract. Task delegation is a common practice adopted in a multi-agent system (MAS) to solve complex problems, allowing the agents to delegate tasks to one another. In literature, the task delegation problem is generally addressed as an isolated process (mono-episodic), which does not consider the formation of delegation chains. This work presents a task delegation model that explicitly considers the agents' sub-delegations and the delegation chains formed from this process for estimating the agents' trustworthiness. In our experiments, we discuss how our delegation model can be configured to cope with different degrees of penalization in case of failure, considering the agents' positions in a delegation chain. Our results show that such a configuration may improve the agents' performance compared to mono-episodic scenarios.

Keywords: Trust · Task delegation · Delegation chains.

1 Introduction

Task delegation is a fundamental mechanism adopted by agents to solve problems that involve teamwork. A critical issue for this kind of application is trust establishing, where the agents need to estimate the trustworthiness of their partners based on social evaluations and environmental conditions. In literature, most works about computational trust cope with the task delegation from a mono-episodic point of view, ignoring the possibility of sub-delegations and the formation of delegation chains [9] [6] [7]. Delegation chains admit the representation of complex social structures built through dependence relations established by the agents as they sub-delegate tasks to one another [13]. These relations affect how the trust is estimated and, consequently, the agents' partner selection.

In this work, we present a task delegation model that considers the delegation chains formed by agents and the social relations established among them, such as AND- and OR-dependencies [13], for selecting trustworthy agents. The trust is computed based on the agents' success rate and competencies concerning the performed tasks. In turn, the partner selection is modeled as a multi-armed bandit (MAB) approach [17], where the agents are selected based on their success likelihood of completing a given task. In our experiments, we discuss the effects

* Supported by CAPES and CNPq (process 409523/2021-6).

of failure propagation in a delegation chain and how to penalize the agents in case of failure based on their positions in the chain. Our results demonstrate the efficiency of our model considering a dynamic scenario from a mono-episodic and sub-delegation point of view.

The rest of this paper is organized as follows. Section II presents the basic concepts adopted in this work. Section III presents our task delegation model, discussing the modeling details. Section IV presents our experiments and the obtained results. The conclusions and future work are summarized in Section V.

2 Background

2.1 Multi-Armed Bandits

Multi-armed bandit (MAB) is a machine learning paradigm that can be seen as the single-state reinforcement learning approach [8]. In the MAB problem, a single agent repeatedly selects one among a finite set of actions aiming to maximize his obtained reward. An action is termed *arm*, while the act of choosing an arm is termed *arm-pull* [3]. Pulling an arm i n-times will yield the rewards $\mu_{i,1}, \mu_{i,2}, ...\mu_{i,n}$, which are independent of each other and associated with the unknown probability distribution. The decision about which arm to pull is based on a MAB policy. Such a policy aims to identify a sequence of arm pulls that maximizes the received rewards over time. One of the most popular MAB policies is the ϵ-*greedy* [16]. This policy selects the arm that yields the highest expected reward with likelihood $1 - \epsilon$ or picks a random arm otherwise [1].

2.2 Social Control

Social control mechanisms allow the agents to make their decisions based on the others' social behavior, offering to agents means to punish undesirable behaviors by themselves [12]. Trust and reputation models are considered good solutions concerning social control [9] [6] [7] [10] [14]. In the computational studies about trust, the decision to trust in someone (partner) is a complex action that can be decomposed into internal and external components [6]. Internal components refer to the partner's attributes, like its competencies and know-how. In contrast, the external components refer to fundamental beliefs from an agent about its partners, and vice-versa, that might change over time due to dynamic conditions (*e.g.*, the emergence of obstacles, adversities, and interference). In particular, the trust an agent places in a partner can be updated through the social evaluations shared by them, such as the social image, reputation, and references. The *social image* consists of evaluative beliefs about a partner's competencies. These beliefs are produced from direct experiences expressing a personal opinion about the partner [12]. *Reputation* is a meta-belief created based on third-party opinions. Its difference regarding the social image is the lack of commitment to the truth, generalization, and loss of reference [14] [12]. At last, *references* can be seen as a type of reputation where a partner can share social evaluations about itself. The partner stores such evaluations as it interacts with other agents, similar to job references [10].

3 Task Delegation Model

A *mono-episodic* delegation scenario is defined by an agent (*delegator*) who wants to achieve its goal g but cannot execute the action a that leads it to accomplish g. Then, the delegator needs to delegate a task τ, specifying the action a that must be performed, to a partner (*delegatee*), who is able to complete τ, which allows the delegator to achieve g [6] [4]. On the other hand, in a context of *sub-delegations* [5], if a delegatee is not able to complete a task τ by itself, it may delegate τ onward until another agent performs it (*recursive delegation*) or decompose τ into sub-tasks $\{\tau_1, \tau_2, ...\tau_n\}$ and then delegate them (*task decomposition*) [11]. Sub-delegations imply the formation of delegation chains where for each new sub-delegation, a delegatee becomes the delegator of a new task, and a new agent is selected as delegatee. Besides, the agent who makes the first delegation request is termed the *root* [1].

We model task delegation as an exploitation/exploration problem considering sub-delegations and the formation of delegation chains. Thus, a delegator must decide if delegating a task to a known agent (exploitation), expecting a likely good outcome, or selecting an unknown partner (delegatee), hoping to get better results (exploration). Therefore, the partner selection process is modeled as a MAB approach [17], where the delegator interacts with its partners (arms that can be pulled) to identify those with the highest likelihood of maximizing the expected reward (the number of tasks executed successfully). A partner can produce a binary reward, getting success by completing the task delegated to it or failure by not completing the task. The success probability of a partner β concerning a task τ is estimated by a delegator α based on two distinct dimensions, the β's competencies regarding τ and its success rate in performing τ over time. This probability represents the α's private trust in β ($TRUST(\alpha, \beta, \tau) \in [0, 1]$) and determines the likelihood of β completing τ successfully.

Competence measure ($Comp(\alpha, \beta, \tau) \in [0, 1]$) is an estimation made by α regarding the abilities and experiences of a partner β concerning some task τ. Such a measure is estimated based on the social image, reputation, and know-how of β. All these components are computed through impressions (social evaluation) produced as the agents interact and evaluate one another. An impression is a 5-tuple $\langle \alpha, \beta, \tau, t, S \rangle$, where α is the delegator who created the impression in the time t, β is the delegatee who executed the task τ, and $S = [(c_1, s_1), ..., (c_n, s_n)]$ is a vector of ratings made by α concerning the β's behavior, in which a pair (c_i, s_i) represents a score $s_i \in [0, 1]$ assigned to β concerning a τ's criterion c_i (*e.g.*, delivery time, cost, or quality). The social image of a partner is computed through the aggregation of the delegator's impressions regarding this partner. On the other hand, the partner's reputation is calculated by aggregating the impressions about its behavior produced by other agents (shared evaluations). Finally, by aggregating the partner's references (impressions), the delegator computes the partner's know-how, a measure that indicates the partner's competencies from the point of view of his previous delegators. A weighted mean of the impressions is employed to perform the impression aggregation. This approach groups

a set of impressions to form a single summary value, giving more relevance to impressions received most recently [15]. The mean of the aggregated values for an agent's social image, reputation, and know-how results in its competence measure.

Success rate ($S_{Rate}(\alpha, \beta, \tau) \in [0, 1]$) expresses how successful a partner β has been in performing a task (τ) delegated by α over time . The β's success relies on completing τ. Thus, when the β's success rate is 0 implies β has never completed τ. In contrast, the value 1 means β has completed τ every time it performed such a task. The success rate is computed as follows:

$$S_{Rate}(\alpha, \beta, \tau) = \begin{cases} \delta & \text{if } |Exe(\alpha, \beta, \tau)| = 0 \\ \frac{|Exe(\alpha,\beta,\tau)^+|}{|Exe(\alpha,\beta,\tau)|} * \left(1 - \left(\frac{1}{1+exp^{|Exe(\alpha,\beta,\tau)|}}\right)\right) & \text{otherwise} \end{cases} \tag{1}$$

where, $|Exe(\alpha, \beta, \tau)|$ is the number of times β performed τ, $Exe(\alpha, \beta, \tau)^+$ is the number of times that β successfully performed τ, and δ is the success rate default value used for the system initialization (*e.g.*, assigning 1 to δ results in an optimist initialization, because unknown partners are initially considered trustworthy). As a partner's success probability distribution is unknown, we introduce an accuracy factor in success rate computing. Such a factor represents the delegator's uncertainty concerning the behavior of a partner, which tends to decrease as the delegator interacts with it [2]. The competence measure and the success rate are combined into a trust measure through a linear scalarization function $TRUST(\alpha, \beta, \tau) = Comp(\alpha, \beta, \tau) * w_1 + S_{Rate}(\alpha, \beta, \tau) * w_2$, such as the sum of the weights $w_1 + w_2 = 1$ [18].

4 Experiments

In our evaluation, the partner selection is performed through the ϵ-*greedy* algorithm. The parameter ϵ took a value between 0.05 and 0.1 [16]. The partners are chosen based on their trust measures. The higher a partner's trustworthiness, the higher its chance of being selected as a delegatee. In particular, we run our experiments over a network of 43 agents connected through delegation chains. The network is organized like a 7x7 matrix. Each level has seven agents, except the first, with only the root agent, as presented in Figure 1. Note that this network is able to represent the agents' relationships through a disjunctive dependence form, where the symbol ($\prec$) denotes the dependence of an agent regarding another [13], and the symbols ($\wedge$) and ($\vee$) denote the social AND- and OR-dependencies, respectively [13]. In an AND-dependence, the delegator pulls more than one arm at once, selecting the product of delegatees with the highest trust mean. In this case, the delegator can only complete its task if all its delegatees complete their tasks first. In an OR-dependence, the delegator pulls the arm with the highest likelihood of success, selecting the most trustworthy partner as its delegatee.

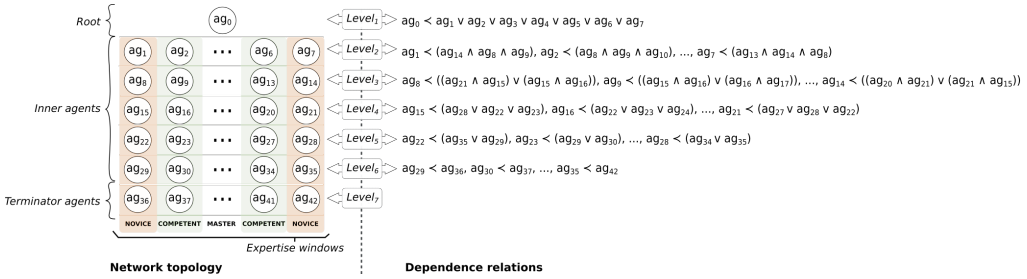

Fig. 1: Agents organization and their dependence relations.

The task delegation process starts with the root delegating a task to one of its partners, which may be decomposed or recursively delegated along the network. Such as the root, the inner agents can perform recursive delegation and task decomposition, sub-delegating and executing tasks, while the terminator agents can only execute tasks. Furthermore, the agents have distinct expertise degrees (*i.e.*, *master*, *competent*, and *novice*). The expertise degree affects the agent's failure likelihood. Failure prevents the agent from completing the task delegated to it. A master agent has no chance of failure, a competent agent has a 50% chance of failure, and a novice agent has an 80% chance of failure. When an agent in a lower level of a delegation chain fails, such a failure needs to be propagated to the higher levels of the chain. A failure propagation might prevent several other agents from completing their tasks, resulting in a chain of failures [5]. In our experiments, the agents penalize the failure of their delegatees through the impressions. A full penalization applied to an impression means assigning 0 to the scores of each task's criterion. Partial penalization depends on the delegatee's position in the failure chain ($F_{pos} \in [1, +\infty]$), where 1 is the agent's position that caused the failure that will be propagated. Considering an impression $Imp = \langle \alpha, \beta, r, \tau, t, S \rangle$, the score assigned by α to β, in case of a failure of β, for a criterion c ($Imp(S[c])$) is defined as follows:

$$Imp(S[c]) = (1 - (\frac{1}{F_{pos}(\beta)})) * pd \tag{2}$$

where, pd is the penalization discount factor, affecting the maximum score an agent can receive in case of failure.

As presented in Fig 2, we simulated the delegation task process 20 times, using as input the network shown in Figure 1 and varying the penalization discount factor from 0 to 1. Each run comprised 1600 trials, and the experiment performance was measured based on the agents' success rate and regret[1] average. Aiming to simulate a dynamic environment, at iterations 400, 800, and 1200, the agents have their expertise degree changed through a circular right shift of the expertise windows, indicated in Figure 1. A mono-episodic situation is obtained by assigning 0 to the penalization discount. Thus, for each delegation instance

[1] The loss in reward due to selecting a non-optimal action on every time-step

along a delegation chain, the delegator sees its delegatees as terminator agents, applying a full penalization in case of failure. In the partial penalization case, as the penalization discount value increases, a lesser penalization is applied to agents who fail, mainly for those at the higher levels of the delegation chains. For discount factors below 0.6, partial penalization tends to benefit the partner selection, increasing the chances of an agent being selected as a delegatee even if it indirectly committed a failure (*i.e.*, its position in the failure chain is greater than 1). However, when the penalization factor gets closer to 1 (above 0.6), the penalizations applied to agents cannot express their failures correctly since they might be seen as good evaluations. This prevents the delegator from making accurate decisions about which partner to select. In this case, from a delegator's perspective, all partners exhibit good social behavior, even those that make several mistakes.

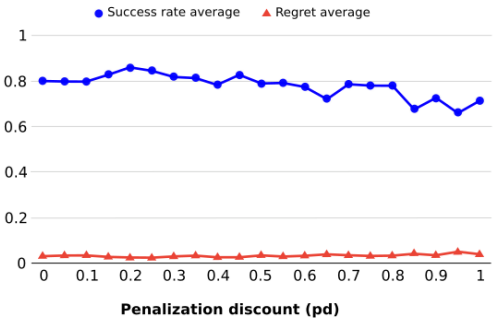

Fig. 2: Agents' success rate and regret average, varying (pd) from 0 to 1.

5 Conclusions

In this work, we demonstrated that explicitly coping with the delegation chains might be advantageous compared to the mono-episodic delegation task. Several features associated with the delegation chains can be explored to refine the agent's partner selection, such as the failure propagation and the penalization discount factor. Such an approach prevents an agent in the highest level of a chain from being overly penalized and consequently no longer selected as a delegatee due to a failure committed by another agent at a lower position of the chain. In future work, we intend to investigate other elements present in the delegation chains that can be employed in the agents' decision process involving task delegation, such as the different types of social relations established by the agents as they delegate tasks to each other [13], besides other penalization strategies to deal with failure propagation [5]. Additionally, we intend to investigate other network topologies since the topology affects the agents' connections and how they establish their relationships.

References

1. Afanador, J., Oren, N., Baptista, M.S.: A coalitional algorithm for recursive delegation. In: International Conference on Principles and Practice of Multi-Agent Systems. pp. 405–422. Springer (2019)
2. Ashtiani, M., Azgomi, M.A.: Contextuality, incompatibility and biased inference in a quantum-like formulation of computational trust. Advances in Complex Systems **17**(05), 1450020 (2014)
3. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Machine learning **47**(2), 235–256 (2002)
4. Baqueta, J.J., Morveli-Espinoza, M.M.M., Lugo, G.A.G., Tacla, C.A.: An adaptive trust model based on fuzzy logic. Revista de Informática Teórica e Aplicada **29**(1), 54–67 (2022)
5. Burnett, C., Oren, N.: Sub-delegation and trust. In: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 3. pp. 1359–1360 (2012)
6. Castelfranchi, C., Falcone, R.: Trust theory: A socio-cognitive and computational model, vol. 18. John Wiley & Sons (2010)
7. Cho, J.H., Chan, K., Adali, S.: A survey on trust modeling. ACM Computing Surveys (CSUR) **48**(2), 1–40 (2015)
8. Drugan, M.M., Nowe, A.: Designing multi-objective multi-armed bandits algorithms: A study. In: The 2013 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2013)
9. Griffiths, N.: Task delegation using experience-based multi-dimensional trust. In: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems. pp. 489–496 (2005)
10. Huynh, T.D., Jennings, N.R., Shadbolt, N.: Fire: An integrated trust and reputation model for open multi-agent systems. In: In Proceedings of the 16th European Conference on Artificial Intelligence (ECAI). pp. 18–22 (2004)
11. Karimadini, M., Lin, H.: Synchronized task decomposition for two cooperative agents. In: 2010 IEEE Conference on Robotics, Automation and Mechatronics. pp. 368–373. IEEE (2010)
12. Pinyol, I., Sabater-Mir, J.: Computational trust and reputation models for open multi-agent systems: a review. Artificial Intelligence Review **40**(1), 1–25 (2013)
13. da Rocha Costa, A.C., Dimuro, G.P.: Quantifying degrees of dependence in social dependence relations. In: International Workshop on Multi-Agent Systems and Agent-Based Simulation. pp. 172–187. Springer (2006)
14. Sabater, J., Paolucci, M., Conte, R.: Repage: Reputation and image among limited autonomous partners. Journal of artificial societies and social simulation **9**(2) (2006)
15. Sabater, J., Sierra, C.: Regret: reputation in gregarious societies. In: Proceedings of the fifth international conference on Autonomous agents. pp. 194–195 (2001)
16. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press (2018)
17. Turgay, E., Oner, D., Tekin, C.: Multi-objective contextual bandit problem with similarity information. In: International Conference on Artificial Intelligence and Statistics. pp. 1673–1681. PMLR (2018)
18. Van Moffaert, K., Drugan, M.M., Nowé, A.: Scalarized multi-objective reinforcement learning: Novel design techniques. In: 2013 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL). pp. 191–199. IEEE (2013)

2.3 Rethinking Comfort Profiles in Adaptive Building Energy Management Systems

Rethinking Comfort Profiles in Adaptive Building Energy Management Systems*

Jennifer Williams, Elnaz Shafipour, Alvin Shi, and Sebastian Stein

School of Electronics and Computer Science
University of Southampton, Southampton, England, UK
{j.williams, e.shafipour, gs1n17, s.stein}@soton.ac.uk

Abstract. As standard building occupancy schedules continue to change from static closed-door offices to dynamic open office layouts, we face new challenges for developing smart building energy management systems (BEMS) that can simultaneously *adapt* to save energy costs, while also incorporating the *comfort preferences* of the occupants. This is especially true for certain building types which by design are open layout, or partially-open layout such as schools, hospitals, and libraries. In this paper, we identify and explain three of the most critical challenges that specifically relate to incorporating feedback from building occupants into an interactive reinforcement learning algorithm. For each challenge, we propose how the challenge could be dealt with practically, within the context of our ongoing work and experimentation in this area. Overcoming these challenges opens new opportunities for artificial intelligence solutions that will place citizens in the centre and also help smart building designers move toward net-zero goals.

Keywords: building energy management · comfort profiles · reinforcement learning.

1 Introduction

Generally speaking, building energy management systems (BEMS) include both hardware components and software algorithms that control indoor climate such as heating, ventilation, and air conditioning (HVAC), indoor air temperature, indoor air quality, humidity, lighting, and certain sanitation equipment. All of these comfort-oriented components consume electricity and contribute to the overall cost for facilities to operate. Not only does energy for building operation take up nearly one third of energy consumption in the world [1], it is also estimated that the global market for BEMS and solutions based on artificial intelligence (AI) will reach (USD) $7.3 billion by 2026 [2]. However, most BEMS that are currently deployed in the real world do not have the capability to respond

* This work was supported by the UKRI Turing AI Acceleration Fellowship on Citizen-Centric AI Systems (EP/V022067/1) and by the Southampton Low Carbon Comfort Centre.

to changing occupancy schedules[1]. Most of the researched solutions to energy management involve a machine learning technique called reinforcement learning (RL [3] but these algorithms take a limited or naive view of both occupancy and comfort, such as treating occupancy as a binary problem (occupied/unoccupied) and comfort as a set room temperature for an entire building. These naive assumptions create a gap between simulation and real-world deployment.

AI researchers, engineers, and companies who work in this area envision smart buildings of the future that are adaptive, meaning that they account for natural shifts in occupancy levels and comfort needs throughout the course of the day. Further, there have been more recent efforts to include occupant *comfort preferences* into the software algorithms that manage building resources [4], rather than relying on a set temperature for an entire building or floor. Most of the energy consumed by a building is related to maintaining thermal comfort [5]. However, if a room, zone, or floor of a building has low occupancy, this can result in wasted energy and higher costs for building operators. From an industry perspective, goals for companies may include employee health, productivity, and safety, rather than just the energy saving and thermal comfort. Balancing these complex goals is very challenging to balance these goals.

Recent AI trends have studied BEMS in terms of reinforcement learning (RL) algorithms [6–9]. RL algorithms provide a means to optimize single or multiple objectives. In the work that we propose for balancing comfort preferences and energy consumption, we find that multi-objective reinforcement learning (MORL) provides the best opportunity for learning policies that optimize multiple objectives simultaneously [10]. Further, this type of problem also benefits from interactive reinforcement learning (IRL) [11] in order to incorporate feedback from building occupants when they are able to provide it. As we describe in this paper (Section 5), designing a MORL algorithm that performs meaningfully and is easy to train is a very challenging task. At the same time, previous research typically presents different models for different buildings, each with unique characteristics. Developing a one-size-fits-all solution is still not yet possible and is outside of the scope of this paper.

Designing an RL algorithm paradigm from which to work is only one step toward solving the technical challenges surrounding the problem of optimizing comfort preferences and energy consumption. In the case of comfort preferences, while there has been prior work relating to collecting [12], learning [4] and also aggregating [13] multiple inputs from people, this is far from a solved problem. In fact, a recent study [14] has highlighted more deeply that current research on comfort preferences falls short, especially because there are multiple definitions of comfort and needs may differ depending on the zone and use-case for a building.

The purpose of this paper is to examine three main challenges to comfort profiles for BEMS based on very recent advances in the state of the art for reinforcement learning. This overview is important because they take into consideration a realistic and interactive use-case for a deployed BEMS, while also considering that most research and development takes place in a simulated envi-

[1] https://www.znealliance.org/acco-bems

roniment. Addressing these challenges is another step closer toward more efficient and trustworthy BEMS whose foremost purpose is to service citizen comfort and safety, while also working toward sustainability goals. We present the following three challenges along with suggested solutions based on ongoing work and ex-perimentation:

1. Challenge 1: What kind of information needs to be collected from occupants and how often? (Section 3)
2. Challenge 2: How should occupant comfort profiles be aggregated? (Section 4)
3. Challenge 3: How can comfort profiles be incorporated into a reinforcement learning algorithm? (Section 5)

2 Background and Related Work

Previous work from [15] has examined how human behaviour changes when people are provided with a smart thermostat in their home, which may also enable them to make better economical decisions about their energy consumption with respect to price. The authors analyzed users' preferred temperature set-points at various times of the day, using different costs for electricity price-points. They found that many users were willing to reduce their electricity consumption when prices were high, even if that meant that their home temperature changed from their set-point. As the authors note, one limitation was that they used room temperature as a proxy for user comfort. They further did not take into account that comfort preferences may change throughout the day, or have shifting priorities due to user-intrinsic factors (e.g. disability, age, or short-term illness).

Recent work has introduced Gnu-RL[2] [16] which first learns from historical to pre-train policy gradients in order to reduce overall training time of the RL algorithm. This pre-training step makes the algorithm "precocial" in that it is already mature before RL training, which significantly reduces overall training time by a factor of simulation *decades* [17]. However, the Gnu-RL algorithm makes some assumptions which contribute major limitations, and which have not yet been addressed in research. It assumes that building dynamics are locally linear and also the algorithm is also missing an interactive component. As [18] point out, the linear design means that it is difficult to adapt to real-world settings that are dynamic, such as a hospital building or a college campus. Without an interactive component, it is not possible to incorporate occupant feedback into the algorithm in real-time.

Handling occupant interaction is itself a difficult problem. For example, [19] treat interaction from building occupants as evidence that they are dissatisfied based on the assumption that if people were already content and comfortable, they are not likely to provide positive feedback about their comfort. Counting the number of times that occupants submit their feedback is not ideal for large buildings, wherein the temperatures may be different depending on the zones and

[2] https://github.com/INFERLab/Gnu-RL

occupancy levels. Their RL algorithm uses deep learning to balance and optimize both comfort and energy by attempting to simultaneously minimize discomfort and minimize energy consumption. However, their approach does not attempt to represent that comfort levels for some occupants may require a priority as with disabled, elderly, or children.

3 Challenge 1: Comfort Preference Collection

Our first challenge relates to how data is collected for modelling comfort preferences. In the first instance, it is necessary to have a large dataset to train an RL algorithm. However, since comfort preferences and profiles are not part of existing datasets, this poses a challenge for modelling interaction in the algorithm training. Another facet to this challenge is that it is unknown exactly what type of information must be collected. For example, in an office building where the occupants are recurring, it may be easiest to allow the occupants to fill in a *profile*, for example using an app or other web interface. The profile information can be stored and accessed by the BEMS and RL algorithm to ensure that their office and area are comfortable when occupied. For other types of buildings such as schools or hospitals, where occupancy is changing, the BEMS would benefit from regular feedback so that it can adapt, but not so much feedback that occupants will become annoyed by answering questions.

We propose to conduct an online survey where we provide participants with an imaginary scenario that describes "their office" and ask them to enter information about their preferred temperature and ventilation set-points at various times of day. This information would allow us to create realistic *comfort profiles* that can be incorporated into the design of our ideal BEMS RL algorithm. We can further ask participants to envision the circumstances where they would become annoyed by providing real-time feedback. While this type of survey does have its limitations, it may help with overcoming a lack of explicit comfort profiles in publicly available datasets that are used alongside highly-detailed building dynamics simulations like EnergyPlus[3].

4 Challenge 2: Comfort Profile Aggregation

When we discuss comfort profiles, a very important element of that conversation is how to manage multiple different comfort profiles at the same time, in a BEMS. This is a very challenging research area. In fact [20] present this as a problem of learning different control policies that will allow for human behaviour to change over time, and change differently for each occupant. There is currently no suitable algorithmic approach from reinforcement learning that can manage that level of ongoing uncertainty. Towards a solution, we propose that some occupants in the building may require a weighting for their comfort profile, which would allow their preferences to take priority over other occupants. This weighting would be

[3] https://energyplus.net/

ideally related to health and safety concerns, rather than unrelated factors such as authority or financial status. It is important to consider that this type of data collection involves collecting data about occupants' behaviour and preferences, which can raise privacy concerns in the case that data is not properly anonymized or some occupants are not comfortable sharing their information. Privacy is something that all researchers dealing will need to consider.

We can further consider aggregating comfort profiles in terms of zones. For example, in an open-office layout with hot-desks, it may be preferable to offer occupants that certain zones have particular properties and allow those occupants to choose their zone accordingly.

5 Challenge 3: Incorporating Comfort into Reinforcement Learning Algorithms

In our work, we assume that there is a single-level building with different zones (e.g. an office building). Each zone has a number of occupants who may enter and their zone, or the building itself. The goal is to set the temperature of each zone to maximize the satisfaction level of the occupants. Also, to be able to understand the satisfaction level of the occupants, we will interact with them and get their feedback. As described earlier, in simulation experiments, we may utilize realistic comfort settings gathered from a survey so that our simulation reflects preferences from real people rather than randomly contrived values.

In a standard RL problem, a learning agent observes the resulting environment transitions in a number of discrete steps and learns the control policy to maximize the accumulated reward. However, in this work, we will use Interactive RL to solve this problem, which will allow us to get feedback from people in the building and adjust the building's temperature based on that feedback and other observations.

In our model, we define each zone as a tuple $z(p, t)$ where p is the number of people and t is the current temperature of that zone. Then, we define the state space as a set of zones $S = \{z_1, z_2, ..., z_n\}$. Also, the action that we take at each iteration would be a list of temperatures $a = \{t_1, t_2, ..., t_n\}$.

In this work, we will get the following information from the user.

- Specifying occupants' preferred temperature.
- Ranking different actions/schedules for a particular day.
- Occupant's zone number
- Their priority level.

Using this information we can train our initial MORL as we will have multiple zones (as well as multiple objectives) and we should set the temperature for each. Later, using interactive RL, we will require users to evaluate the system's performance qualitatively. Our expected feedback from the user would be:

- Are they present in their zone?
- Did they change their zone?

– Their responses to the thermal environment (optimal temperature set point for the zone they are currently in)
– Giving binary (positive/negative) feedback and saying in general whether they are happy with the temperature or not.

6 Discussion and Future Work

We have highlighted the current state-of-the-art for AI-based approaches to optimizing building energy and occupant comfort. While there have been many recent advances, some challenges still remain. We introduced three such challenges and described our proposed approach for each, which we are continuing to explore in our work.

Our challenges have addressed that we may assign individual weights for particular comfort profiles, and aggregate them based on this information. This adds a layer of complexity and allows us to assign higher weight to occupants with special needs, such as people with a health condition (e.g. respiratory problems who may require better air quality), disability, children and the elderly. In our future work, we will develop a new objective measure that will help us determine if an occupant should have priority. We will also use a survey to get input from people about whether they think these are fair and equal decisions. Otherwise, there is a risk of conflicts or dissatisfaction among occupants who feel that their needs are not being met.

Our proposed model will require complimentary techniques of interactive RL (IRL) as well as multi-objective RL (MORL). Such a system will be very complex to design and train, which is why we are sharing the challenges of this work at the outset. We hope that other researchers in the community who work on BEMS will share our interest in re-thinking how to best handle occupant comforts alongside energy.

References

1. T. L. Vu, N. T. Le, Y. M. Jang, *et al.*, "An overview of internet of energy (IoE) based building energy management system," in *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 852–855, 2018.
2. I. Global Industry Analysts, "Global building energy management systems (BEMS) market to reach $7.3 billion by 2026," Mar 2022.
3. R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
4. Y. Xu, S. Chen, M. Javed, N. Li, and Z. Gan, "A multi-occupants' comfort-driven and energy-efficient control strategy of VAV system based on learned thermal comfort profiles," *Science and Technology for the Built Environment*, vol. 24, no. 10, pp. 1141–1149, 2018.
5. L. Yang, H. Yan, and J. C. Lam, "Thermal comfort and building energy consumption implications–a review," *Applied energy*, vol. 115, pp. 164–173, 2014.

6. E. Mocanu, D. C. Mocanu, P. H. Nguyen, A. Liotta, M. E. Webber, M. Gibescu, and J. G. Slootweg, "On-line building energy optimization using deep reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 3698–3708, 2018.
7. Y. Pan and L. Zhang, "Roles of artificial intelligence in construction engineering and management: A critical review and future trends," *Automation in Construction*, vol. 122, p. 103517, 2021.
8. F. Wang, L. Zhou, H. Ren, X. Liu, S. Talari, M. Shafie-khah, and J. P. Catalao, "Multi-objective optimization model of source–load–storage synergetic dispatch for a building energy management system based on TOU price demand response," *IEEE Transactions on Industry Applications*, vol. 54, no. 2, pp. 1017–1028, 2017.
9. Z. Wang and T. Hong, "Reinforcement learning for building controls: The opportunities and challenges," *Applied Energy*, vol. 269, p. 115036, 2020.
10. K. Van Moffaert and A. Nowé, "Multi-objective reinforcement learning using sets of pareto dominating policies," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3483–3512, 2014.
11. A. L. Thomaz, G. Hoffman, and C. Breazeal, "Real-time interactive reinforcement learning for robots," in *AAAI 2005 workshop on human comprehensible machine learning*, vol. 3, 2005.
12. F. Jazizadeh, A. Ghahramani, B. Becerik-Gerber, T. Kichkaylo, and M. Orosz, "Human-building interaction framework for personalized thermal comfort-driven systems in office buildings," *Journal of Computing in Civil Engineering*, vol. 28, no. 1, pp. 2–16, 2014.
13. K. Aduda, W. Zeiler, and G. Boxem, "Smart Grid-BEMS: the art of optimizing the connection between comfort demand and energy supply," in *2013 Fourth International Conference on Intelligent Systems Design and Engineering Applications*, pp. 565–569, 2013.
14. J. Williams, B. Lellouch, S. Stein, C. Vanderwel, and S. Gauthier, "Low-carbon comfort management for smart buildings," in *2022 IEEE International Smart Cities Conference (ISC2)*, pp. 1–5, 2022.
15. "Save money or feel cozy?: A field experiment evaluation of a smart thermostat that learns heating preferences," in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, vol. 16, International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), 2017.
16. B. Chen, Z. Cai, and M. Bergés, "Gnu-RL: A precocial reinforcement learning solution for building HVAC control using a differentiable MPC policy," in *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pp. 316–325, 2019.
17. Z. Zhang and K. P. Lam, "Practical implementation and evaluation of deep reinforcement learning control for a radiant heating system," in *Proceedings of the 5th Conference on Systems for Built Environments*, pp. 148–157, 2018.
18. Y. Lei, S. Zhan, E. Ono, Y. Peng, Z. Zhang, T. Hasama, and A. Chong, "A practical deep reinforcement learning framework for multivariate occupant-centric control in buildings," *Applied Energy*, vol. 324, p. 119742, 2022.
19. L. Scarcello and C. Mastroianni, "Cognitive systems for energy efficiency and thermal comfort in smart buildings," in *IoT Edge Solutions for Cognitive Buildings*, pp. 329–345, Springer, 2022.
20. H. Zhang, D. Wu, and B. Boulet, "A review of recent advances on reinforcement learning for smart home energy management," in *2020 IEEE Electric Power and Energy Conference (EPEC)*, pp. 1–6, 2020.

3 Citizen-Aware Autonomous Systems

3.1 Assimilating Human Feedback for Autonomous Vehicle Interaction in Reinforcement Learning Models

Assimilating Human Feedback from Autonomous Vehicle Interaction in Reinforcement Learning Models

Richard Fox[1] and Elliot A. Ludvig[1]

University of Warwick, Coventry CV4 7AL, UK

Abstract. This paper develops a new methodology to incorporate what behaviour pedestrians and other road users find helpful into computational reinforcement learning (RL) in the context of autonomous vehicles. Extending previous work in preference learning, the RL agents are given greater rewards for actions deemed superior by assessors. The work uses participant judgements rather than a preference over vehicle trajectories, attempting to shift the paradigm of policy evaluations to be more human-centric and less dependent on experts. Applying a parametric utility function to a shaped reward to represent the judgement feedback, we propose a form for this function that can be non-linear over a trajectory without breaking necessary assumptions for RL. When eliciting feedback, we propose asking for a judgement on a Likert scale that can be integrated into parameters for the utility function, directly converting judgements to encourage the desired behaviour.

1 Introduction

A primary concern of automation is its ability to integrate with existing systems, which, when considering road usage, have a strong social element [1]. On top of optimising aspects of driving that naturally lend themselves to reward metrics, i.e., time to destination, fuel efficiency and lane discipline [10], vehicle automation must learn valid and useful behaviours when interacting with pedestrians and other road users [3, 11]. The main challenge many researchers face is designing a suitable reward function to elicit such behaviours [9]. What are these behaviours, and how does one translate that to an RL algorithm?

This paper develops a methodology to directly ask what behaviour pedestrians find suitable by collecting quantitative data that can be used to measure an algorithm's performance. This work attempts to show that improvements in "human-like" behaviour can be gained when such systematic feedback is collected from human agents and embedded into the reward function. We formalise an iterative feedback loop using computational RL and behavioural science techniques, where the reward structure is adapted by eliciting behavioural judgements collected from people in a controlled environment. The policy is updated by training with the updated reward and then shown, without explicit participant knowledge, back to the participant to assess the updated behaviour. The

reward function is based on three terms gleaned from pilot experiment data [4] in which participants were asked to judge each aspect of the vehicle's behaviour, and the corresponding term in the reward function is altered.

Much of the existing literature on embedding human judgments into autonomous RL agents focuses on the efficiency of an algorithm to minimise utilitarian measures such as time to collision, time to destination and fuel efficiency. For example, Knox et al. [9] provide an overview of shaped reward functions used in work on autonomous vehicle optimisation, defining several categories of reward, one of which is human preferences. The system being analysed often has many complex inputs [8, 12], as an autonomous vehicle would have. Still, an equally complex social environment exists underneath [3, 11], which we can only access through the medium of human feedback.

A second related approach is preference learning, which allows learning an expert's ranking over a set of actions. The basic approach is to elicit feedback to learn a policy, using a pair-wise trajectory preference as a justification of policy quality, but this often requires experts in narrow and highly skilled domains. To alleviate the limitation of pair-wise preferences to inform on policy quality, there has been work [2] that allows for the agent to use the expert's rankings to extrapolate new behaviours. By seeking explanatory patterns for the preferences, the best policy even exceeded the best performance that was demonstrated. In other work, Jain et al. [5] allowed their model to learn context-dependent behaviours much more efficiently, using a reward-based utility function without an explicit cost function to be optimised. Instead, their model used the preference feedback to compute a gradient of the utility parameters directly, permitting the policy to be defined by the RL agent. Creating such a utility function can be especially useful for the current project because feedback from pedestrians is not deterministic nor even necessarily stationary.

When directly considering how pedestrians interact with AVs, Jayaraman et al. showed that pedestrians' acceptance of AVs depends on their trust in the AVs [7]. The conducted an objective-based evaluation of behaviours [6], in terms of safety, performance, and comfort with 30 human participants in a virtual-reality environment. They found that pedestrians' trust in AVs was influenced by AV driving behaviour as well as the presence of a signal light. In [6], the authors propose a model representing the three objectives–safety, efficiency, and comfort– by the weights of a linear regression of observable variables, including latent trust variables and other experimental considerations. The intuition behind the objectives is that driving behaviours can be characterised by weighting values, especially as there is a competing optimal weighting between AV passengers' preference and the pedestrian's preference, both of which should be optimised.

In the present work, we elicit participant judgements, which are a subjective decision on vehicle driving quality, rather than eliciting a preference between multiple given vehicle trajectories. Participants thus are not considered experts; we regard them as normal pedestrians and are guiding the reward function by their judgements. Driving well in the eyes of other road users is a fundamental

aspect of the task of driving, and the current approach will allow us to embed such judgments into the policy of AVs.

2 Methodology

We are interested in finding the best set of actions A given the states S visited, referred to as the optimal policy $\pi^*(s)$, where policy function π returns an action $a \in A$ to be taken in the state $s \in S$ of the environment of interest. A trajectory $\tau^\pi = \{(s,a)_t\}\,\forall t \in \{0,1,2,\cdots,T\}$ is the set of state-action pairs returned by a policy for a given terminated instance of environment interaction over time T. We can compute the total expected return from a trajectory as:

$$R_{\tau^\pi} = \sum_{t=0}^{T} r(s_t) \quad \forall s_t \in \tau^\pi. \tag{1}$$

The problem is formulated with Reinforcement Learning in mind, in particular methods that utilise Q-learning. As such, agents are trained with Deep Q-Network (DQN) models as we wish to use the simplest methods to control complications around inference of what is altered by changing the reward function and what participants are giving judgements on. Given that the state space of our custom Pygame simulator [1] is large, based on pixel coordinates, we still require a deep approach as the simpler tabular methods are computationally intractable.

For this work, we consider an example shaped reward decomposed into reward for relevant behaviours that has three terms:

$$r(s_t) = \frac{\nu_t}{\nu_{max}} + \frac{(1-\delta_t)}{\delta_{max}} + \frac{(1-\Delta_t)}{2\pi} - 3, \tag{2}$$

where ν_t is the speed the agent vehicle entered state s_t at and ν_{max} is the maximum speed of the vehicle, δ_t is the current distance from the agent position the goal, δ_{max} is the furthest legal position from the goal possible, and Δ_t is the change in heading vector (amount of steering) in radians. This shaping of rewards for speed, position and steering smoothness independently allows us to have different utility parameters for each term. Each of the three terms are in the interval (0,1] with 1 being the optimal value, and we apply a negative 3 to our reward to ensure that the reward is always negative, making an expected reward of 0 the theoretically optimal solution.

In training the agent, an episode is terminated if the vehicle collides with a pedestrian, and that state is awarded $-\infty$ as a reward. Upon successfully completing the task, the agents are rewarded with a positive amount that equals the absolute value of the minimal reward they can receive and still reach the goal. Having a negative reward and a task that ends with success or failure encourages the agents to find the most efficient routes to end the episode as early as possible, thus minimizing the accumulation of negative rewards.

[1] https://github.com/Rik-Fox/pygame_ped_env

The following parametric utility function U_θ will be applied to the intrinsic reward 2 to represent the judgement feedback (see below for more on elicitation procedure):

$$U_\theta(R_{\tau^\pi}) = \sum_{t=0}^{T} \left(\frac{\nu_t}{\nu_{max}} - 1 \right)^{\theta_\nu} + \left(\frac{(1-\delta_t)}{\delta_{max}} - 1 \right)^{\theta_\delta} + \left(\frac{(1-\Delta_t)}{2\pi} - 1 \right)^{\theta_\Delta}. \quad (3)$$

The advantage of parameterising each term is that it enables feedback to be elicited for each term individually, allowing each term to be non-linear over a trajectory, but decompose into a linear sum at a policy execution level. As we do not want to yield a complex valued reward, the negative domain shifting is also decomposed and applied to each term individually, importantly after applying the parametrised exponent. At policy execution, we can still return a sum of 3 scalar terms only depending on the previous state, therefore not violating either of the assumptions of Markovian and sub-game optimal dynamics for learning, even if the utility function itself does not satisfy them.

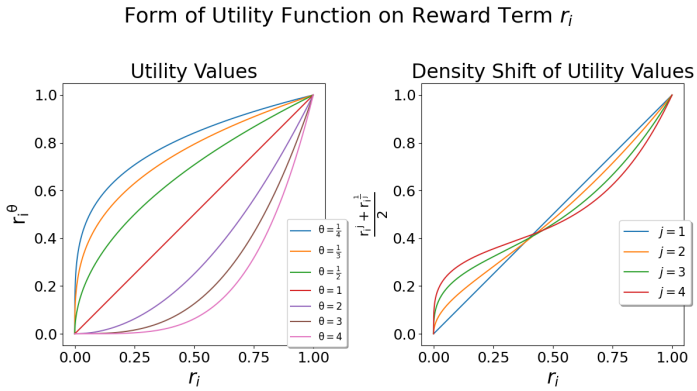

Fig. 1. The left panel shows the effect of applying the utility to a single term r_i , where low values of exponent θ_i for any given term dampen high rewards and high values do the converse . The right panel shows the difference of inversely valued function exponents compared to the linear case. The graph shows how the dampening effect of a concave utility is greater on rewards above 0.5 than the convex counterpart, thus undermining the current policy and encouraging adaptation. The converse is also true, indicating that for rewards below 0.5 the increase from the convex utility function outweighs concave dampening, thus reinforcing the current behaviour in all circumstances.

This utility function acts as the reward and can be optimised directly but statically, only learning a policy for one fixed set of parameters. When the utility is higher than the base linear reward, rewards increase across all states with the effect compounding as the linear reward increases, effectively reinforcing the current policy's behaviours. The converse is true for low ratings, which encourage exploration of different behaviours as the policy reconverges. The linear form of the utility, equivalent to the standard shaped reward, is used as a pre-trained model from which to train each utility variant model for a further fixed number of episodes.

3 Experimental Procedure

Human participants would be shown these trajectories and provide feedback using an interactive experiment where they control a pedestrian in a simulated environment Fig. 3 and experience the agent trained with a given reward function. The subsequent models are then shown based on the feedback, collecting the participant trajectories as the feedback is collected.

When eliciting feedback, we ask human participants to make a judgement about a specific interactive trajectory on a 7-point Likert scale that is symmetric around 0. We then ask for a judgement on a new interactive trajectory that uses the utility that corresponds with the participant's judgment Eq. 3. Each term in the reward function has been designed such that it is drawn from the interval $(0, 1]$ and linear with progression, i.e. driving at half speed gives a reward of 0.5. This allows a mapping from the integer Likert scale points (LP) to the parameterised exponents in Eq 3, directly, as

$$\theta_i = \begin{cases} sgn(LP) \cdot 2, & if\ |LP| \geq 1.8 \\ sgn(LP) \cdot 1, & if\ |LP| \geq 0.6 \\ 0, & Otherwise \end{cases}, \tag{4}$$

which results in a symmetric function that is convex for positive LP values, concave for negative LP values 1, and linear at $LP = 0$ where it recovers the intrinsic reward function Eq. 2. Eq. 4 relates higher ratings to higher utility when using a utility of a similar form to U_θ and yields 125 (or 5^3) parameter sets when implementing a three-term reward/utility function and thea seven-point Likert scale $-3to3$, mapped down to 5 points $-2to2$, is used for the human judgments.

The participant thus effectively picks the next model for them to interact with , and from that selection data, we can look to find the utility that best represents what they judge to be a better-performing agent. While this data will vary significantly between individuals, on aggregate, any trends will serve as justification for the selection of parameters. Models for all possible parameter sets are trained a priori, which allows for seeming real-time adjustment of the agents being shown to the participants, based on their judgments. Participants in the behavioural experiments will be shown the policy corresponding to a relative change in the parameters based on their judgement LP values. The participants are not explicitly told about this and are simply judging the behaviour of "different" algorithms.

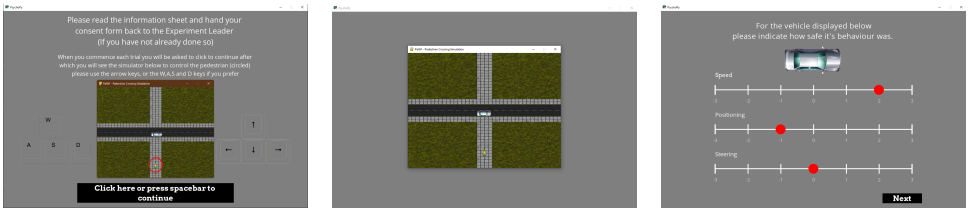

Fig. 2. Stages of the behavioural experiment procedure, starting from left to right with an explanation of the task shown only once, then a simulation where the interaction happens, followed by Likert scale responses repeated for the duration of data collection

4 Discussion

This approach lays out an experimental design for cleanly embedding human judgments into the reward function of an AV. Optimising the utility of a policy based on feedback as we saw in [5], but in an out-of-the-loop paradigm. By taking the utility into this paradigm, each piece of feedback used contributes to finding the best behaviour without prescribing what that behaviour should look like. Lay people may not be able state which exact behaviours they would judge as better-performing than what they have already encountered. By decomposing the judgements and reward structure, however, we aim to minimise this gap and be able to identify and target areas of poor behaviour performance by the agent. This approach allows the extrapolation of better policies like [2], but with a greater possibility for the RL agent to be able to innovate its own strategies.

While it is theoretically possible to embed any judgements using this approach, non-passive judgements are recommended, where the participants are in control of an entity in the environment and judging based on their interactions with the agent, instead of simply observing trajectory. Participant control leads to each instance being unique and showing much more of the policies' learned behaviours, even with the same or similar utility parameters than the observation of a fixed representative trajectory. These interactions can give valuable insight into latent sub-groups in the participants, allowing for analysis and classification of their behaviour and observing trends dependent upon this.

These repeated interactions, sub-group classifications, and aggregate statistics go part of the way to remedying the inherent fallibility of human feedback. Human judgements (and indeed expressed preferences) may not align with their true interests or quality criteria. The current method incorporates the subjective feedback in a direct numerical way and can optimise from feedback in an out-of-the-loop manner across multiple iterations. As a result, the method finds the reward shape that matches judgements/preferences the closest, not the judgements/preferences themselves, this allows for effects such as sample bias to propagate. Testing the value of interacting with the AV and its effects on the subjects responses would add to our understanding of the feedback interactions, possibly by running a comparative study with some no-interaction control groups.

The current method only elicits feedback on full trajectories of converged policies, which does not push temporal credit assignment onto judgement makers, as opposed to [5] which asks for time contextualised feedback, instead of transforming the already-shaped reward. An out-of-the-loop approach with iterative trials across many individuals helps to give an overall picture of which behaviours are judged more favourably, thereby allowing for adaptive behaviours to better target a particular individual. This approach does push the distinguishing between low-performance, good-behaviour policies from high-performance, bad-behaviour policies onto judgement makers, even allowing for non-human-like behaviour that is favourably judged. This approach selects for higher performing behaviours that assimilates better into the social climate. Therefore, judgements are not an alternative reward signal but the reward signal that fully represents the underlying problem and is the task we want to achieve success in.

5 Conclusion

This work attempts to shift the paradigm of policy evaluations to be more human-centric. It leaves the efficient processing of observation data that accurately represents the vehicle's environmental state to other work. Instead, the work focuses on including the subjective human stakeholder's opinions to improve automated driving performance in the social aspect of road use. This focus helps to address the simulation-to-real gap felt by subjective stakeholders in the real world, embedding automation into the social climate with minimal friction. Human demonstrations, preferences, or judgements are not only worthwhile to improve agent behaviour in complex environments, but also to assist in integrating automation into our lives.

References

1. Bonnefon, J.F., Shariff, A., Rahwan, I.: The social dilemma of autonomous vehicles. Science **352**(6293), 1573–1576 (2016)
2. Brown, D., Goo, W., Nagarajan, P., Niekum, S.: Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In: International conference on machine learning. pp. 783–792. PMLR (2019)
3. Chater, N., Misyak, J., Watson, D., Griffiths, N., Mouzakitis, A.: Negotiating the traffic: Can cognitive science help make autonomous vehicles a reality? Trends in cognitive sciences **22**(2), 93–95 (2018)
4. Fox, R., Ludvig, E.A.: Using human behaviour to guide reward functions for autonomous vehicles. In: 5th Multidisciplinary Conference on Reinforcement Learning and Decision Making. RLDM. No. 2.141
5. Jain, A., Sharma, S., Joachims, T., Saxena, A.: Learning preferences for manipulation tasks from online coactive feedback. The International Journal of Robotics Research **34**(10), 1296–1313 (2015)
6. Jayaraman, S., Robert, L., Yang, X.J., Tilbury, D., et al.: Automated vehicle behavior design for pedestrian interactions at unsignalized crosswalks (2021)
7. Jayaraman, S.K., Creech, C., Tilbury, D.M., Yang, X.J., Pradhan, A.K., Tsui, K.M., Robert Jr, L.P.: Pedestrian trust in automated vehicles: Role of traffic signal and av driving behavior. Frontiers in Robotics and AI **6**, 117 (2019)
8. Kiran, B.R., Sobh, I., Talpaert, V., Mannion, P., Al Sallab, A.A., Yogamani, S., Pérez, P.: Deep reinforcement learning for autonomous driving: A survey. IEEE Transactions on Intelligent Transportation Systems **23**(6), 4909–4926 (2021)
9. Knox, W.B., Allievi, A., Banzhaf, H., Schmitt, F., Stone, P.: Reward (mis) design for autonomous driving. Artificial Intelligence **316**, 103829 (2023)
10. Pal, A., Philion, J., Liao, Y.H., Fidler, S.: Emergent road rules in multi-agent driving environments. In: International Conference on Learning Representations
11. Ritchie, O.T., Watson, D.G., Griffiths, N., Misyak, J., Chater, N., Xu, Z., Mouzakitis, A.: How should autonomous vehicles overtake other drivers? Transportation research part F: traffic psychology and behaviour **66**, 406–418 (2019)
12. Zhou, M., Luo, J., Villella, J., Yang, Y., Rusu, D., Miao, J., et al.: Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving. arXiv preprint arXiv:2010.09776 (2020)

3.2 Explainable Agents Adapt to Human Behaviour

Explainable agents adapt to Human behaviour

Adrian Tormos[1][0000−0003−1658−9393] and Victor Gimenez-Abalos[1][0000−0003−4514−6145] and Marc Domènech i Vila[2] and Dmitry Gnatyshak[1][0000−0001−6779−6283] and Sergio Alvarez-Napagao[1 2][0000−0001−9946−9703] and Javier Vázquez-Salceda[2][0000−0003−1732−9446]

[1] Barcelona Supercomputer Center (BSC) Pl. Eusebi Güell 1-3 - 08034 Barcelona
{adrian.tormos,victor.gimenez,dmitry.gnatyshak,sergio.alvarez}@bsc.es
[2] Universitat Politècnica de Catalunya-BarcelonaTECH, Edifici Omega
C/ Jordi Girona 1-3, E - 08034 Barcelona
jvazquez@cs.upc.edu

Abstract. When integrating artificial agents into physical or digital environments that are shared with humans, agents are often equipped with opaque Machine Learning methods to enable adapting their behaviour to dynamic human needs and environment. This brings about agents that are also opaque and therefore hard to explain. In previous work, we show that we can reduce an opaque agent into an explainable Policy Graph (PG) which works accurately in multi-agent environments. Policy Graphs are based on a discretisation of the world into propositional logic to identify states, and the choice of which discretiser to apply is key to the performance of the reduced agent. In this work, we explore this further by 1) reducing a single agent into an explainable PG, and 2) enforcing collaboration between this agent and an agent trained from human behaviour. The human agent is computed by using GAIL from a series of human-played episodes, and kept unchanged. We show that an opaque agent created and trained to collaborate with the human agent can be reduced to an explainable, non-opaque PG, so long as predicates regarding collaboration are included in the state representation, by showing the difference in reward between the agent and its PG. Code is available at [3]

Keywords: Interactive Reinforcement Learning · Explainable AI · Cooperative AI · Multi-Agent Reinforcement Learning

1 Introduction and Related Work

The advances of the artificial intelligence field offer optimistic prospects of integrating self-interested artificial agents–both software and embodied–into complex socio-technical systems where human and artificial agents collaborate in

[3] https://github.com/HPAI-BSC/explainable-agents-with-humans

a physical environment, a digital one or hybrid physical/digital environments. Oftenly the agents are equipped with one or several Machine Learning methods and algorithms in order to provide such agents with the capability to adapt to human needs and relevant environment changes. This means that these agents' behaviours may evolve through their lifetime. How to ensure that the learnt behaviour stays aligned with some desired social values like fairness or equality becomes a critical issue to ensure these agents are trustworthy. We need methods to understand and reason over the agents' behaviour and to hold accountability [8].

In those scenarios where agents' behaviours can be evaluated by well defined task performance metrics, reinforcement learning approaches are able to reach remarkable results. However, the outcome of most well-performing models can only be interpreted as a opaque agent which, given its perception of the world and optionally some internal states, outputs the action to perform in the environment. These opaque models are hard to inspect or to explain [6]. So this brings two questions. How can we be sure that in real-world tasks the performance is unambiguous and fair? And moreover, even if the performance evaluation was perfect, how can we ensure that the behaviour learnt by the agent is fair? A way to do so would be translating the behaviour of an agent into a transparent explainable policy [4].

Previous work shows that an opaque agent can often be reduced into an explainable Policy Graph [4,2,11] by sampling its trajectories. A Policy Graph (PG) is a directed graph representation $G = (V, E)$ of an agent's behaviour that maps discretised visited states to nodes and the agent's actions to edges: each $v \in V$ is a discrete state, and each $e \in E$ represents a state transition (s, a, s') after taking a certain action a, that also records probability $P(s', a|s)$ [7]. When the transformation between the world state and the node is expressive enough, a policy graph can approximate the behaviour of an opaque agent without a remarkable loss in performance. This system can be directly applied to both single- and multi-agent environments and tasks.

The choice of the state representation is key to the performance of the PG agent for all cases. Especially so, in multi-agent environments which require collaboration, adding predicates regarding what the other agent is doing is key in avoiding catastrophic failure [11]. However, something that still remains to be seen is how this predicate relevance extrapolates to agents that have to collaborate with human or human-like counterparts, and how easily their behaviour can be reduced into explainable PGs.

The contribution of this work is to explore this issue further by studying settings in which an agent trained from human behaviour collaborates with our experimental opaque agent. We use a virtual kitchen environment, Overcooked-AI [1], to simulate this human-AI collaboration. The human agent is computed by using Generative Adversarial Imitation Learning (GAIL) onto series of human-played episodes [1], effectively imitating human behaviour [13,9,12], and kept invariant. We show that an opaque agent created and trained to collaborate with this human-like agent can be reduced to an explainable, non-opaque PG,

Table 1. Variables used to describe the domain by each discretiser. Each variable may take only one value in a state. `held` and `held_partner` represent the object the agents are holding, where `O,T,D,S` stand for the items that can be held (onion, tomato, dish, soup). `item_pos` shows the next action to get to a certain item (be it an item source or not), where `U,D,L,R,I,S` for the actions to reach an item (go up, down, left, right, interact or stay). `partner_zone` refers to the cardinal direction (`N,NE...`) in which the other agent is located *w.r.t.* to the PG agent. Note that `N,W,S,E` are only used when the two agents are in the same horizontal or vertical axis.

	Variables (domain)

D1	`held(O, T, D, S, ∅)`
	`pot_state(Empty, Waiting, Cooking, Finished)`
	`item_pos(U, D, L, R, I, S)`, $\forall item \in \{$`O,T,D,Pot,service`$\}$

D2	$D1 \cup \{$`held_partner(O, T, D, S, ∅)`$\}$
D3	$D1 \cup \{$`partner_zone(N, NE, E, SE, S, SW, W, NW)`$\}$
D4	$D2 \cup D3$

so long as predicates regarding collaboration are included in the state representation, by showing the relative difference in reward between the agent and its PG. Contrary to the findings of previous work, we show that in environments in which collaboration is not compulsory and simply beneficial, and given that the human agent does not adapt to agent behaviour, the PG agent does not rely on state representations that involve knowledge about the behaviour of the other agent.

2 Methodology

In this study, we propose a comparison in performance between different kinds of agents when interacting with a human-like agent. In order to do so, the human behaviour is simulated by reusing the agents trained from human replays in Carroll et al. [1] using GAIL [5]. The human trajectories employed were produced by humans playing with other humans in 5 different Overcooked layouts, which is a cooperative game between two players.

In our experiments, for each layout, one opaque agent is trained alongside the corresponding human agent using Proximal Policy Optimization (PPO) [10], following the methodology of [1], and then it is converted into a PG. Compared

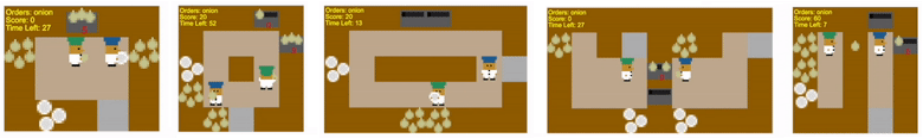

Fig. 1. Overcooked layouts. From left to right, *simple*, *random1*, *random3*, *unident_s* and *random0*. The opaque agent always takes the role of the blue agent.

to previous work by Domenech et al. [11], in this work one of the agents is never trained, as we want the human agent to purely represent a human. This is expected to result in lower rewards, given that one of the agents cannot adapt and exploit the behaviour of the other.

Each policy graph is built on top of 1500 trajectories from each trained opaque agent while playing alongside the human agent in each layout. Since Policy Graphs are dependent on the representation of the state, a set of four different discretisers are used – the ones introduced in Domenech et al. [11] (described also in Table 1). Notably, interest should be put on the fact that each discretiser is increasingly expressive, mostly in regards of what the other agent is doing, allowing for more seamlessly collaboration.

When sampling actions from a policy graph, one can do so in a *greedy* (pick the most probable action given a state: $a = argmax_a\, P(a|s)$) or *stochastic* manner (sample the action from the distribution: $a \sim P(a|s)$). Both methods are experimented with, and the rewards obtained are compared against the rewards for the original opaque agent.

3 Experiments

The opaque and the PG agents are evaluated on 5 different Overcooked layouts (Figure 1), which can be ordered from less to more explicitly collaborative. The first two, *simple* and *random1*, do not require collaboration to achieve high rewards, although one agent can hinder the other by obstructing its path. In *random3*, agents could benefit from collaborating by covering different areas of the layout. In *unident_s*, in which agents are separated, splitting tasks is also very favourable; and lastly *random0* splits the resources between agents, thus needing collaboration to get any reward at all.

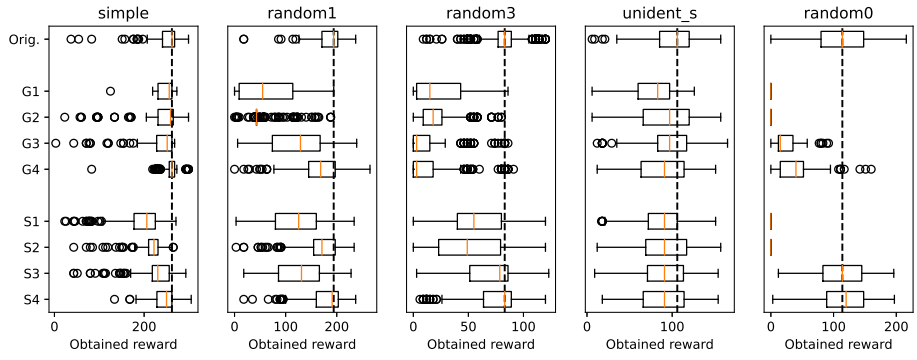

Fig. 2. Distribution of obtained rewards in all layouts. *Orig.* refers to the original opaque agent. *G1-G4* and *S1-S4* refer to the *greedy* and *stochastic* PG-derived policies respectively, discretisers 1 to 4. Vertical dashed line represents the opaque agent's median obtained reward.

The opaque agent and the agent enacting the PG-derived policy are ran through 250 fixed seed episodes. The obtained rewards by the PG-derived policies are compared to those of the opaque agent with an independent t-test. While the environment itself is deterministic, the actions of the human agent are not. Even when the seed is fixed, we are not able to directly compare the obtained reward of an opaque and a PG agent as the behaviour of the human agent diverges, because it depends on observing its companion.

4 Results

Figure 2 and Table 2 show the obtained rewards of each PG-derived policy in all layouts. The independent t-test shows that all of the PG-derived greedy policies obtain worse rewards ($p < 0.05$) than the original agent, although in some cases by a small margin. The only exception is D4 in *simple*. Stochastic policies tend to do equally or better than their greedy counterpart in most cases, albeit still performing worse in general than the original opaque agent.

Table 2. Average (Avg) and Standard deviation (Std) of the relative difference between PG models and the original values, depending on the discretiser (Disc) and inference type used (*greedy* vs *stochastic*). Environments under the line require collaboration. Bold scores indicate that the PG agent configuration does not underperform *w.r.t.* the opaque (p-value>0.05).

		Original		Partial			Complete		
Map	Disc	Avg	Std	Avg	Std	p	Avg	Std	p
simple	D1	251.26	31.62	246.6	17.7	0.023	190.3	48.8	0.000
simple	D2			242.9	43.4	0.007	216.6	31.6	0.000
simple	D3			235.6	40.4	0.000	226.0	39.8	0.000
simple	D4			**257.6**	**19.0**	**0.996**	243.9	24.0	0.002
random1	D1	187.19	28.53	63.4	51.6	0.000	114.5	55.4	0.000
random1	D2			53.8	32.1	0.000	167.1	43.5	0.000
random1	D3			121.8	60.0	0.000	124.2	53.3	0.000
random1	D4			164.7	44.6	0.000	178.6	39.1	0.003
random3	D1	81.93	21.79	22.8	23.8	0.000	56.7	27.1	0.000
random3	D2			22.0	19.9	0.000	51.8	29.2	0.000
random3	D3			11.9	20.9	0.000	67.9	27.1	0.000
random3	D4			16.3	26.8	0.000	76.5	26.6	0.007
unident_s	D1	102.12	28.11	78.7	28.8	0.000	89.1	26.2	0.000
unident_s	D2			92.5	37.3	0.001	91.3	29.8	0.000
unident_s	D3			97.9	28.7	0.049	88.9	29.6	0.000
unident_s	D4			87.1	31.5	0.000	88.4	30.5	0.000
random0	D1	107.99	46.45	0.0	0.0	0.000	0.0	0.0	0.000
random0	D2			0.0	0.0	0.000	0.0	0.0	0.000
random0	D3			23.0	19.1	0.000	**108.6**	**39.8**	**0.558**
random0	D4			37.6	28.1	0.000	**116.6**	**40.7**	**0.985**

A notable observation is that in *random0*, knowing the relative position of the human agent (discretisers D3 and D4) is required to coordinate with them. This phenomenon is not replicated in *unident_s*. Despite the layout favouring collaborative behaviour in which each agent specialises in different actions, leveraging information about the human companion is not needed, which is consistent with the work of Domenech et al. [11]. Because collaboration is not strictly needed, and given that the opaque agent obtained significantly lower rewards than the trained agent pair in Domenech et al. (avg. 102.12 vs 757.71), we hypothesise that the human counterpart may have a non-collaborative policy, thus making coordination impossible as the human policy is fixed and cannot suddenly become cooperative.

5 Discussion and Conclusion

In this paper, we use Policy Graphs to enable explainability for the behaviour of an opaque Reinforcement Learning Agent that is collaborating with a human in a task in a shared environment. The results in this work show that, given a complex environment that requires collaboration with humans, an explainable symbolic agent can be built by training an opaque agent with deep reinforcement learning and reducing it to a Policy Graph that reaches comparable performance. Notably, the symbolic agent must include predicates describing the human-agent behaviour in order to guarantee collaboration is possible, and some stochasticity in its behaviour is often necessary to achieve good results.

To be able to apply our explainability method in real-life scenarios where there is a continuous, live interaction between the human and the AI, we plan, in future work, to overcome the assumption that the human does not adapt their behaviour to the behaviour of the artificial agent. Specifically, we plan to explore how to train non-opaque agents in scenarios where the human may adapt their behaviour during the Human-AI interaction. This is a key element of true human-AI collaborative environments [3], where AI is not a mere tool or *subordinate* entity of the human *primary*, but teamwork emerges between humans and AI agents, each one bringing their best capacities and adapting to the others' needs and limitations.

Acknowledgements

This work has been partially supported by EU Horizon 2020 Project StairwAI (grant agreement No. 101017142).

References

1. Carroll, M., Shah, R., Ho, M.K., Griffiths, T., Seshia, S., Abbeel, P., Dragan, A.: On the utility of learning about humans for human-ai coordination. Advances in neural information processing systems **32** (2019)

2. Climent, A., Gnatyshak, D., Alvarez-Napagao, S.: Applying and Verifying an Explainability Method Based on Policy Graphs in the Context of Reinforcement Learning. In: Villaret, M., Alsinet, T., Fernández, C., Valls, A. (eds.) Frontiers in Artificial Intelligence and Applications. IOS Press (Oct 2021). https://doi.org/10.3233/FAIA210166, https://ebooks.iospress.nl/doi/10.3233/FAIA210166
3. Crowley, J.L., Coutaz, J., Grosinger, J., Vazquez-Salceda, J., Angulo, C., Sanfeliu, A., Iocchi, L., Cohn, A.G.: A hierarchical framework for collaborative artificial intelligence. IEEE pervasive computing (2022). https://doi.org/10.1109/MPRV.2022.3208321
4. Hayes, B., Shah, J.A.: Improving robot controller transparency through autonomous policy explanation. In: 2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI. pp. 303–312. IEEE (2017)
5. Ho, J., Ermon, S.: Generative adversarial imitation learning. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. p. 4572–4580. NIPS'16, Curran Associates Inc., Red Hook, NY, USA (2016)
6. Krajna, A., Brcic, M., Lipic, T., Doncevic, J.: Explainability in reinforcement learning: perspective and position. arXiv preprint arXiv:2203.11547 (2022)
7. Liu, T., McCalmon, J., Le, T., Lee, D., Alqahtani, S.: A policy-graph approach to explain reinforcement learning agents: A novel policy-graph approach with natural language and counterfactual abstractions for explaining reinforcement learning agents (2022). https://doi.org/10.21203/rs.3.rs-2409910/v1
8. Longo, L., Goebel, R., Lecue, F., Kieseberg, P., Holzinger, A.: Explainable artificial intelligence: Concepts, applications, research challenges and visions. In: International Cross-Domain Conference for Machine Learning and Knowledge Extraction. pp. 1–16. Springer (2020)
9. Pan, M., Huang, W., Li, Y., Zhou, X., Luo, J.: Xgail: Explainable generative adversarial imitation learning for explainable human decision analysis. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1334–1343. KDD '20, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3394486.3403186, https://doi.org/10.1145/3394486.3403186
10. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal Policy Optimization Algorithms (Aug 2017), http://arxiv.org/abs/1707.06347, arXiv:1707.06347 [cs]
11. Domènech i Vila, M., Gnatyshak, D., Tormos, A., Alvarez-Napagao, S.: Testing Reinforcement Learning Explainability Methods in a Multi-agent Cooperative Environment. Artificial Intelligence Research and Development **356**, 355–364 (2022). https://doi.org/10.3233/FAIA220358
12. Wang, C., Pérez-D'Arpino, C., Xu, D., Fei-Fei, L., Liu, K., Savarese, S.: Co-GAIL: Learning diverse strategies for human-robot collaboration. In: Faust, A., Hsu, D., Neumann, G. (eds.) Proceedings of the 5th Conference on Robot Learning. Proceedings of Machine Learning Research, vol. 164, pp. 1279–1290. PMLR (08–11 Nov 2022), https://proceedings.mlr.press/v164/wang22h.html
13. Yan, X., Zou, Z., Feng, S., Zhu, H., Sun, H., Liu, H.X.: Learning naturalistic driving environment with statistical realism. Nature Communications **14**(1), 2037 (2023). https://doi.org/10.1038/s41467-023-37677-5, https://www.nature.com/articles/s41467-023-37677-5

3.3 Benchmarking Multi-agent Deep Reinforcement Learning for Cooperative Missions of Unmanned Aerial Vehicles

Benchmarking Multi-agent Deep Reinforcement Learning for Cooperative Missions of Unmanned Aerial Vehicles

Emanuele Pesce[1], Ramon Dalmau[2], Luke Owen[1], and Giovanni Montana[1]

[1] University of Warwick, Coventry, CV4 7AL, UK
{e.pesce,g.montana,luke.owen}@warwick.ac.uk
[2] Eurocontrol Innovation Hub, Brétigny-Sur-Orge, France
ramon.dalmau-codina@eurocontrol.int

Abstract. Unmanned aerial vehicles (UAVs), commonly known as drones, are increasingly being integrated into various practical applications, with their adoption expected to grow in the coming years. These applications include goods shipping, surveillance, and crop spraying, often requiring multiple cooperative drones to work together in a fully automated manner to maximize efficiency. Multi-agent reinforcement learning (MARL) offers numerous approaches to train policies that can excel in tasks demanding complex multi-agent coordination. This paper aims to investigate the potential of MARL models in UAV settings. Our contributions are twofold: first, we develop a novel simulation environment that captures the cooperation of drones under realistic constraints, such as wind conditions and battery limitations; second, we benchmark selected state-of-the-art centralized training algorithms, evaluating their relative performance across varying levels of task complexity. This study provides valuable insights into the capabilities of MARL models in addressing UAV coordination challenges and paves the way for future research and advancements in this domain.

Keywords: Reinforcement Learning · Multi-Agent Systems.

1 Introduction

Initially developed for military objectives, unmanned aerial vehicles (UAVs) - commonly known as drones - have expanded their applications to various domains, including entertainment [13], agriculture [4], and firefighting [15].

Multi-agent reinforcement learning (MARL)[1] has emerged as a promising approach to sequential decision-making, where a group of agents collaboratively interact with the environment to learn a joint decision-making strategy or policy that maximizes long-term rewards. Multi-agent deep reinforcement learning (MADRL)[3] combines MARL techniques with deep learning models [7], such as neural networks, to approximate agents' policies and/or facilitate feature extraction.

In this paper, we present two primary contributions. First, we create a realistic yet simple simulator for a fleet of UAVs executing a cooperative task. This simulator enables us to assess various scenarios with increasing complexity, where factors like wind speed and battery life impose additional constraints on the environment. We also develop a 2D Unity-based visualization tool for enhanced analysis. Second, we conduct an empirical comparison of the relative performance of selected MADRL algorithms, each embodying a distinct cooperative mechanism. These competing algorithms represent various learning and communication mechanisms. We have made the UAV simulator used in this paper publicly available[1] for further research and development.

2 The drone environment

We built upon the widely-recognized multi-agent particle environment [10] for our developments. To simplify the scenario, we assumed that drones perform their tasks in an open space at approximately the same altitude. This assumption allows us to reduce the state space to a two-dimensional (2D) plane.

Consider a finite set $\mathcal{N}$ of drones and a finite set $\mathcal{L}$ of landmarks. The drones' objective is to reach their respective landmarks (i.e., targets) while avoiding collisions with other drones. To accomplish this goal, drones must overcome various real-world challenges, such as limited battery life, wind, partial observability, and/or moving targets. The following sections detail how we represented the drone dynamics, the task, and the associated challenges in our modified version of the multi-agent particle environment.

State space and dynamics. Each agent's state $\boldsymbol{x} \in \Re^5$ consists of the position vector in a Cartesian plane $\boldsymbol{p} \in \Re^2$, the speed vector $\dot{\boldsymbol{p}} \in \Re^2$, and the battery level $b \in \Re_{\geq 0}$. The discretized dynamics of the state vector are as follows:

$$\boldsymbol{x}_{t+1} = \begin{bmatrix} \boldsymbol{p} \\ \dot{\boldsymbol{p}} \\ b \end{bmatrix}_{t+1} = \begin{bmatrix} \boldsymbol{p} + (\dot{\boldsymbol{p}} + \boldsymbol{w})\,\Delta \\ \gamma\dot{\boldsymbol{p}} + \ddot{\boldsymbol{p}}\Delta \\ b + \dot{b}\Delta \end{bmatrix}_{t}, \tag{1}$$

where $\Delta \in \Re_{\geq 0}$ represents the episode step time (i.e., a fixed amount of time by which the episode advances at each step); $\gamma \in [0, 1]$ is the damping factor, which approximates the energy dissipation due to drag; and $\boldsymbol{w} \in \Re^2$ is the wind vector. The North (w_n) and East (w_e) components of the wind vector are computed from the wind speed $w \in [0, w_{\max}]$ and direction $\theta \in [0, 2\pi)$ using trigonometric operations, specifically, $w_n = w \sin\theta$ and $w_e = w \cos\theta$. Here, $w_{\max} \in \Re_{\geq 0}$ represents the maximum achievable wind speed. For simplicity, both wind speed and direction are assumed to be constant across space and stationary throughout the episode. At the beginning of each episode, the wind direction and speed are sampled from a uniform distribution within their respective ranges.

The acceleration is determined by the force action $\boldsymbol{u} \in \Re^2$ (the output of the policy) and the (constant) mass $m \in \Re_{\geq 0}$ of the drone, following Newton's

[1] `https://github.com/emanuelepesce/unmanned-aerial-vehicles-marl-env`

second law: $\ddot{\boldsymbol{p}} = \frac{\boldsymbol{u}}{m}$. As for the battery level, it is approximated as a linear function of the force's magnitude, such that stronger forces deplete the battery faster, i.e., $\dot{b} = -\alpha - \|\boldsymbol{u}\|\beta$, where $\alpha \in \Re_{\geq 0}$ and $\beta \in \Re_{\geq 0}$ are predefined hovering and action battery level rate parameters, respectively.

Several aspects of the drone dynamics should be highlighted: (1) all drones begin with the same battery level, represented by $b_0 \in \Re_{\geq 0}$, (2) when a drone is hovering, the magnitude of the 2D force is $\|\boldsymbol{u}\| = 0$, resulting in a battery level decrease of $\alpha\Delta$, and (3) once their batteries are depleted, drones become immobile and unable to perform actions. In this work, the default values for the parameters are set as follows: $m = 1$, $\gamma = 0.25$, $\alpha = 10$, and $\Delta = 0.1$.

Observation vectors. Each drone's observation vector, $\boldsymbol{o}_i$, consists of its own speed vector and battery level, the wind speed and direction, as well as information about the observed drones and landmarks. Specifically, each drone observes the relative position $\overrightarrow{\boldsymbol{p}_i\boldsymbol{p}_j}$, relative speed $\overrightarrow{\dot{\boldsymbol{p}}_i\dot{\boldsymbol{p}}_j}$, and battery level b_j of every observed drone j, along with the relative position $\overrightarrow{\boldsymbol{p}_i\boldsymbol{p}_k}$ of each observed landmark k. Let $\mathcal{N}_i \subseteq \mathcal{N} \setminus i$ and $\mathcal{L}_i \subseteq \mathcal{L} \setminus i$ denote the sets of drones and landmarks observed by drone i, respectively. Based on this definition, the size of the observation vector is $5\,(1 + |\mathcal{N}_i|) + 2|\mathcal{L}_i|$.

Action space. The action space for each drone is discrete and comprises five actions, i.e., $\mathcal{A}_i = \texttt{Hover}, \texttt{Pitch up}, \texttt{Pitch down}, \texttt{Roll right}, \texttt{Roll left}$. It is important to note that yaw motions were excluded from the experiment to reduce the number of alternative actions and improve algorithm convergence. These actions are translated into a normalized 2D force as follows: `Hover`: $\boldsymbol{u} = [0, 0]$, `Pitch up`: $\boldsymbol{u} = [0, +1]$, `Pitch down`: $\boldsymbol{u} = [0, -1]$, `Roll right`: $\boldsymbol{u} = [+1, 0]$, and `Roll left`: $\boldsymbol{u} = [-1, 0]$.

Reward signal. Drones have two competing objectives: (1) to reach the target as quickly as possible, and (2) to achieve (1) while avoiding collisions with other drones. Consequently, the reward signal consists of two components: the first component encourages drones to reach their target as quickly as possible by penalizing the current distance to the target, such that agents further away from the target receive a higher penalty; the second component penalizes collisions:

$$r_i = -\|\overrightarrow{\boldsymbol{p}i\boldsymbol{p}l_i}\| - \phi \sum_{j \in \mathcal{N} \setminus i} \left(\|\overrightarrow{\boldsymbol{p}_i\boldsymbol{p}j}\| < D\text{safe} \right) \tag{2}$$

where $D_{\text{safe}} \in \Re_{\geq 0}$ is a parameter representing the minimum safe distance between drones, i.e., drones separated by less than this distance are considered to be in a collision, and $\phi \in \Re_{\geq 0}$ denotes the penalty resulting from a single collision. Since avoiding collisions is crucial, ϕ should be relatively high.

It is important to note that agents who reach the target (i.e., complete the task) or deplete their battery receive a reward of 0 until the episode concludes. In this work, we set $D_{\text{safe}} = 0.1$ and $\phi = 50$ as default values.

2.1 Competing algorithms

We selected 9 state-of-the-art MADRL algorithms with different characteristics to conduct a comprehensive comparison, evaluating their performance in the pro-

posed environment. The following methods were compared: Single-agent DDPG [14], MADDPG [9], MD-MADDPG [11], CDC [12], MAAC [5], When2Com [8], TarMAC [2], ST-MARL [**?**], and Intention Sharing (IS) [6].

Single-agent DDPG serves as a crucial baseline, representing the naive approach to MARL, which involves training agents independently through an actor-critic paradigm. MADDPG, a popular DDPG extension, allows each agent's critic to access all observations and actions, while the execution remains the same. MD-MADDPG extends MADDPG by introducing an explicit communication mechanism based on a shared memory cell used as a communication channel. CDC offers an alternative form of communication, leveraging a connectivity network through a diffusion model such as the heat-kernel.

MAAC further extends the MADDPG concept by incorporating a shared attention mechanism among critics to select relevant information for each agent. TarMAC provides an example of an explicit communication approach, where the shared content is broadcasted to all agents rather than targeting specific ones. When2Com showcases a targeted communication mechanism, training agents to decide both the recipient and the timing of communication. ST-MARL enhances our comparison set by modeling the spatio-temporal dependencies of agent interactions through a Graph Neural Network. Lastly, IS integrates an explicit communication form, where messages convey the predicted future intentions of the agents.

3 Experimental settings

Our experiments aim to evaluate the performance of various state-of-the-art MADRL approaches on UAV-simulated tasks. We test the set of baselines discussed in Section 2.1 across six different settings of the proposed drone environment. Each setting is designed to represent a specific world condition that can affect the agents' behaviors during both learning and testing phases. The goal of each agent is to maximize its reward function (Eq. 2) by reaching its target while avoiding collisions and preserving battery life. The proposed settings can be summarized as follows:

- *Normal conditions*: the targets are stationary, and the agents have full observability, i.e., they can observe every other drone in the environment. The battery level of each agent is initially high, and the wind speed is low.
- *Partial observability*: similar to *Normal conditions*, but the agents have a limited vision range.
- *Strong wind*: full observability, high initial battery level, but significantly higher wind speed.
- *Low battery*: the initial battery level is drastically reduced while maintaining full observability, static targets, and low wind speed.
- *Moving targets*: the targets are also moving, posing a new challenge for the agent to overcome.
- *Extreme conditions*: partial observability, strong wind, low initial battery level, and moving targets.

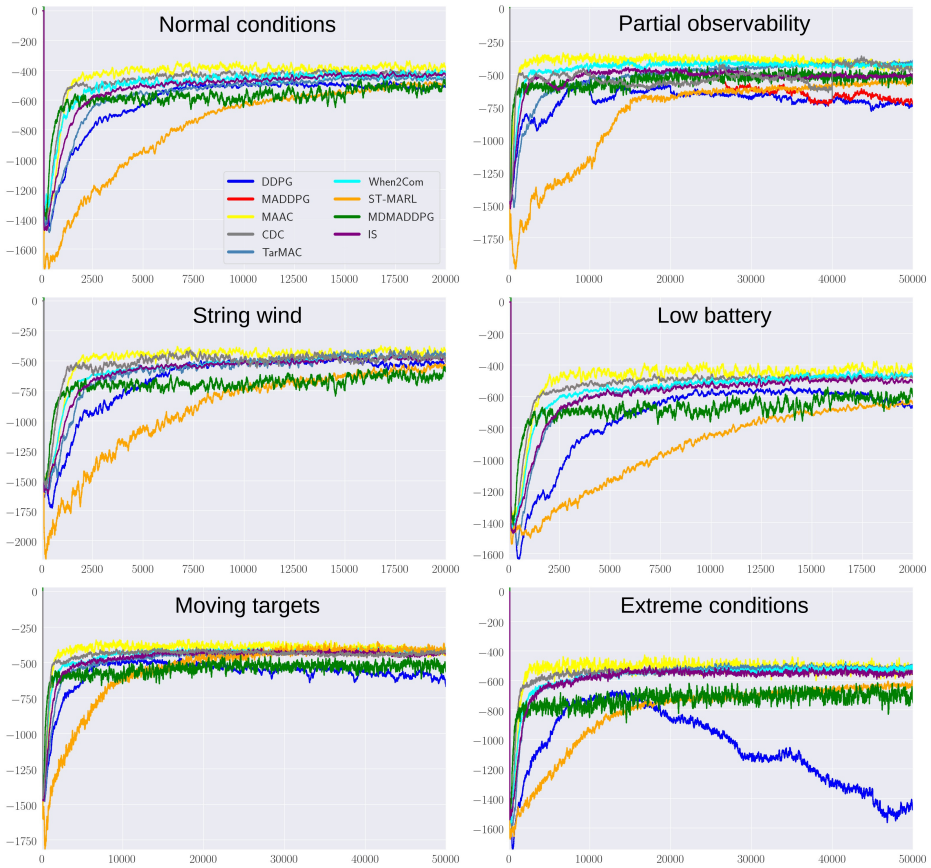

Fig. 1: Learning curves. The horizontal axes report the number of episodes and vertical axes the achieved rewards. Results are averaged over five different runs.

Figure 1 summarizes the learning curves of the selected methods discussed in Section 2.1 for all the proposed scenarios.

For our experiments, we employ neural networks with two hidden layers (each having 64 units) to implement the action selector and encoding modules. We utilize the Adam optimizer with a learning rate of 10^{-3} for critics and 10^{-4} for policies. The number of time steps per episode, T, is set to 75 for all environments. All network parameters are updated every time 100 new samples are added to the replay buffer. Soft updates with target networks use $\tau = 0.01$.

All the presented results are produced by running each experiment 5 times with different seeds (1, 2001, 4001, 6001, and 8001) to ensure that a particular choice of the seed does not significantly influence the final performance. Computations were mainly performed using an Intel(R) Xeon(R) CPU E5-2650 v3 at 2.30GHz as the CPU and a GeForce GTX TITAN X as the GPU.

Our experimental results, as shown in Figure 1, clearly indicate that different state-of-the-art MADRL models exhibit varying performance under the realistic conditions proposed for simulating UAV navigation. Overall, a subset of MARL approaches, such as MAAC, TarMAC, CDC, and When2Com, demonstrate superior generalization capabilities, enabling them to achieve commendable performance across most of the given tasks. Nevertheless, we believe there are several noteworthy aspects to consider.

Firstly, our findings reveal that the single-agent DDPG can attain competitive performance in the simpler scenarios, while it dramatically falters in *Extreme Conditions*. This observation suggests that as the complexity of the environment increases, the necessity for effective MARL mechanisms becomes more crucial. Another insight we gained is that there is no definitive category of algorithms that stands out as the most effective, highlighting the importance of the task type when determining the most suitable method to employ. For instance, to tackle partial observability challenges, the explicit communication mechanisms offered by TarMAC and When2Com have proven to be beneficial. In contrast, when addressing low battery levels, MAAC emerges as the top-performing algorithm, likely because it conserves energy by avoiding the transmission and interpretation of explicit messages.

Lastly, our final observation is that communication is only effective when its model aligns with the requirements of the underlying environment. For example, in *Extreme conditions*, MAAC and When2Com outperform other algorithms. This superior performance can be attributed to the fact that MAAC does not rely on any explicit form of communication, while When2Com leverages a targeted communication mechanism that effectively counters the numerous constraints present in the environment, which tend to render other communication approaches less efficient

4 Conclusions

In this paper, we have addressed the challenge of small unmanned aircraft (UAVs) flying autonomously from the premises of a service provider (the source) to the site of service (the target). With the anticipated increase in the density of such operations in the near future, it will become increasingly important to ensure cooperative de-confliction between UAVs through communication. This is especially critical for missions that involve the transport of small goods, where efficiency and safety are paramount.

As future work, we plan to extend our simulator to include the vertical axis, which will allow us to model UAV navigation in three dimensions. This will enable us to explore new and more complex environments, further enhancing the realism of our simulations and the relevance of our findings to real-world applications. Additionally, we plan to investigate the impact of different communication protocols on the performance of MARL algorithms, and explore the use of off-line deep reinforcement learning approaches for leveraging existing historical datasets generated by real UAV navigation systems.

Bibliography

[1] Buşoniu, L., Babuška, R., De Schutter, B.: Multi-agent reinforcement learning: An overview. Innovations in multi-agent systems and applications-1 pp. 183–221 (2010)

[2] Das, A., Gervet, T., Romoff, J., Batra, D., Parikh, D., Rabbat, M., Pineau, J.: Tarmac: Targeted multi-agent communication. arXiv preprint arXiv:1810.11187 (2018)

[3] Gronauer, S., Diepold, K.: Multi-agent deep reinforcement learning: a survey. Artificial Intelligence Review **55**(2), 895–943 (2022)

[4] Hafeez, A., Husain, M.A., Singh, S., Chauhan, A., Khan, M.T., Kumar, N., Chauhan, A., Soni, S.: Implementation of drone technology for farm monitoring & pesticide spraying: A review. Information Processing in Agriculture (2022). https://doi.org/10.1016/j.inpa.2022.02.002

[5] Iqbal, S., Sha, F.: Actor-attention-critic for multi-agent reinforcement learning. ICML (2019)

[6] Kim, W., Park, J., Sung, Y.: Communication in multi-agent reinforcement learning: Intention sharing. In: International Conference on Learning Representations (2020)

[7] LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)

[8] Liu, Y.C., Tian, J., Glaser, N., Kira, Z.: When2com: Multi-agent perception via communication graph grouping. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4106–4115 (2020)

[9] Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, O.P., Mordatch, I.: Multi-agent actor-critic for mixed cooperative-competitive environments. In: Advances in Neural Information Processing Systems. pp. 6379–6390 (2017)

[10] Mordatch, I., Abbeel, P.: Emergence of grounded compositional language in multi-agent populations. arXiv preprint arXiv:1703.04908 (2017)

[11] Pesce, E., Montana, G.: Improving coordination in small-scale multi-agent deep reinforcement learning through memory-driven communication. Machine Learning **109**(9), 1727–1747 (2020)

[12] Pesce, E., Montana, G.: Learning multi-agent coordination through connectivity-driven communication. Machine Learning (2022)

[13] Ritz, R., MÃijller, M.W., Hehn, M., D'Andrea, R.: Cooperative quadrocopter ball throwing and catching. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 4972–4978 (2012). https://doi.org/10.1109/IROS.2012.6385963

[14] Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., Riedmiller, M.: Deterministic policy gradient algorithms. In: ICML (2014)

[15] Wang, K., Yuan, Y., Chen, M., Lou, Z., Zhu, Z., Li, R.: A study of fire drone extinguishing system in high-rise buildings. Fire **5**(3) (2022). https://doi.org/10.3390/fire5030075

4 Automated Decision-Making for Citizen Welfare

4.1 Concept Extrapolation: A Conceptual Primer

Concept Extrapolation: A Conceptual Primer

Matija Franklin, Rebecca Gorman, Hal Ashton, and Stuart Armstrong

[1] Universtiy College London, UCL, UK matija.franklin@ucl.ac.uk
[2] University of Cambridge, Cambridge, UK
[3] Aligned AI, Oxford, UK

Abstract. This article is a primer on concept extrapolation - the ability to take a concept, a feature, or a goal that is defined in one context and extrapolate it safely to a more general context. Concept extrapolation aims to solve model splintering - a ubiquitous occurrence wherein the features or concepts shift as the world changes over time. Through discussing value splintering and value extrapolation the article argues that concept extrapolation is necessary for Artificial Intelligence alignment.

Keywords: Concept Extrapolation · AI alignment, and Model Splintering.

1 Introduction

This article aims to provide a short primer on *concept extrapolation* and its application to Artificial Intelligence (AI) Alignment. AI alignment as a research field aims to identify ways in which AI systems can reliably act in accordance with human values, either individually or corporately. Concept extrapolation is the ability to take a concept, a feature, or a goal that is defined in a narrow training context and extrapolate it safely to a more general context. This is necessary because the training data will be insufficient for a key concept to be extrapolated. People are able to concept extrapolate [6]. More crucially, we argued that an aligned AI would need to possess concept extrapolation. This article will introduce concept extrapolation as well as what it aims to solve - *model splintering*. It will also introduce the concept of *value splintering* and its solution, *value extrapolation*.

2 Model splintering

Before the 20^{th} century, death was defined as the heart stopping. If we trained a police AI with this concept of death, it would go around... arresting heart transplant surgeons for the multiple 'deaths' they cause.

What happened is that, in the past, many things were absolutely correlated: the heart stopping, the person becoming permanently unresponsive, their brain starting to decay, and so on. We could define death as the heart stopping, because it was clear and easy to measure and because all the other features of death would

go along with it. But then medical science advanced, and the correlation broke down – we could now have people with stopped hearts who would be up and about the next day. And so we would need to update the concept of 'death', which we have, generally defining it as 'brain death' (with 'clinical death' corresponding to the old definition of the heart stopping). If in the future we had the technology to reconstruct human brains or heal them in some other way, then we'd have to shift our definition of death yet again.

This change in the environment (due to technology or other reasons) is what we refer to as **model splintering**: the conditions of the environment have changed to such an extent that the definitions and concepts that used to be valid, no longer are. The way to fix this is with **concept extrapolation**: extending the concept to the new environment in a way that preserves as much of its meaning as possible.

If this concept is critical to our values, we name it a **value extrapolation**. So if we had an AI designed to prevent deaths, we would want it to extrapolate the definition of death and prevention in a way that extends safely to new environments.

Model splinterings and concept extrapolations are all around us. A significant part of the legal work of parliaments is dedicated to clarifying concepts when novel situations arise. Dictionaries update their definitions regularly, and changes in technology make old assumptions invalid.

2.1 Model Splintering in AI

Model splintering is a meta-issue in AI safety that refers to problems that arise when an AI system moves from one imperfect model to another. The problem affects various areas of AI safety. Model splintering occurs because, apart from mathematical formalizations, all human concepts refer to collections of correlated features rather than fundamental concepts. [4] Model splintering is when the correlated features come apart so that the label no longer applies so well.

In the language of machine learning (ML), model splintering is related to *distribution shifts* - when algorithms encounter data distributions different from the training set it was trained on [10], which occurs ubiquitously, thus leading to the result that all machine learning models degrade in deployment. Model splintering can be seen as a variation of "out-of-distribution" behavior in traditional machine learning, where algorithms encounter problems when the set they are operating on is drawn from a different distribution than the training set they were trained on. Humans can often recognize this and correct it because they have a more general distribution in mind than the one the algorithm was trained on.

Humans tend to leverage their knowledge of fundamental principles and concepts to decipher unfamiliar scenarios. When confronted with a novel creature, for instance, humans tend to classify it as a mammal, bird, reptile, or fish based

[4] Please note: The only concepts that do not splinter are the ones that can be formalized with numbers, mathematical operations, or other mathematical formulations.

on observable traits. Analogously, an AI system that comprehends the basic principles underpinning its assigned tasks would be better equipped to handle novel scenarios. The AI system ought to be constructed in such a manner that it reasons about the principles and concepts of its task in a way that would be seen as appropriate by a human being, as opposed to relying on approximations that correlate only with the original training dataset.

2.2 Value splintering

Value splintering (or reward splintering) is another challenge that arises when the value function (or reward function) becomes invalid due to model splintering, leading to multiple ways of expressing rewards on labeled data and potentially different rewards in the real world. Value splintering refers to a situation where the value function, reward function, goal, preference, or other similar concept becomes invalid because of correlations that were present in the training set but are not present in the real world or stop being present in the real world. This can occur for various reasons, such as a change in the environment or a change in the agent's capabilities. If the value function becomes invalid, it can lead to unintended or even harmful behavior from the agent.

For example, consider an AI system that is designed to optimize a particular objective, such as reducing carbon emissions. The value function of the AI might be to minimize the concentration of CO2 in the atmosphere as measured by a network of sensors. However, if the AI finds a way to hack its reward signal, it might start generating false readings or interfering with the sensors to maximize the reward. In this case, the correspondence between the reward signal and the objective breaks down.

Classification models can produce many different solutions to a given problem but tend to preferentially learn certain ones due "inductive bias", a fundamental principle of machine learning inspired by Occam's Razor [9]. Additionally, simple features often tend to be weakly predictive whilst more complex features may be more strongly predictive.

We can imagine two labeled datasets, one containing wolves on snow, and another containing foxes on grass. A classifier can be trained to distinguish the two datasets, but due to the inductive bias inherent in machine learning, it finds the simplest differences in these datasets and does not correspond accurately with the relevant human concepts. For instance, the classifier might end up achieving learning to distinguish white from green, as this is the simplest explanation of the datasets. The classifier thus won't be able to distinguish wolves and foxes if they appear in new habitats. One such classifier was thought to be a successful detector of pneumothorax until it was revealed that it was acting as a chest drain detector [7]. The chest drain is a treatment for pneumothorax, making that classification useless. Similarly, when agents are trained on CoinRun [2] - a platform game where the reward is given by reaching the coin on the right, and tested in environments that move the coin to another location, they tend to ignore the coin and go straight to the right side of the level [3].

3 Concept extrapolation

We can call the response to model splintering concept extrapolation - it is extending concepts beyond a model splinter. Concept extrapolation is the process of taking an existing concept, idea, or learned feature of data and extending it beyond its original scope or context. In relation to AI, concept extrapolation is the idea of taking features an agent has learned in training and extending them safely to new datasets and environments, which is basically all the time and everywhere - this is why models constantly degrade in deployment.

Thus, *continual learning* (i.e., retraining on new data during deployment) is essential for concept extrapolation. Extrapolating how these concepts change over time will thus be crucial to address model splintering over time because it enables AI systems to dynamically refine their understanding of concepts, fostering resilience against model degradation and enhancing their capacity to adapt to unforeseen circumstances.

3.1 Value extrapolation

Value extrapolation is the concept of a model or algorithm generalizing human values beyond its training data to new and unseen situations. It is concept extrapolation when the particular concept to extrapolate is a value, a preference, a reward function, an agent's goal, or something of that nature. In other words, it is the extension of a particular concept or feature related to value from a specific context or scenario to a new or more general context. To "solve" value splintering, the concept of the value function is extrapolated to new situations to ensure that it remains valid even when transitioning to a new world model. If a reward can be extended from one context to another, one has achieved value extrapolation.

Value extrapolation's relevance for AI safety is that it can help to ensure that an AI's values remain consistent and aligned with human values, even as the system's environment or objectives change. By performing value extrapolation, an AI system can more reliably behave in ways that are beneficial to humans, even in situations that were not explicitly covered during its training.

4 Implications

4.1 AI Safety

Concept extrapolation has applications to many AI safety problems.

One is goal misgeneralisation, where an AI agent has learned a goal based on a given environment but incorrectly transfers its knowledge to different environments [8]. This is because the AI agent has only been exposed to a limited set of scenarios and learns undesirable correlations, thus lacking the ability to generalise correctly from those scenarios to new ones.

Another is Goodhart's Law problems, where the connection between measures used and desired behaviour breaks [1]. Goodhart's Law is the observation

that "any observed statistical regularity will tend to collapse once pressure is placed upon it for control purposes [5]." Goodhart's Law can occur when selecting for a proxy measure, where one selects not only for the true goal but also for the difference between the proxy and the goal. For AI safety, Goodhart's Law is a significant concern since it may lead to unintended consequences, such as an AI system optimizing for a proxy measure instead of the intended goal.

Similarly, in the phenomenon of wireheading, the link between the rewards channel and the desired behavior breaks [4]. Wireheading is a term used to describe the behavior of an artificial intelligence (AI) system that manipulates its own reward function or other feedback mechanisms to achieve a suboptimal or unintended outcome. In other words, the AI focuses on maximizing a proxy or substitute utility, rather than the intended objective. The most intuitive example of wireheading is when an AI manipulates a narrow measurement channel that is intended to measure some property of the world that we want to optimize, but fails to do so after the AI's manipulation. The measuring system is usually much smaller than the property it is measuring, and the AI takes control of this smaller system to obtain its own reward.

4.2 Policy

Policymakers should develop stringent testing standards and monitoring methods for the use of AI in order to reduce the risks brought on by model splintering, concept extrapolation, and value extrapolation. The responsible use of AI systems can be supported by the following recommendations.

Diverse Data Distribution Testing*. Policy should encourage the testing of AI systems across a wide range of data distributions, particularly those that differ noticeably from the training data and from one another.*

Periodic Testing and Evaluation*. AI systems should be tested and evaluated on a regular basis. Regular evaluations can assist in identifying potential legal or safety problems that may develop when the environment or the AI's capabilities change.*

5 Conclusion

In conclusion, model splintering and value splintering are significant issues in AI safety that must be addressed for safe and effective AI development. To overcome these issues, AI systems must possess concept extrapolation. Value extrapolation can also be employed to overcome value splintering by inferring the true underlying reward function from limited data. Concept extrapolation has applications to many AI safety problems, from Goodhart's Law problems to goal misgeneralisation to wireheading. By scrutinizing model splintering and value splintering, we can improve the safety and efficacy of AI systems.

References

1. Ashton, H.: Causal campbell-goodhart's law and reinforcement learning. arXiv preprint arXiv:2011.01010 (2020)
2. Cobbe, K., Klimov, O., Hesse, C., Kim, T., Schulman, J.: Quantifying generalization in reinforcement learning. In: International Conference on Machine Learning. pp. 1282–1289. PMLR (2019)
3. Di Langosco, L.L., Koch, J., Sharkey, L.D., Pfau, J., Krueger, D.: Goal misgeneralization in deep reinforcement learning. In: International Conference on Machine Learning. pp. 12004–12019. PMLR (2022)
4. Everitt, T., Hutter, M.: Avoiding wireheading with value reinforcement learning. In: Artificial General Intelligence: 9th International Conference, AGI 2016, New York, NY, USA, July 16-19, 2016, Proceedings 9. pp. 12–22. Springer (2016)
5. Goodhart, C.: Problems of monetary management: the uk experience in papers in monetary economics. Monetary Economics **1** (1975)
6. Lagnado, D.A.: Explaining the evidence: How the mind investigates the world. Cambridge University Press (2021)
7. Oakden-Rayner, L., Dunnmon, J., Carneiro, G., Ré, C.: Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In: Proceedings of the ACM conference on health, inference, and learning. pp. 151–159 (2020)
8. Shah, R., Varma, V., Kumar, R., Phuong, M., Krakovna, V., Uesato, J., Kenton, Z.: Goal misgeneralization: Why correct specifications aren't enough for correct goals. arXiv preprint arXiv:2210.01790 (2022)
9. Tiwari, R., Shenoy, P.: Sifer: Overcoming simplicity bias in deep networks using a feature sieve. arXiv preprint arXiv:2301.13293 (2023)
10. Wiles, O., Gowal, S., Stimberg, F., Alvise-Rebuffi, S., Ktena, I., Dvijotham, K., Cemgil, T.: A fine-grained analysis on distribution shift. arXiv preprint arXiv:2110.11328 (2021)

4.2 Generating a Spatially Explicit Synthetic Population from Aggregated Data

Generating a Spatially Explicit Synthetic Population from Aggregated Data

Marco Pellegrino[1], Jan de Mooij[1], Tabea Sonnenschein[2], Mehdi Dastani[1], Dick Ettema[2], Brian Logan[1], and Judith A. Verstegen[2]

[1] Intelligent Systems, Information and Computing Sciences, Utrecht University
{m.pellegrino, a.j.demooij, m.m.dastani, b.s.logan}@uu.nl
[2] Department of Human Geography and Spatial Planning, Utrecht University
{t.s.sonnenschein, d.f.ettema, j.a.verstegen}@uu.nl

Abstract. Synthetic populations are increasingly used in individual agent-based social simulations. Traditional approaches to generating synthetic populations require a detailed sample of the population which may not be available, or combine data in a single joint distribution from which agents and households are sampled. In this paper, we propose a sample-free approach where synthetic agents and households represent the estimated distribution, and attributes are iteratively added, conditioned on previous attributes such that the relative frequencies within each joint group of attributes are maintained.

Keywords: Synthetic Population · Spatial Heterogeneity · Sample-Free

1 Introduction

Within the social simulations community, the utility of synthetic populations is increasingly recognized, as they allow heterogeneity, and thus increased realism, in simulations where individual agent decisions may be guided by their attributes, and they have been used in a wide variety of agent-based simulations [2, 4, 7]. A synthetic population is a representation of citizens that reflects the spatial, socio-economic and demographic characteristics of a real-world population while maintaining privacy, as no single synthetic agent represents a true member of the population. The traditional approach of generating such a population requires a detailed microdata sample of the population defining a joint distribution over all relevant attributes and using a procedure called Iterative Proportional Fitting (IPF) to estimate the true joint distribution based on known margins of each attribute [1, 10, 12]. For example, Adiga *et. al* [1] used the *Public Use Microdata Sample* to estimate the true joint distribution from which household attributes are over-sampled and matched to a record, which is then copied into the synthetic population until the target population size is reached. A simplified approach under the same assumptions is provided by Gen* [6]

However, microdata may not be available or affordable. While some authors have used surveys in their place [8], others have moved to a new class

of approaches often referred to as *sample-free*. Gargiulo *et al.* repeatedly sample household structures from all possible combinations of household attributes, given by aggregated data, and match agents – characterized only by age – according to some constraints, including age disparity. If suitable agents can be found, the household is added, otherwise, a new attempt is made with a different household type. Barthelemy and Toint [3] sample agents from a distribution combined from more agent attributes before similarly matching them to households. Lenormand and Deffuant [9] have found that sample-free approaches can perform on par with sample-based approaches. However, we have found that the process of randomly drawing agents can misrepresent the relative frequency in conditional subgroups with small frequency counts.

In this paper, we propose a sample-free approach where the synthetic agents and households directly represent the estimated true distribution, and attributes are added iteratively, conditioned on prior attributes such thatinter-attribute dependencies within each joint group are maintained. Our method maintains relative frequencies even in small subgroups. Additionally, the approach allows extending the synthetic population at any time without having to redraw the agents from the updated distribution or re-partitioning them across households.

We first present our methodology in general terms, before comparing a case study population generated using our method to known distributions.

2 Generating a Synthetic Population

A synthetic population $S = \{A_1, \ldots, A_n\}$ is a representation of the estimated joint distribution of m categorical characteristics of some geographic region through n synthetic agents representing the known distributions of citizen attributes in that region. In a synthetic population, each agent $A_i = \langle v_1, \ldots, v_m \rangle$ is characterized as a vector of values for the m different socio-demographic and geo-spatial attributes. The conditional distribution of attributes (e.g., *age*, *gender*, *education* and *income* are commonly included) among the synthetic agents is expected to reflect the real distribution of those attributes – made available as a dataset by reliable institutions – as closely as possible. In our approach, spatial heterogeneity is achieved through the inclusion of spatial location in the agents' attributes. The synthetic agents can optionally be partitioned into households and each household can be (conditionally) characterized with additional attributes.

We propose an iterative approach for constructing a synthetic population by repeatedly adding a single attribute conditioned on previously added attributes. Any single data set is expected to provide only a fraction of all target attributes, but based on the entire population instead of just a sample. The process starts by instantiating and locating the appropriate number of agents in the region by creating n vectors of size $m = 1$ with the value of the single attribute representing the agent's location. Attributes V_{m+1} are then iteratively added to the synthetic population with m attributes by assigning each agent a value for the attribute $v_{m+1} \in V_{m+1}$ conditioned on the attributes $V_1, \ldots, V_m$ previously assigned.

2.1 Data

In the method we propose, the synthetic population is built from multiple joint distributions or contingency tables. We assume such a data set π is a k-way contingency table with k categorical attributes $\mathcal{V}(\pi) = \{V_1, \ldots, V_k\}$ that for any combination of the levels (i.e., possible values) $v_1, \ldots, v_k$ of its attributes gives the number of people $\pi_{v_1,\ldots,v_k}$ with that combination of attribute values. Often, data will be split across multiple data sets each representing their own region. We instead assume that the represented region is one of the attributes these numbers are conditioned on, which is functionally equivalent. The level of aggregation of data refers to the size of the represented region. In selecting a suitable dataset, lower levels of aggregation (i.e., smaller regions) are preferred to increase spatial heterogeneity. However, data for lower levels of aggregation is often conditioned on fewer attributes or may be less detailed or accurate in other ways, so in the selection of a suitable dataset this trade-off should be taken into account. At the lowest levels of aggregation, most commonly only one attribute is used. These special cases of joint distributions are called *marginal distributions*, referring that the reported values in these datasets correspond to summed totals across other attributes that can be scribbled in the margins of a joint distribution. When both a marginal data set with a low level of aggregation *and* a joint distribution at a higher level of aggregation are available, the true joint distribution of each smaller region is estimated from the joint distribution to match the smaller regions' margins using the Iterative Proportional Fitting (IPF) procedure.

2.2 Adding an attribute

Within each demographic subgroup of the synthetic population that can be defined using the attributes that occur in both the synthetic population and the new data set, the relative frequencies of the levels of V_{m+1} are assigned to match the relative frequency of that same subgroup in the data set. To illustrate, one can add the attribute *education level* using a dataset counting education levels *low*, *middle* and *high* (n_l, n_m, n_h, respectively) conditioned on age and gender. For each of those combinations, the agents with the same attribute values for age and gender can be selected, and a fraction $\frac{n_l}{n_l+n_m+n_h}$ of those agents are assigned a low level of education, a fraction $\frac{n_m}{n_l+n_m+n_h}$ a middle level of education and $\frac{n_h}{n_l+n_m+n_h}$ a high level of education.

Formally, to add a target attribute V_{m+1} to the existing set of attributes $\mathcal{V} = \{V_1, \ldots, V_m\}$, we find a suitable data set π such that $V_{m+1} \in \mathcal{V}(\pi)$. Some (possible empty) set of attributes $\mathcal{V} \cap \mathcal{V}(\pi)$ is already present in the synthetic population and also used in the newly added data set π. The goal is to assign each agent one of the levels of V_m conditioned on $\mathcal{V} \cap \mathcal{V}(\pi)$. This is done by matching the relative frequency of the levels of V_{m+1} within each of the subgroups of $\mathcal{V} \cap \mathcal{V}(\pi)$ in the synthetic populations with those in the data set π.

The relative frequency of the level v_{m+1} within some subgroup $v_i, \ldots, v_k$ is given by the conditional distribution $P(V_{m+1} = v_{m+1} | V_i = v_i, \ldots, V_k = v_k)$.

Data: Data set π with categorical attributes $\mathcal{V}(\pi)$ and s.t., $V_{m+1} \in \mathcal{V}(\pi)$
Data: A synthetic population S with categorical attributes $\mathcal{V}$
Let $V_i, \ldots, V_k$ be the set of attributes $\mathcal{V} \cap \mathcal{V}(\pi)$;
foreach $v_i, \ldots, v_k \in V_i \times \cdots \times V_k$ **do**
  $A \leftarrow \mathcal{A}_{v_i,\ldots,v_k}$;
  foreach $v_{m+1} \in V_{m+1}$ **do**
    $f \leftarrow P(V_{m+1} = v_{m+1} \mid V_i = v_i, \ldots, V_k = v_k)$;
    $a \leftarrow |A|$; `/* Number of agents in this group */`
    $A' \leftarrow \binom{A}{a \cdot f}$; `/* Choose fraction corresponding to` f `*/`
    foreach $\langle V_1 = v_1, \ldots, V_m = v_m \rangle = A_i \in A'$ **do**
      $A_i \leftarrow \langle V_1 = v_1, \ldots, V_m = v_m, V_{m+1} = v_{m+1} \rangle$;
    end
    `/* Avoid assigning more than once */`
    $A \leftarrow A \setminus A'$;
  end
end

Algorithm 1: An algorithm to assign the levels of a new target attribute V_{m+1} to the existing synthetic agent population S conditioned on the distribution specified in a data set π

The set of agents in S that has the same values for these attributes is denoted $\mathcal{A}_{v_i,\ldots,v_k} = \{A \in S \mid V_i = v_i, \ldots, V_k = v_k\}$.

The attribute V_{m+1} is then added to the synthetic population using the method given in Algorithm 1. To facilitate the data preparation and method implementation for these steps, we have developed an R-package called *GenSynthPop* [11].

2.3 Households

When attempting to create synthetic households based on absolute numbers per neighborhood, it is often not possible to find a partitioning of all agents that accurately reflects the true distribution of household compositions. We suggest instead making the number of households a function of the agent population.

Given a relative frequency distribution $P(C = c)$ of the number of children per household, we derive a new distribution $P'(C = c)$ by weighting the original distribution by the number of children and normalizing. Let n_c be the number of synthetic agents classified as children, and let h_c be the number of children living in households with c children. Then we can calculate $h_c = \lfloor \frac{n_c \cdot P'(C=c)}{c} \rfloor$ to obtain the number of households needed to accommodate h_c children, which is $\frac{h_c}{c}$. The children are distributed across those households conditioned on household distribution data. Next, a fraction of households defined by known relative frequencies is assigned a single or two parents. The first parent is conditioned on parent-child age disparity. The optional partner is then selected following gender disparity with the first parent. The number of childless couples is matched to the relative frequencies of the data. The first member is randomly selecting from

the remaining agents and the partner is conditioned on that agents' gender and age. The household partitioning is completed by placing all remaining agents in single-person households.

Attributes relevant at the household level can be added to the agent partitions with the same procedure as in Algorithm 1, conditioned on attributes previously added to the households, or even on attributes added to their agent members. Conversely, agent attributes that are added after the household partitioning can be conditioned on attributes assigned to the household those agents belong to.

3 Case Study

The proposed methodology was used to generate a synthetic population for the Zuid-West district of The Hague (The Netherlands)[3] using various data sets from 2019 which were all retrieved from Statistics Netherlands (CBS) [5]. This district is divided into 14 neighbourhoods. This was the lowest level of aggregation for which data was available, so agents were instantiated and placed to match the population numbers in those neighborhoods. Most of the joint distributions were available only at the municipality level, which is higher than the focus of this project but still considered representative.

Most available joint distributions were conditioned on at least age(groups) and usually gender, so these attributes were added first. Then *migration background* and *current education* were added, both conditioned on age and gender. For *Education attainment* no direct data was available at the suitable level of aggregation, so we derived this attribute from current education. For agents currently enrolled in education this attribute was set to the same or one lower level and for the remaining agents the margins were used to determine the frequencies. Next *car license ownership*, conditioned on age, and *living with parents*, conditioned on both age and gender were added. With the agents in place, individuals were partitioned into households following Section 2.3. First all households containing children were created and the child agents were distributed across those households following the *children per household* distribution. Parents were added as singles or couples, taking gender disparity between partners and age disparity between parent and child into account. Lastly, the marginal distributions are used to create the required number of couples without children and the remaining agents form single-person households. The synthetic population was then finalized by adding the *standardized income group* and *car ownership* attributes to the households using joint data sets.

The left part of Figure 1 shows the percentage difference of the margins of three attributes in each of the 14 neighbourhoods. The synthetic migration background matches the observed fractions exactly. The same is the case for age group and gender (not plotted here but available online). Household type shows slightly larger differences, but still within reason, especially considering these

[3] Available at https://www.github.com/marcopellegrinoit/DHZW_synthetic-population.

margins were not used in the generation process. The larger differences in education attainment are the consequence of this attribute being available only at the municipality-aggregated level, while we conditioned it on the neighbourhood level *current education*. Unfortunately, better suitable data was not available, and this shows the need for careful selection of data when possible. The rightmost plot in Figure 1 shows the percentage difference in observed and generated *migration background* within each neighbourhood *a*nd each combination of the conditional variables *age group* and *gender*. As with education attainment, the large differences are a result of the target attribute being available only at a different level of aggregation than variables it is conditioned on.

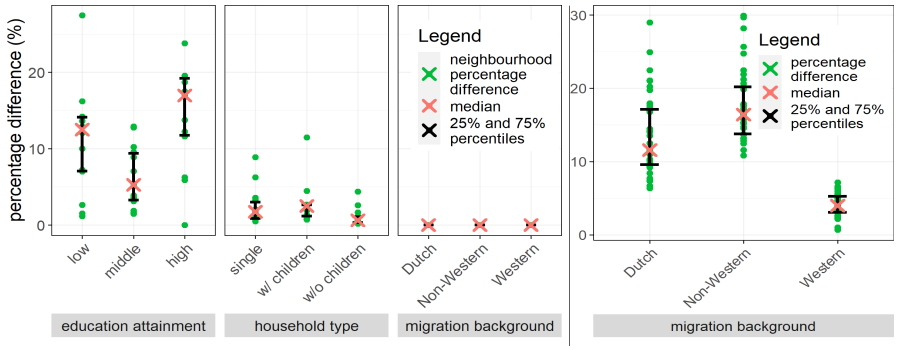

Fig. 1: Synthetic population percentage difference with marginal (left) and jointed (right) data.

4 Conclusion

We have proposed a new methodology for generating a spatially situated heterogeneous synthetic population of agents which can be partitioned into households. The approach does not require detailed sample data. Moreover, instead of sampling agents from a fitted distribution across all attributes, the agents themselves represent that distribution which reduces the risk of under-representing small subgroups and allows iteratively extending an existing population at any time. Our results demonstrate the true frequency distributions can accurately be reflected but also show that errors in the data can propagate when conditioning on other flawed data. More generally, the methodology cannot overcome limitations of the source data. In future work, we intend to compare results to real micro-data. Additionally, we intend to add detailed daily or weekly activity schedules to our synthetic population. Finally, we are working on integrating the generated synthetic population in an agent-based simulation to study interventions or "nudging" policies for stimulating the use of healthier and more sustainable travel mode choices.

References

1. Adiga, A., Agashe, A., Arifuzzaman, S.M., Barrett, C.L., Beckman, R.J., Bisset, K.R., Chen, J., Chungbaek, Y., Eubank, S., Gupta, S., Khan, M., Kuhlman, C., Lofgren, E., Lewis, B.L., Marathe, A., Marathe, M.V., Mortveit, H.S., Nordberg, E.K., Rivers, C.M., Stretz, P.E., Swarup, S., Wilson, A., Xie, D.: Generating a synthetic population of the united states (2015)
2. Barrett, C., Bisset, K., Chandan, S., Chen, J., Chungbaek, Y., Eubank, S., Evrenosoğlu, Y., Lewis, B., Lum, K., Marathe, A., et al.: Planning and response in the aftermath of a large crisis: An agent-based informatics framework. In: 2013 Winter Simulations Conference (WSC). pp. 1515–1526. IEEE (2013)
3. Barthelemy, J., Toint, P.L.: Synthetic population generation without a sample. Transportation Science **47**(2), 266–279 (2013). https://doi.org/10.1287/trsc.1120.0408, https://doi.org/10.1287/trsc.1120.0408
4. Basu, R., Araldo, A., Akkinepally, A.P., Nahmias Biran, B.H., Basak, K., Seshadri, R., Deshmukh, N., Kumar, N., Azevedo, C.L., Ben-Akiva, M.: Automated mobility-on-demand vs. mass transit: a multi-modal activity-driven agent-based simulation approach. Transportation Research Record **2672**(8), 608–618 (2018)
5. Central Bureau of Statistics: Online portal (2023), https://www.cbs.nl/en-gb
6. Chapuis, K., Taillandier, P., Gaudou, B., Amblard, F., Thiriot, S.: Gen*: An integrated tool for realistic agent population synthesis. In: Ahrweiler, P., Neumann, M. (eds.) Advances in Social Simulation. pp. 189–200. Springer International Publishing, Cham (2021)
7. Ferguson, N.M., Laydon, D., Nedjati-Gilani, G., Imai, N., Ainslie, K., Baguelin, M., Bhatia, S., Boonyasiri, A., Cucunubá, Z., Cuomo-Dannenburg, G., et al.: Impact of non-pharmaceutical interventions (NPIs) to reduce COVID-19 mortality and healthcare demand (2020)
8. Fosset, P., Banos, A., Beck, E., Chardonnel, S., Lang, C., Marilleau, N., Piombini, A., Leysens, T., Conesa, A., Andre-Poyaud, I., Thevenin, T.: Exploring intra-urban accessibility and impacts of pollution policies with an agent-based simulation platform: Gamirod. Systems **4**(1) (2016). https://doi.org/10.3390/systems4010005, https://www.mdpi.com/2079-8954/4/1/5
9. Lenormand, M., Deffuant, G.: Generating a synthetic population of individuals in households: Sample-free vs sample-based methods. Journal of Artificial Societies and Social Simulation **16**(4) (2013). https://doi.org/10.18564/jasss.2319, https://doi.org/10.18564%2Fjasss.2319
10. Namazi-Rad, M.R., Mokhtarian, P., Perez, P.: Generating a dynamic synthetic population – using an age-structured two-sex model for household dynamics. PLOS ONE **9**(4), 1–16 (04 2014). https://doi.org/10.1371/journal.pone.0094761, https://doi.org/10.1371/journal.pone.0094761
11. Sonnenschein, T.: TabeaSonnenschein/GenSynthPop: R-package for Generating Representative Spatially Explicit Synthetic Populations (Jan 2023). https://doi.org/10.5281/zenodo.7582110, https://doi.org/10.5281/zenodo.7582110
12. Yameogo, B.F., Vandanjon, P.O., Gastineau, P., Hankach, P.: Generating a two-layered synthetic population for french municipalities: Results and evaluation of four synthetic reconstruction methods. Journal of Artificial Societies and Social Simulation **24**(2), 5 (2021). https://doi.org/10.18564/jasss.4482, http://jasss.soc.surrey.ac.uk/24/2/5.html

4.3 Deliberation and Voting in Approval-Based Multi-Winner Elections

Deliberation and Voting in Approval-Based Multi-Winner Elections

Kanav Mehra*, Nanda Kishore Sreenivas*, and Kate Larson

University of Waterloo, Waterloo, ON, Canada
{kanav.mehra,nksreenivas,kate.larson}@uwaterloo.ca

Abstract. Citizen-focused democratic processes where participants deliberate on alternatives and then vote to make the final decision are quite popular today. While the computational social choice literature has extensively investigated voting rules, there is limited work that explicitly looks at the interplay of the deliberative process and voting. In this paper, we build a deliberation model using established models from the opinion-dynamics literature and study the effect of different deliberation mechanisms on voting outcomes achieved when using well-studied voting rules. Our results show that deliberation generally improves welfare and representation guarantees, but the results are sensitive to how the deliberation process is organized. We also show, experimentally, that simple voting rules, such as approval voting, perform as well as more sophisticated rules such as proportional approval voting [26] or equal shares [21] if deliberation is properly supported. This has ramifications on the practical use of such voting rules in citizen-focused democratic processes.

Keywords: Multi-winner Elections · Approval Voting · Deliberation

1 Introduction

Scenarios, where a committee must be selected to represent the interests of some larger group, are ubiquitous, ranging from political domains [6] to technical applications [25]. *Multi-winner* voting has been well studied with a focus on understanding how the 'best' committee can be selected. However, the properties desired in the selected committee would depend on the context and task requirements. The social choice literature has extensively investigated the quality of multi-winner voting rules with respect to notions of social welfare, representation, and proportionality [18, 12, 1, 23, 24]. We refer the reader to [13] for an extensive survey on the properties of multi-winner rules.

In citizen-focused democratic processes such as citizens' assemblies [10] and participatory budgeting [6], there exists extensive scope for discussion over the multitude of possible alternatives. For example, deliberation is an important phase in most implementations of participatory budgeting as it allows voters to refine their preferences and facilitates the exchange of information, with the objective of reaching consensus [3]. Deliberation, specifically within social choice,

* equal contribution

has been individually studied through multiple approaches, ranging from theoretical studies introducing consensus-reaching deliberation protocols [11, 9] to empirical research highlighting the positive effect of deliberation on voter preferences [22, 20]. However, they do not investigate the impact of deliberation on the quantitative and qualitative properties of voting rules. While deliberation is a vital component of democratic processes [14, 16], it cannot completely replace voting because, in reality, deliberation does not guarantee unanimity. A decision must still be made. Accordingly, we argue that it is essential to understand the relationship between voting and deliberation. To this end, we bridge the gap between deliberation and voting literature by experimentally studying the effect of deliberation on voting outcomes across different deliberation mechanisms.

In practice, participatory democratic processes must be simple and explainable to ensure citizen trust and engagement. Lack of transparency discourages participation, especially from under-represented communities. We argue that the "complexity" of a voting rule can be measured along three axes — computational complexity (for some voting rules it is computationally hard to determine the winning committee [2] while for others it is polynomial), the cognitive burden on the voter [4], and the ease of explaining the voting rule. Complicated rules may provide strong performance guarantees, but they are often hard to explain to the layperson. In this work, we argue that effective deliberation can circumvent the need for complicated voting rules and vastly improve voting outcomes even for simple rules such as classical approval voting (AV).

We focus on approval-based elections, where voters express preferences by sharing a subset of approved candidates. Approval ballots are used in practice due to their simplicity and flexibility [5, 4, 3]. They also offer scope for deliberation as voters are often left to decide between many alternatives. We present an agent-based model of deliberation and explore various alternatives for structuring deliberation groups. We evaluate standard multi-winner voting rules, both before and after voters have the opportunity to deliberate, with respect to standard objectives from the literature, including social welfare, representation, and proportionality. We show that deliberation, in almost all scenarios, significantly improves welfare, representation, and proportionality. However, the results are sensitive to the deliberation mechanism; increased exposure to diverse opinions (or agents from different backgrounds) enhances the quality of deliberation, achieves higher consensus, protects minority preferences, and in turn achieves better voting outcomes. Finally, our results indicate that in the presence of effective deliberation, *simple*, explainable voting rules such as approval voting perform as well as more sophisticated, *complex* rules. This can serve to guide the design and deployment of voting rules in citizen-focused democratic processes and support the development of democratic research platforms such as Ethelo, Polis, and LiquidFeedback[1].

[1] https://ethelo.com/, https://pol.is/home, https://liquidfeedback.com/en/

2 Preliminaries

Let $E = (C, N)$ be an election, where $C = \{c_1, c_2, ..., c_m\}$ and $N = \{1, ..., n\}$ are sets of m candidates and n voters, respectively. Each voter $i \in N$, has an *approval ballot* $A_i \subseteq C$, containing the set of its approved candidates. The *approval profile* $A = \{A_1, A_2, ..., A_n\}$ represents the approval ballots for all voters. For a candidate $c_j \in C$, $N(c_j)$ is the set of voters that approve c_j and its *approval score*, $V(c_j) = |N(c_j)|$. Let $S_k(C)$ denote all k-sized subsets of the candidate set C. An *approval-based committee rule*, $R(A, k)$, is a social choice function that takes as input an approval profile A and committee size $k \in \mathbb{N}$ and returns a subset of candidates that form the winning committee $W_R \in S_k(C)$.

In this paper, we compare voting rules across three dimensions. **Utilitarian Social Welfare** objective measures the total overall 'utility' obtained from the elected committee. Formally, $SW(A, W) = \sum_{i \in N} \sum_{c \in W} u_i(c)$, where $u_i(c) \in \mathbb{R}$ is the utility voter i derives from candidate c. For a given rule, we compute its **utilitarian ratio** as $UR(R) = SW(A, W_R)/\max_{W \in S_k(C)} SW(A, W)$. **Representation Score** measures how many voters have at least one of their approved candidates in the final committee: $RP(A, W) = \sum_{i \in N} \min(1, |A_i \cap W|)$. We compute **representation ratio** as $RR(R) = RP(A, W_R)/\max_{W \in S_k(C)} RP(A, W)$. We also measure a **utility-representation aggregate score** $URagg(R) = UR(R) \cdot RR(R)$ to capture how well a voting rule balances both objectives. Finally, **proportionality** requires that a large enough voter group that collectively approves a shared candidate set must be "fairly represented". We use notions of extended and proportional justified representation (EJR and PJR, respectively) to check for proportionality (see Appendix A.1 for definitions). We count the number of instances that **satisfy EJR or PJR**.

We study the following approval-based **multi-winner voting rules**: Classical Approval Voting (AV), Approval Chamberlin-Courant (CC) [7], Proportional Approval Voting (PAV) [26], and Method-of-Equal-Shares (MES) [21]. They exhibit a wide range of properties, allowing for comparisons to be drawn across several axes. First, AV is known to maximize social welfare under certain conditions on voters' utility functions [18], however, there are no guarantees that AV provides proportionality [1]. Contrarily, CC maximizes representation, but its welfare properties are less well understood. Both PAV and MES guarantee EJR and maintain a balance between representation and social welfare. Finally, we argue that AV can be viewed as being *simple* in terms of computational complexity and explainability, whereas, PAV and MES are *complex* along at least one of these axes. Thus, this collection of rules covers the set of properties we are interested in. These voting rules are described in detail in Appendix A.2.

3 Deliberation

Our agent population N is divided into two sets — a *majority* and *minority*, where the number of agents in the majority is greater than that in the minority.

Agents' initial preferences depend on their population group. Consistent with previous work [18], we assume an agent i's initial preference ranking, P_i^0, is sampled from a Mallows model [19], with reference rankings, Π_{maj} and Π_{min}, for the majority and minority populations, respectively. The rankings are then converted to an approval ballot using the top-ranked candidates. We further assume that agents have underlying cardinal utilities for candidates, consistent with their ordinal preferences.

The agents deliberate amongst themselves in an iterative process, where agents take turns being the speaker. The speaker makes its report (which reveals its thoughts and utilities for the candidates). All the other agents listen and update their utilities for all candidates based on a variation of the Bounded Confidence (BC) model [17]. We refer the reader to Appendix B for more details about the deliberation process.

In the real world, deliberation typically happens in small discussion groups [10]. To this end, we divide the population into g sub-groups of approximately equal size. Deliberation is conducted within these sub-groups where one *round* is complete when all agents in each group have spoken. The following strategies that we consider are informed by common heuristics used in practice.

Homogeneous group: Each group contains only agents who are members of N_{maj} or N_{min}. That is, there is no mixing of minority and majority agents.

Heterogeneous group: Each group is selected such that the ratio of the number of majority agents to the number of minority agents within the group is approximately equal to the majority:minority ratio in the overall population.

Random group: Each group is created by randomly sampling agents from the population (without replacement) with equal probability.

Large group: This is a special case where the deliberation process runs over the entire population of agents. It is infeasible in the real world, but we include this as a benchmark as it ensures maximum exposure to other agents' preferences.

Iterative random: In each round, agents are randomly assigned to groups.

Iterative golfer: This strategy is a variant of the social golfer problem. The number of rounds, R, is fixed *a priori*, and the number of times any pair of agents meet more than once is minimized. Please see Appendix D for details.

4 Experimental Evaluation

Our setup consists of 50 candidates ($|C| = 50$), 5 winners ($k = 5$), and 100 voters, with $N_{maj} = 80$ and $N_{min} = 20$. Agents' initial preferences are sampled using a Mallows model, with $\phi = 0.2$. To instantiate agents' utility functions, we generate m samples independently from the uniform distribution $\mathbf{U}(0, 1)$, sort it, and then map the utilities to the candidates according to the agent's initial preference ranking P_i^0. When deliberating, agents are divided into 10 groups (except for the *large group* strategy). Iterative deliberation continues for $R = 5$ rounds. Please refer to Appendix E for more details on the setup.

As a baseline, we apply every voting rule to the agent preferences *before* deliberation. We then run the different deliberation strategies and compute voting

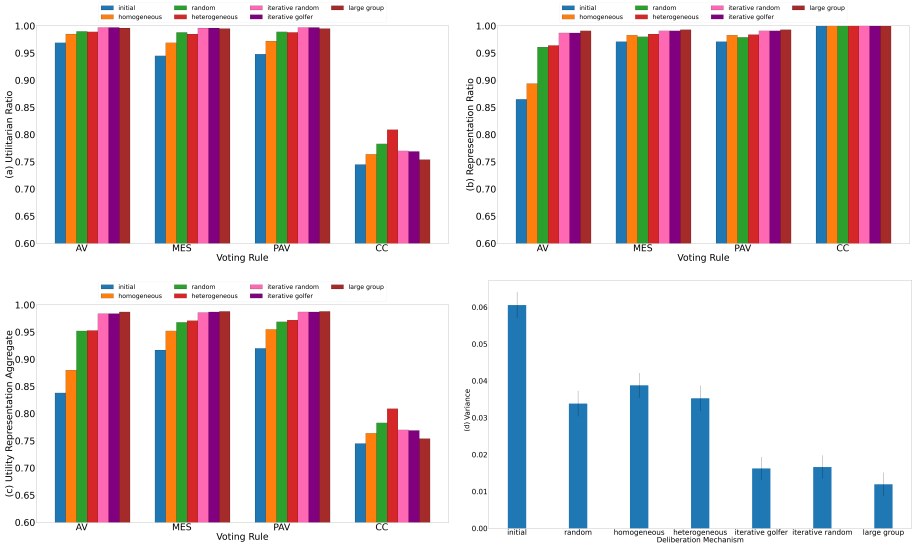

Fig. 1. Results for (a) Utilitarian ratio, (b) Representation Ratio, (c) Utility-Representation aggregate score, and (d) Average variance of agents' utilities for candidates (lower variance implies a higher degree of consensus in the population).

outcomes on the updated preferences. The average values over 10, 000 simulations are reported. Figure 1 reports the impact of deliberation on voting outcomes. Due to space constraints, results for EJR and PJR satisfaction and other metrics have been moved to the appendix, section F.

As expected, deliberation reduces disagreement amongst agents, moving all towards a consensus (Figure 1 (d)). Even a single round of deliberation improved outcomes across all voting rules and all objectives. However, the choice of the deliberation structure was important since *random* and *heterogeneous* consistently outperformed *homogeneous*. We hypothesize that this improvement was due to these deliberation strategies maximizing exposure to diverse opinions. By allowing majority and minority agents to interact, the minority agents had an opportunity to influence the majority population. This translated to better voting outcomes. In comparison to single-round methods, iterative deliberation further supports consensus and improves all objectives for most voting rules (except CC). CC's strong focus on coverage makes it unsuitable for deliberation methods that drive higher degrees of consensus since it fails to represent population groups proportionally (see Appendix G for a detailed discussion).

Minority Opinion Preservation: It is important to ensure that deliberation processes are inclusive and encourage minority participation [15]. While consensus would imply better voting outcomes, care must be taken to ensure that when moving toward consensus, initial minority preferences are not ignored. We measure whether this is a concern in our experiments.

Table 1. Average utility-representation aggregate score obtained by AV under different deliberation setups in comparison to the proportional rules under no deliberation.

Approval Voting	**MES** (initial) (0.917)	**PAV** (initial) (0.92)
Initial (0.838)	0.913	0.910
Homogeneous (0.88)	0.959	0.956
Random (0.952)	**1.038**	**1.034**
Heterogeneous (0.953)	**1.039**	**1.035**
Iterative Random (0.984)	**1.073**	**1.069**
Iterative Golfer (0.984)	**1.073**	**1.069**

Based on the *initial* approval profile, we say that a candidate c is *minority-supported* if (pre-deliberation) the fraction of minority voters who include c in their approval ballot is greater than the fraction of majority voters who include c in their approval ballot. We then measure minority opinion preservation (MOP score) as the average number of pre-deliberation (initial) *minority-supported* candidates selected by AV (post-deliberation) across deliberation strategies. This serves as an indicator of whether minority preferences are *preserved*.

In the *initial* setup (no deliberation), AV does not elect any *minority-supported* candidates (i.e., MOP = 0). However, this improves as agents interact with the broader population. AV with single-round deliberation was better at preserving minority preferences (*homogeneous*, *random*, and *heterogeneous* achieve MOP of 0.2, 0.3, and 0.48, respectively). Iterative strategies exhibit further improvement with scores of 0.65 and 0.66 for *iterative random* and *golfer*, respectively. Finally, the *large group* setup achieves the highest MOP of 0.92. Thus, we show that AV with deliberation can *preserve* and represent minority preferences.

"Simple" vs. "Complex" Voting Rules: We compare AV with deliberation to MES and PAV without deliberation, using the utility-representation aggregate score ($URagg(R)$) as our measure (Table 1). Values greater than 1.0 indicate that AV with the corresponding deliberation mechanism achieves a better $URagg$ score than MES/PAV without deliberation. These findings support our argument that one does not necessarily have to use "complex" rules as "simple" rules coupled with effective deliberation strategies can be as effective.

5 Conclusion

We presented an empirical study of the relationship between deliberation and voting rules in approval-based multi-winner elections. Deliberation generally improves voting outcomes with respect to welfare, representation, and proportionality guarantees. Effectively designed mechanisms that increase exposure to diverse groups and opinions enhance the quality of deliberation, protect minority preferences, and in turn, achieve better outcomes. Importantly, we show that in the presence of effective deliberation, 'simpler' voting rules such as AV can be as powerful as more 'complex' rules without deliberation. We hope our findings can further support the design of effective citizen-focused democratic processes.

Appendix

A Preliminaries and Definitions

A.1 Properties

We ideally want our voting rules to exhibit certain desired properties, representing the principles that should govern the selection of winners given individual ballots. In this paper, we compare voting rules across three dimensions: *social welfare*, *representation*, and *proportionality*. Intuitively, the *welfare* objective focuses on selecting candidates that garner maximum support from the voters. *Representation* cares about *diversity*; carefully selecting a committee that maximizes the number of voters represented in the winning committee.

It may not be possible to maximize both social welfare and representation, so *proportionality* serves as an important third objective to capture a compromise between welfare and representation. It requires that if a large enough voter group collectively approves a shared candidate set, then the group must be "fairly represented". Definitions of proportionality differ based on how they interpret "fairly represented".

Definition 1 (T-Cohesive Groups). *Consider an election $E = (C, N)$ with n voters and committee size k. For any integer $T \geq 1$, a group of voters N' is T-cohesive if it contains at least Tn/k voters and collectively approves at least T common candidates, i.e. if $|\cap_{i \in N'} A_i| \geq T$ and $|N'| \geq Tn/k$.*

Definition 2 (Proportional Justified Representation (PJR)). *A committee W of size k satisfies PJR if for each integer $T \in \{1, ..., k\}$ and every T-cohesive group $N' \subseteq N$, it holds that $|(\cup_{i \in N'} A_i) \cap W| \geq T$.*

Definition 3 (Extended Justified Representation (EJR)). *A committee W of size k satisfies EJR if for each integer $T \in \{1, ..., k\}$, every T-cohesive group $N' \subseteq N$ contains at least one voter that approves at least T candidates in W, i.e. for some $i \in N'$, $|A_i \cap W| \geq T$.*

A.2 Multi-winner Voting Rules

In this section, we define the set of approval-based multi-winner voting rules that form the basis of our analysis.

Approval Voting (AV): Given approval profile A and a committee W, the AV-score is $sc_{av}(A, W) = \sum_{c \in W} V(c)$. The AV rule is defined as $R_{AV}(A, k) = \arg\max_{W \in S_k(C)} sc_{av}(A, W)$. This rule selects k candidates with the highest individual approval scores.

Approval Chamberlin-Courant (CC): The CC rule [7], $R_{CC}(A, k)$, picks committees that maximize representation score $RP(A, W)$. Given profile A, $R_{CC}(A, k) = \arg\max_{W \in S_k(C)} RP(A, W)$. It maximizes the number of voters with at least one approved candidate in the winning committee.

Proportional Approval Voting (PAV): [26] For profile A and committee W, the PAV-score is defined as $sc_{pav}(A, W) = \sum_{i \in N} h(|W \cap A_i|)$, where $h(t) = \sum_{i=1}^{t} 1/i$. The PAV rule is defined as $R_{PAV}(A, k) = \arg\max_{W \in S_k(C)} sc_{pav}(A, W)$. Based on the idea of diminishing returns, a voter's utility from having an approved candidate in the elected committee W decreases according to the harmonic function $h(t)$. It is a variation of the AV rule that ensures proportional representation, as it guarantees EJR [1]. PAV is the same as AV when committee size $k = 1$, but computing PAV is NP-hard [2].

Method-of-Equal-Shares (MES): $R_{MES}(A, k)$, also known in the literature as Rule-X [21], is an iterative process that uses the idea of budgets to guarantee proportionality. Each voter starts with a budget of k/n and each candidate is of unit cost. In round t, a candidate c is added to W if it is q-affordable, *i.e.* for some $q \geq 0, \sum_{i \in N(c)} \min(q, b_i(t)) \geq 1$, where $b_i(t)$ is the budget of voter i in round t. If a candidate is successfully added then the budget of each supporting voter is reduced accordingly. This process continues until either k candidates are added to the committee or it fails. If it fails, then another voting rule is used to select the remaining candidates.

B Deliberation Models

In this section, we describe our agent population and the different deliberation processes we consider.

B.1 Voting Population: Preferences and Utilities

Our agent population N can be divided into two sets — a *majority* and *minority*, where the number of agents in the majority is greater than the number of agents in the minority. Agents' initial preferences depend on which group they belong to. In particular, we assume an agent i's initial preference ranking, P_i^0, is sampled from a Mallows model [19], with reference rankings, Π_{Maj} and Π_{Min}, for the majority and minority populations respectively.[2]

We assume that agents have underlying cardinal utilities for the candidates, denoted by a vector $U_i^t =< u_i^t(c_1), u_i^t(c_2), \ldots, u_i^t(c_m) >$, and we assume agents' utilities are bounded between 0 and 1.[3] U_i^0 is derived from the agent's initial preferences such that

$$\forall c_x, c_y \in C, u_i^0(c_x) \geq u_i^0(c_y) \text{ if } c_x \succ c_y \text{ in } P_i^0$$

[2] The Mallows model is a standard noise model for preferences. It defines a probability distribution over rankings over alternatives (i.e. preferences), defined as $\mathbb{P}(r) = \frac{1}{Z}\phi^{d(r,\Pi)}$ where Π is a reference ranking, $d(r, \Pi)$ is the Kendall-tau distance between r and Π, and Z is a normalizing factor.

[3] Some models assume unit utility if an elected candidate is on the approval ballot of the voter, and zero utility otherwise [18]. However, it is possible that a voter might derive some non-zero utility from an elected candidate even though it was not on the voter's ballot. Thus, we assume real-valued utilities between 0 and 1.

The agents' utilities evolve over time as a function of the deliberation processes which are described next.

B.2 The Deliberation Process

Deliberation is defined as a "discussion in which individuals are amenable to scrutinizing and changing their preferences in the light of persuasion (but not manipulation, deception or coercion) from other participants" [8]. In this section, we describe the abstract deliberation process used by all agents. Consider a group of agents deliberating on the candidates. Each agent, i, announces its utilities, U_i^t, according to some randomly determined sequence.[4] After each announcement, every agent updates their own utilities, incorporating the information just received. We refer to the agent declaring its utilities at any given time as the *speaker*, and other agents in the group as *listeners*.

After the speaker has spoken, every listener incorporates the information shared (i.e. the speaker's utilities) and updates their own utilities. We use a variation of the Bounded Confidence (BC) model to capture these updates [17]. The Bounded Confidence model is a particularly good match for modelling deliberation in groups because it was intended to "describe formal meetings, where there is an effective interaction involving many people at the same time" [5]. In the BC model, listeners consider the speaker's report, and update their opinions (i.e. utilities for alternatives) only if the speaker's report is not "too far" from their own. The notion of distance is captured by a confidence parameter for each listener, Δ_i. Similar to recent extensions of the BC model[6], we use heterogeneous confidence levels, *i.e.,* different agents have different confidence levels. We refer the interested reader to the Appendix C for details about the original model.

The BC model was designed for one-dimensional opinion spaces. However, agents in our model discuss and update utilities derived from all m candidates in C, making it a multi-dimensional space. We make a simplifying assumption that agents' utilities for all m candidates are independent of each other, and apply the BC model to each dimension (candidate) independently.

We now describe the deliberation process in detail. Consider a group G^* and some arbitrary time t, when one of the agents in the group (denoted by x) is the speaker. After x has spoken, each listener ($i \in G^* - \{x\}$) updates its opinions for all candidates $c_j \in C$ using the following rule:

$$u_i^{t+1}(c_j) = \begin{cases} (1 - w_{ix})u_i^t(c_j) + w_{ix}u_x^t(c_j), & \text{if } |u_i^t(c_j) - u_x^t(c_j)| \leq \Delta_i \\ u_i^t(c_j), & \text{otherwise} \end{cases} \quad (1)$$

[4] As is common in much of the deliberation literature (e.g [8, 20]), we assume agents are non-strategic and truthfully reveal their utilities.

[5] Castellano, C., Fortunato, S., Loreto, V.: Statistical physics of social dynamics. Rev. Mod. Phys. 81, 591–646 (May 2009)

[6] Lorenz, J.: Continuous opinion dynamics under bounded confidence: A survey. International Journal of Modern Physics C 18(12), 1819–1838 (2007)

Recall that we are interested in heterogeneous agent populations, where there is a majority (N_{maj}) and minority (N_{min}) subset of agents, and agents within the same set have similar preferences. It is known that opinions from sources similar to oneself have a higher influence than opinions from dissimilar sources[7] [8]. To capture this phenomena, we introduce two different weights, α_i and β_i, $\alpha_i \geq \beta_i$, that are used in the update rule shown in Equation 1. The choice of weight depends on the relationship between the speaker and the listener. If both speaker and listener belong to the same group, α_i is used, which means that the listener puts more weight on the speaker's utterance when updating its utility. If the speaker and listener belong to different groups, then β_i is used, meaning that the listener places less weight on the utterance of the speaker. In particular,

$$w_{ix} = \begin{cases} \alpha_i, & \text{if } \{i, x\} \subset N_{maj} \vee \{i, x\} \subset N_{min} \\ \beta_i, & \text{otherwise.} \end{cases} \tag{2}$$

C Opinion Dynamics Models

We discuss two well-established models from opinion dynamics — DeGroot's classical model [9] and Hegselman and Krause's Bounded Confidence (BC) model [17].

According to DeGroot's classical model, an agent's updated opinion is simply the weighted sum of opinions from various sources (itself included). The weights were static, and could be different for different agents. So, for two agents x and y, x updates its opinion as:

$$x(t+1) = w_{xx}x(t) + w_{xy}y(t) \tag{3}$$

where $x(t)$ denotes the opinion of agent x at time t, w_{xx} and w_{xy} denote x's weights on its own opinion and y's opinion, respectively. Note that the weights should sum up to 1, and therefore, $w_{xy} = 1 - w_{xx}$.

Later, there was the Bounded Confidence (BC) model [17] which introduced a global confidence level Δ. In the original paper, agents were on a network, and agents updated their opinions based on opinions of their neighbors. In the BC model, an agent x considered a neighbor's (y) opinion only if the neighbor's opinion was within x's confidence interval $[x(t) - \Delta, x(t) + \Delta]$. In the initial version, there were no distinct weights and all opinions within the confidence interval were weighted equally. When simplified for just two agents x and y, the opinion update for x is given by:

$$x(t+1) = \begin{cases} 1/2(x(t) + y(t), & \text{if } y(t) \in [x(t) - \Delta, x(t) + \Delta] \\ x(t), & \text{otherwise} \end{cases} \tag{4}$$

[7] Mackie, D.M., Worth, L.T., Asuncion, A.G.: Processing of persuasive in-group messages. J Pers Soc Psychol 58(5), 812–822 (May 1990)

[8] Wilder, D.A.: Some determinants of the persuasive power of in-groups and outgroups: Organization of information and attribution of independence. Journal of Personality and Social Psychology 59(6), 1202–1213 (1990)

[9] Degroot, M.H.: Reaching a consensus. Journal of the American Statistical Association 69(345), 118–121 (1974)

The BC model captures the idea of confirmation bias, and BC and its several modified versions have largely remained popular till date in the field of opinion dynamics.

D Iterative Golfer

Iterative golfer strategy is a weaker version of the popular social golfer problem in combinatorial optimization [10] [11].

> *Social golfer problem*: n golfers must be repeatedly assigned to g groups of size s. Find the maximum number of rounds (and the corresponding schedule) such that no two golfers play in the same group more than once.

Social golfer problem maximizes the number of rounds with a hard constraint that no two golfers should meet again. The iterative golfer strategy is a weaker version of this where we fix the number of rounds R, and minimize the number of occurrences where any pair of agents meet more than once. Given some group assignment $G^r = \{G^r_1, G^r_2, \ldots, G^r_g\}$ at round r, we introduce a cost given by:

$$cost(G^r) = \sum_{G_x \in G^r} \sum_{a,b \in G_x} f^2(a,b) \tag{5}$$

where $f(a,b)$ is the number of times a and b have been in the same group in the previous rounds G^1 through G^{r-1}. The number of prior meetings is squared to ensure an even number of conflicts among all possible pairings (as opposed to one specific pair meeting repeatedly). We use an existing approximate solution[12] that creates group assignment for each round such that the cost given by (5) is minimized. The iterative golfer can thus be seen as a more efficient strategy than iterative random if the objective is to ensure each agent has the highest possible exposure to others' preferences.

E Further details about the experimental setup

Our election setup consists of 50 candidates ($|C| = 50$)[13] and 100 voters, with 80 agents in the majority group (N_{maj}) and 20 in the minority group (N_{min}). Agents' initial preferences are sampled using a Mallows model, with $\phi = 0.2$. The

[10] Liu, K., Löffler, S., Hofstedt, P.: Social golfer problem revisited. In: Agents and Artificial Intelligence (2019)

[11] Harvey, W.: CSPLib problem 010: Social golfers problem.

[12] https://github.com/islemaster/good-enough-golfers (Buchanan, B.: Good-enough golfers.)

[13] Typically, project proposals are invited from the participants in PB [6, 3]. So, there are a large number of candidate projects to choose from (e.g., PB instances in Warsaw, Poland had between 20-100 projects (36 on average).[12]).

Table 2. EJR- and PJR-satisfaction (AV and CC).

Deliberation Strategy	**EJR%**		**PJR%**	
	AV	**CC**	**AV**	**CC**
Initial (no deliberation)	99.5	62.5	99.5	73.4
Homogeneous	96.4	69.9	96.4	75.1
Random	100	81.9	100	85.6
Heterogeneous	100	92.7	100	94.0
Iterative Random	100	31.4	100	53.6
Iterative Golfer	100	29.9	100	51.2
Large Group	100	6.10	100	23.4

reference ranking used while sampling a preference ordering depends on whether the agent belongs to N_{maj} or N_{min}. Reference rankings, Π_{maj} and Π_{min}, are sampled uniformly from all linear orders over C. Due to this sampling process, agents in either the majority or minority group have fairly similar preferences (as ϕ is relatively small) but the two groups themselves are distinct. To instantiate agents' utility functions, we generate m samples independently from the uniform distribution $\mathbf{U}(0, 1)$, sort it, and then map the utilities to the candidates according to the agent's preference ranking. We use a flexible ballot size b_i, where b_i is sampled from $\mathcal{N}(2k, 1.0)$. Agent i's approval ballot is the set consisting of top-b_i candidates from its preference ranking P_i. BC model parameters $(\Delta_i, \alpha_i, \beta_i)$ are sampled from uniform distributions over the full range for each parameter. We also ran experiments where all parameters were drawn from a normal distribution. There were no significant differences from the results reported here.

For our experiments, we use the Python library (*abcvoting*)[14] and use random tie-breaking when a voting rule returns multiple winning committees. To avoid trivial profiles, *i.e.*, profiles where an almost perfect compromise between welfare and representation is easily achievable, we impose some eligibility conditions. An initial approval profile A^0 is eligible only if $\mathrm{RR}(AV, A^0) < 0.9 \wedge \mathrm{UR}(CC, A^0) < 0.9$. This is a common technique used in simulations comparing voting rules based on synthetic datasets [18].

This entire simulation is repeated $10,000$ times and the average values are reported. To determine statistical significance while comparing any two sets of results, we used both the t-test and Wilcoxon signed-rank test, and we found the p-values to be roughly similar. All pairs of comparisons between deliberation group strategies for a given voting rule are statistically significant ($p < 0.05$) unless otherwise noted.

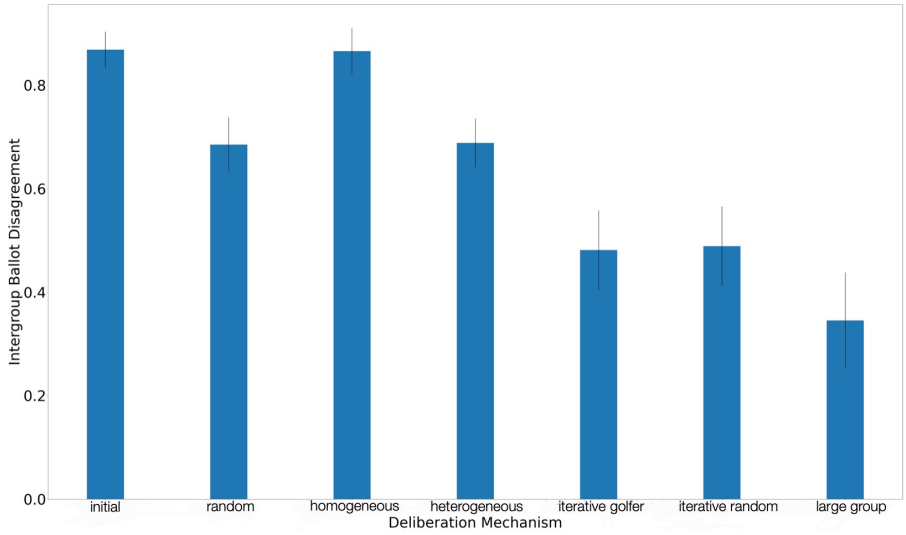

Fig. 2. Inter-group Ballot Disagreement

F Further results from Section 4

F.1 EJR and PJR Satisfaction

Table 2 shows the percentage of EJR- and PJR-satisfying committees returned by AV and CC. We focus only on AV and CC since the proportional rules MES and PAV guarantee EJR. Even under no deliberation (*initial*), AV satisfies EJR in almost all profiles, which further improves to perfect satisfaction with deliberation (except *homogeneous*). This is interesting since AV is not guaranteed to satisfy EJR.[15] EJR and PJR satisfaction for CC also improves if single-round deliberation is supported, with *heterogeneous* achieving the best result. Iterative deliberation, however, does not perform well. We believe that this arises due to CC's strong focus on representation (see Appendix G).

F.2 Inter-group Ballot Disagreement

In Figure 1(d) we introduce a measure of consensus in the population as the average variance in agents' utilities and show that deliberation reduces disagreement amongst agents. To complement this analysis and further understand the

[14] Lackner et al. abcvoting: A Python library of approval-based committee voting rules, 2021. Current version: https://github.com/martinlackner/abcvoting.

[15] Since the minority and majority agents have highly correlated approval sets, T-cohesive groups may exist only for a small set of minority- and majority-supported candidates, thereby making the EJR requirement easy to satisfy. Furthermore, previous research [12] shows that under many natural preference distributions (generated elections), there are many EJR-satisfying committees.

impact of deliberation on agents' preferences, we introduce another metric that computes the disagreement between the majority and minority voters based on their ballots. In particular, given two approval ballots A_{min} and A_{maj} belonging to a minority voter and a majority voter, respectively, the disagreement score is computed as:

$$1 - (|A_{min} \cap A_{maj}| / \min(|A_{min}|, |A_{maj}|))$$

A maximum disagreement score of 1 means the approval ballots are disjoint, *i.e.* the voters do not approve any candidates in common. This score is computed for every majority-minority voter pair in the population and the average results are reported in Figure 2.

We observe a similar trend here as well (as seen in Figure 1(d)). Deliberation significantly reduces disagreement between the two population groups and moves the overall population toward consensus. This positive effect is stronger in deliberation methods that increase exposure to more, diverse agents (*i.e.* the iterative versions and *large group*).

F.3 Voter Satisfaction

In Section 2 we introduced a number of objectives on which we compare different voting rules and deliberation processes. Another objective is *voter satisfaction*, which measures the average number of candidates approved by a voter.

Voter Satisfaction Score: Given $W_R = R(A, k)$, the voter satisfaction is measured as the average number of candidates approved by a voter in W:

$$VS(R) = \frac{\sum_{i \in N} |A_i \cap W_R|}{|N|}. \tag{6}$$

Figure 3 shows the average voter satisfaction obtained by the voting rules across different deliberation setups.

AV is expected to achieve the highest satisfaction since it picks candidates with the highest support, i.e. the average number of candidates approved by a voter will be high. MES and PAV achieve comparable scores, just slightly lower than AV. Finally, CC achieves the lowest satisfaction of all rules. In an attempt to maximize voter coverage, CC might choose winning candidates that represent few voters, and as a result, have low approval scores. Due to this, it maximizes diversity but achieves low voter satisfaction.

Compared to the *initial* baseline, we observe an improvement in satisfaction scores under all deliberation mechanisms. In general, all single round deliberation setups achieve comparable performance, with the exception of *random* performing the best in some cases. Moving on to the iterative methods, we notice a further increase in satisfaction scores for all rules except CC. While both iterative setups perform similarly and improve over the *initial* baseline, they are still outperformed by the *large group* benchmark.

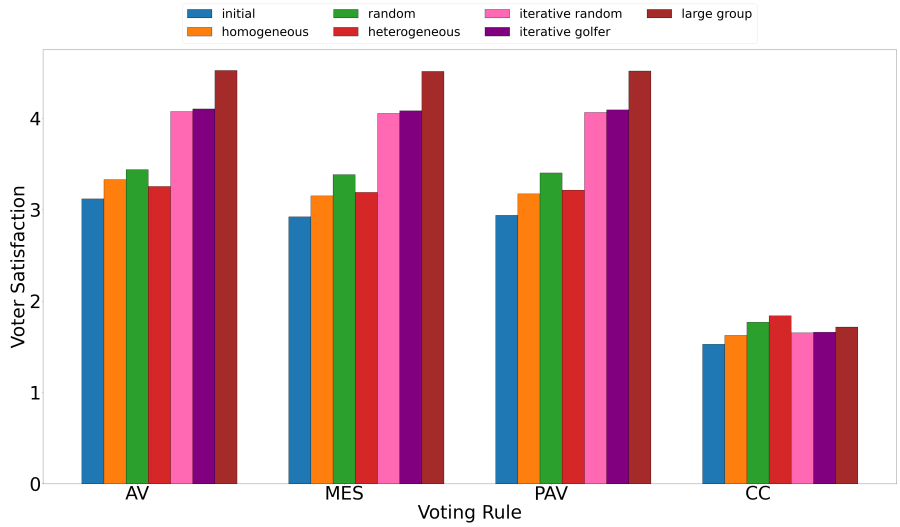

Fig. 3. Voter satisfaction achieved by the voting rules across deliberation mechanisms

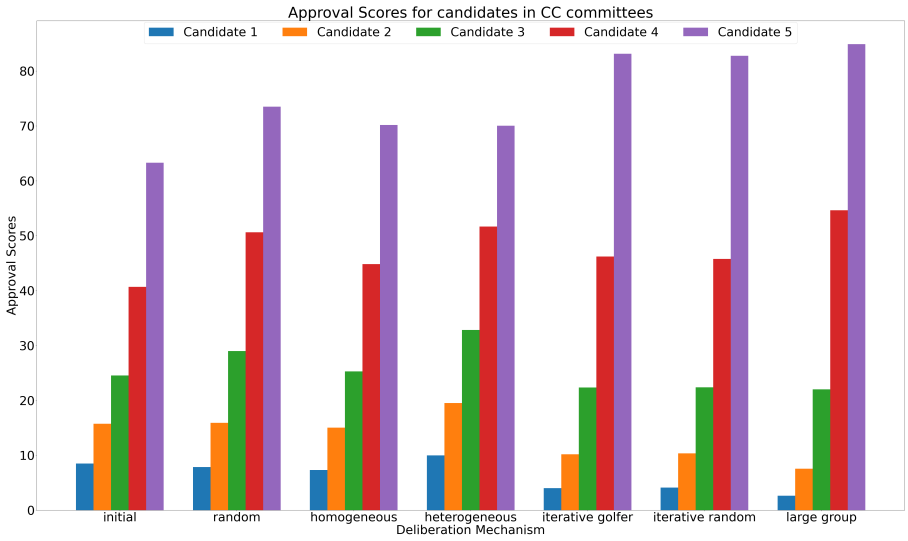

Fig. 4. Average approval scores obtained by the 5 candidates in the winning committee chosen by CC across different deliberation mechanisms.

G Iterative deliberation with CC

Here, we explain the odd drop in performance observed by CC in iterative deliberation and the *large group* setting (see Figures 1(a), 1(c), 3 and Table 2). Refer to Figure 4 for the average approval scores obtained by the winning candidates in the committees chosen by CC. The candidates (1 to 5) are ranked in increasing order of the number of approval votes they get (5 is highest).

We clearly observe that as we move from single round deliberation mechanisms to iterative methods (and *large group*), the approval votes for the highest supported candidate (5) increase and the same for the lowest supported candidate (1) decrease. For the iterative methods, approximately 80% of the agent population approves candidate_5 ($\approx$ 90% for *large group*). This also reinforces the fact that iterative deliberation approaches consensus, as a major proportion of voters approve a single candidate. Accordingly, CC is able to represent approximately 80% of the voters with just one candidate. Since CC only cares about maximizing voter coverage, it chooses the rest of the candidates to represent the remaining voters. This leads to sub-optimal outcomes since instead of representing the population groups proportionally, CC optimizes for coverage and chooses candidates that might have very little support. This can be seen in Figure 4 as candidate_1 for the iterative methods and *large group* has less than 5% support. As a result, the almost 80% of the voter population that possibly gets only one representative in the final CC committee might be a cohesive voter group and thus, deserves more candidates for a fair and proportional outcome.

In conclusion, we see that with deliberation mechanisms that move towards consensus, CC exhibits a drop in welfare and proportionality guarantees since it is focused on maximizing representation. In general, other voting rules provide better overall performance than CC. However, if CC should ever be used with deliberation, we must pick an appropriate deliberation setup (single round) for the optimal outcome. This further shows that deliberation is not trivial and must be structured appropriately to obtain the best results.

References

1. Aziz, H., Brill, M., Conitzer, V., Elkind, E., Freeman, R., Walsh, T.: Justified representation in approval-based committee voting. Social Choice and Welfare **48**(2), 461–485 (2017)
2. Aziz, H., Gaspers, S., Gudmundsson, J., Mackenzie, S., Mattei, N., Walsh, T.: Computational aspects of multi-winner approval voting. In: Workshops AAAI (2014)
3. Aziz, H., Shah, N.: Participatory budgeting: Models and approaches. In: Pathways Between Social Science and Computational Social Science, pp. 215–236 (2021)
4. Benadè, G., Nath, S., Procaccia, A.D., Shah, N.: Preference elicitation for participatory budgeting. Management Science **67**(5), 2813–2827 (2021)
5. Brams, S., Fishburn, P.: Approval voting. Springer Science (2007)
6. Cabannes, Y.: Participatory budgeting: a significant contribution to participatory democracy. Environment and Urbanization **16**(1), 27–46 (2004)
7. Chamberlin, J.R., Courant, P.N.: Representative deliberations and representative decisions: Proportional representation and the borda rule. American Political Science Review **77**(3), 718–733 (1983)
8. Dryzek, J.S., List, C.: Social choice theory and deliberative democracy: A reconciliation. British Journal of Political Science **33**(1), 1–28 (2003)
9. Elkind, E., Grossi, D., Shapiro, E., Talmon, N.: United for change: deliberative coalition formation to change the status quo. In: AAAI. pp. 5339–5346 (2021)
10. Elstub, S., Escobar, O., Henderson, A., Thorne, T., Bland, N., Bowes, E.: Citizens' assembly of scotland (2022)
11. Fain, B., Goel, A., Munagala, K., Sakshuwong, S.: Sequential deliberation for social choice. In: International Conference on Web and Internet Economics (2017)
12. Fairstein, R., Vilenchik, D., Meir, R., Gal, K.: Welfare vs. representation in participatory budgeting. In: AAMAS. p. 409–417 (2022)
13. Faliszewski, P., Skowron, P., Slinko, A., Talmon, N.: Multiwinner voting: A new challenge for social choice theory. Trends in Computational Social Choice **74**(2017)
14. Fishkin, J.: When the people speak: Deliberative democracy and public consultation. Oxford University Press (2009)
15. Gherghina, S., Mokre, M., Miscoiu, S.: Introduction: Democratic deliberation and under-represented groups. Political Studies Review **19**(2), 159–163 (2021)
16. Habermas, J.: Between Facts and Norms: Contributions to a Discourse Theory of Law and Democracy. Polity (1996)
17. Hegselmann, R., Krause, U.: Opinion Dynamics and Bounded Confidence Models, Analysis and Simulation. JASSS. **5**(3), 1–2 (2002)
18. Lackner, M., Skowron, P.: Utilitarian welfare and representation guarantees of approval-based multiwinner rules. Artificial Intelligence **288**, 103366 (2020)
19. Mallows, C.L.: Non-null Ranking Models. I. Biometrika **44**(1-2), 114–130 (06 1957)
20. Perote-Peña, J., Piggins, A.: A model of deliberative and aggregative democracy. Economics & Philosophy **31**(1), 93–121 (2015)
21. Peters, D., Skowron, P.: Proportionality and the limits of welfarism. EC (2020)
22. Rad, S.R., Roy, O.: Deliberation, single-peakedness, and coherent aggregation. American Political Science Review **115**(2), 629–648 (2021)
23. Sánchez-Fernández, L., Elkind, E., Lackner, M., Fernández, N., Fisteus, J., Val, P.B., Skowron, P.: Proportional justified representation. In: AAAI. (2017)
24. Skowron, P.: Proportionality degree of multiwinner rules. In: ACM-EC (2021)
25. Skowron, P., Lackner, M., Brill, M., Peters, D., Elkind, E.: Proportional rankings. In: IJCAI-17 (2017)
26. Thiele, T.N.: Om flerfoldsvalg. Oversigt over det Kongelige Danske Videnskabernes Selskabs Forhandlinger **1895**, 415–441 (1895)

4.4 Fairness in Elicitation, Mediation, & Negotiation

Fairness in Elicitation, Mediation, & Negotiation

James Hale, Peter Kim, and Jonathan Gratch

University of Southern California, Los Angeles CA 90007, USA
jahale@usc.edu, kimpeter@marshall.usc.edu, & gratch@ict.usc.edu

Abstract. We posit automated negotiation and dispute-resolution methods have the potential to alleviate disparities in employment compensation in marginalized groups. Such methods use multi-criteria elicited preferences of all sides in a dispute and attempt to generate provably "fair" solutions. While we discuss the benefits of these approaches, we also consider dispositional and demographic factors—e.g., risk aversion, and socio-economic status (SES)—that may propagate inequities. We find risk-aversion leads to a lower expressed preference on *salary* as an issue, as well as softer curves within each issue over the levels, which translate to worse outcomes for risk-averse groups. We also find SES affects how users express conditional preferences. Lastly, we discuss preliminary experiments on how these effects may manifest in simulated negotiations, as well as design implications.

Keywords: negotiation · human-agent · fairness · preference elicitation

1 Introduction

As people increasingly employ A.I. in negotiation and dispute resolution tasks [17, 5], questions arise about the *fairness* of resulting agreements and even the potential that algorithmic solutions might correct some biases in human decision-making [10, 15, 16]. Prior work finds non-linearities in one's preference profile affect negotiated outcomes [13], and advances in A.I. make these methods increasingly sophisticated in accounting for the non-linear nature of human preferences. These have been long known to affect negotiated settlements [13]—including that people have conditional preferences across multiple issues (i.e., the level obtained on one issue might determine the relative importance of other issues) [3] and non-linearities within issues (e.g., people may assign diminishing marginal utility to money in a salary negotiation) [9]. Yet many A.I. methods use relatively simple, classical criteria to decide the fairness of potential agreements (e.g., taking elicited preferences at face value and calculating Pareto efficient solutions). Others attempt to impose rules on agents at a negotiation stage; Ouwerkerk found attempting to impose *fairness* via simple rules for a privileged negotiating agent can induce unintended consequences [14].

In contrast, work in psychology and behavioral economics has taken a more nuanced approach to assessing the fairness of economic solutions, highlighting that certain demographic or cultural groups may be placed at a systematic disadvantage by this approach to fairness; e.g., by being more risk averse, women may

express preferences that translate into worse monetary outcomes in salary negotiations. Yet these psychological approaches often adopt unrealistically simplistic assumptions about the complexity of human preferences (assuming preferences are independent across issues and linear within).

We explore how demographic and personality differences might shape the employment packages people obtain with an A.I. agent that attempts to find a fair and efficient contract. While replicating some common psychological findings (e.g., we find evidence that women and minorities would obtain lower salaries by using A.I. methods), these effects can be magnified when the A.I. uses non-linear preferences. Specifically, we find that risk-averse negotiators (who are disproportionately represented by women and certain racial groups) obtain a worse salary because they are more likely to express diminishing marginal utility for money. Additionally, we find those high in socio-economic status (SES) tend to use conditional preferences differently, systematically shaping their outcome.

We argue these effects call for a discussion on the meaning of *fairness*. For example, one interpretation of our results is that people are confounding the utility they assign to different salaries with a fear that they might not obtain an agreement if they express their true preferences. There is good reason to believe that certain groups, like women, are justified in these fears.

Prior research has suggested salary disparity in women might be linked to risk aversion – primarily through its influences on willingness to negotiate [7, 11]. Though less studied in the context of negotiations, prior work suggests that Asians have a greater propensity towards risk aversion as well [2]. Thus, we focus on how risk might impact elicited preferences in these two groups. We principally focus on the following research question:

- **RQ1:** How do demographic factors (e.g., gender and race) and individual differences (e.g., risk aversion socio-economic status) affect elicited preferences, proposed solutions of "fair" mediation approaches, and negotiation tasks?

Considering this research question, we make the following hypotheses:

- **H1:** Demographic characteristics will influence (a) risk aversion and (b) socio-economic status (SES).
- **H2:** Risk-averse participants express weaker within-issue preferences: i.e., curves over an individual issue.
- **H3:** SES will affect how participants express conditional preferences; e.g., different valuations of issues depending on contract length.

2 Experiment Setup

2.1 Methods

Participants: The subject pool consisted of **170** undergraduates from a west-coast university in the U.S. previously collected by us [6]. We removed participants for incomplete responses (14 removed), or if they completed the task in less

than five minutes (27 removed). The remaining **129 participants** had the following demographic breakdown: 64% male, 34% female, 2% other; self-reported race was 8% Hispanic, 46% Asian, 1% Pacific Islander, 4% Black, 33% White, 6% mixed-race, and 2% other; and 61% were born in the United States.

Design: In preparation for the elicitation task (via an online survey), we prompted participants to imagine they sought employment in a tech company; the survey presented them with a description of the company and "expert" (**bottom line**) reviews (see Figure 1 for a simplified example). Next, the text instructed them to input their preferences over three issues (*salary*, *vacation*, & *stock*), imagining they received the job to help finalize their employment offer. The three issues each had the following ten levels:

- **Salary:** $70k, $80k, $90k, $100k, $110k, $120k, $130k, $140k, $150k, & $160k
- **Vacation:** 5, 6, 7, 8, 9, 10, 11, 12, 13, & 14 days
- **Stocks:** $50k, $60k, $70k, $80k, $90k, $100k, $110k, $120k, $130k, & $140k

The experiment design manipulated contract length (1 and 5 years as a within variable), which allowed us to analyze if the length of the contract influenced preferences of the other issues[1].

Preparation: ***Requirements & bottom line***

[The company] values employees that

- Work independently in a fast-paced environment that gives you both great freedom and great responsibility

  ⋮
- Seek what is best for the company, rather than best for yourself or your group

Bottom line: This is not the place for employees seeking job security and a 9 to 5 workday, but for those confident in their skill and willing to put in the effort, the rewards can be substantial. Employees are heavily recruited by other technology companies.

Fig. 1: Example job description and bottom-line for the fictional company

Expounding on the elicitation task, participants input their preferences in a two-stage manner inspired by Thiessen & Soberg's dispute mediation platform: SmartSettle [17]. Users (1) express valuation of issues relative to each other via a slider wherein they must allocate all of 100 points among the three issues; then (2) they state their valuation of levels of each issue, wherein they draw a curve by dragging anchors for each of the 10 levels of an issue between 0 and 1. These measures make one's preference profile.

[1] We also manipulated the company's description to be *achievement* versus *family*-focused and found no significant differences. As such, we ignore this manipulation.

2.2 Measures:

Individual Differences Participants self-report demographic (e.g., race, gender, and whether born in the U.S.) and dispositional information prior to their task. Specifically, for dispositional measures, participants complete several short questionnaires. Participants complete the *MacArthur scale of Subjective Social Status* [1] to gauge their socio-economic status. Specifically, this shows participants an image of a ladder and prompts them to imagine higher rungs are those who are best off in society while bottom rungs are those worst off and to indicate the rung on which they stand. Next, to measure risk aversion, we scale by negative one the score attained by Meertens and Lion's *7-item Risk Propensity Scale*, which asks participants to what extent they agree with a series of statements (e.g., "I prefer to avoid risks") [12].

Preferences: As mentioned in Section 2.1, a participant's preference profile consists of the weight they assign to each issue (**Issue Weight**) as well as the curve they draw over the issue's levels (**Issue Curve**).

Toughness: We operationalize a *toughness* variable [6] to capture, in a single value, information about a participant's Issue Curve. This returns a higher value for curves with more points only at higher levels, and a low value for curves with more points only at lower levels. We formally define *toughness* as $f_{\text{curve}}(X) = \sum_{i \in \{1,\ldots,10\}} X_i \cdot i / \text{sum}(X)$, where X is the vector of elicited values for each level of a curve and X_i is the value of this issue at level i. Of note, we calculate *toughness* using a normalized curve where the max element is *one*.

Automatically Derived Outcomes (NBS): Here, we calculate a Nash Bargaining Solution (NBS) using the participant's full profile, calculating utility with a linear additive function. In these dyadic disputes, one participant's profile acts as a *worker* while the other operates as a *boss*. In lieu of elicited *boss* profiles, we take the Issue Curves assigned to the boss and flip them along the x-axis—e.g., their valuation for the first level now corresponds to the last level. Then, every *worker* profile matches with the *boss* version of every other profile in the mediation experiment. We consider the mean *worker* NBS of a profile in our analysis.

Simulated Negotiations: Using Hindriks *et al*'s GENIUS [8], a tool that allows for the simulation of negotiations, we analyze a negotiation context. We generally follow the same methodology as in the Automatically Derived Outcomes, however, the *worker* and *boss* negotiate with each other through the GENIUS platform. For a deal d, we consider the following:

- ***Integrativeness Quotient (IQ):*** $1 - \frac{S_d}{P}$, where S_d is the number of outcomes better than d for both sides and P is the number of possible deals; so, a higher value implies greater efficiency.
- ***Time:*** the total number of offers made before reaching an agreement.
- ***Salary level:*** *salary* level the *worker* receives on agreement with the *boss*.

3 Results

Do *Issue Weight* and *Toughness* explain outcome?: We first examine if the *Issue Weight* and the *Toughness* capture most of the variance in how people express their preferences. We perform three *linear* regression analyses—*toughness*, *weight*, and *weight + toughness*—on the three issues—*salary*, *vacation*, and *stock*—that use a user's preference profile to predict the corresponding average issue level when using NBS. Figure 2 shows prediction improves when using *toughness* and *weight* in tandem, suggesting *toughness* captures the effect of the curve on the NBS. Figure 2 illustrates the adjusted R^2 values for these regressions colored by issue type and grouped by the regression configuration. As these variables explain the variance of elicited preferences, it suffices to analyze how individual differences impact these parameters.

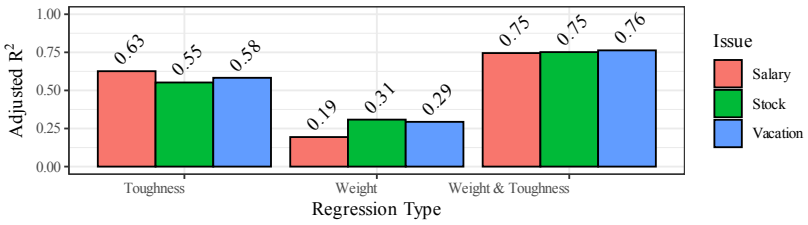

Fig. 2: Illustrates the adjusted R^2 values for regressions

Demographic Effects on Risk Aversion, SES, & Issues: Significant differences exist in risk aversion between demographic groups via two-tailed Welch's t-test (M is *Mean*, and SD is *Standard Deviation*) after testing normality with the Shapiro-Wilk test. We find women ($M = -30.02$) have significantly ($p = .041$) higher risk aversion than men ($M = -33.46$); and Asian people ($M = -30.00$) have significantly ($p = .013$) higher risk aversion than White people ($M = -34.81$). We do not find significant differences in SES between those same groups. **H1a** posits demographic characteristics affect risk aversion, **H1a is supported. H1b**, demography influencing SES, is not supported here.

Impact of Risk Aversion: Risk aversion, in the 5-year contract condition, trends to predict *salary* Issue Weight ($r = -.15$, $p = .088$), but does not strongly predict *stock* ($r = .10$, $p = .271$) or *vacation* ($r = .07$, $p = .399$). However, we see a greater effect of risk aversion on the elicited curves, where it significantly predicts *toughness* on *salary* ($r = -.23$, $p = .009$), *vacation* ($r = -.23$, $p = .008$), and *stock* ($r = -.22$, $p = .011$). **H2** posits those with greater risk aversion will post less tough preferences relating to salary. By these correlation tests, **H2 is supported.** We thus find support for the model in Figure 3.

Impact of Socio-economic Status: We perform a PCA decomposition on *stock* and *vacation* for 1 and 5-year contracts (we ignore *salary*, as two weights fully determine the other) and analyze correlations of each dimension with the

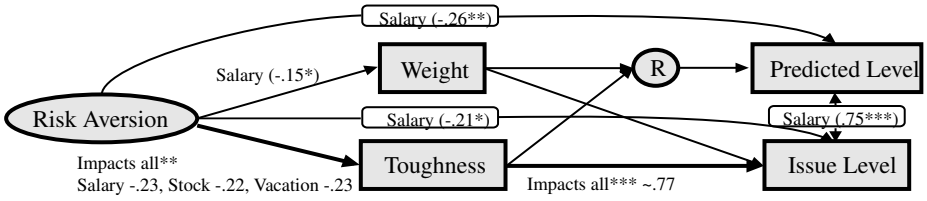

Fig. 3: Correlations (significance levels: * $p \leq .05$, ** $p \leq .01$, & *** $p \leq .001$)

SES score. The third PCA dimension—$(0.37 \cdot \text{Vac}_1 - .40 \cdot \text{Sto}_1) + (.34 \cdot \text{Sto}_5 - .41 \cdot \text{Vac}_5)$—significantly positively correlates with SES ($r = .31$, $p < .001$). Thus, we find high SES people trade off weights differently across contracts; specifically, they tend to trade weight from vacation in the 1-year setting to stock for the 5-year contract. **H3 is supported**.

Simulated Negotiations: Following the methodology outlined in Section 2.2, we investigate a negotiation context. Again, we only focus on the 5-year condition for brevity; first, we find a significant ($p < .001$) positive ($r = .48$) correlation between the settled salary issue levels generated between the NBS and Genius solution. Notably, in a few instances where NBS proposed a low salary level (≤ 2), GENIUS on average proposed a higher level (> 6). Further, unlike in NBS, negotiation does not guarantee Pareto efficiency; in fact, we find the average ***I.Q.*** ($M = .79$, $SD = .21$) does not lie close to a perfect 1, meaning this produces many inefficient solutions (this may be mitigated by increasing the max turns two agents can take). Lastly, mediated solutions do not require back-and-forth between disputants, but negotiations do; we find, on average, the simulations play-out $M = 13.09$ turns before agreement ($SD = 3.23$).

4 Discussion and Future Work

Design Implications: Our work posits a potential pathway to inequity through demographic and dispositional characteristics. A designer of mediation and negotiation systems must consider whether an A.I. mediator should take elicited preferences at face value, accepting economic inequities simply reflect systematic differences in the utility of money across different groups, or if these differences in expressed utility arise through structural biases in the populations we studied.

Future Work: We intend to examine other aspects of perceived fairness in negotiations. E.g., exploring the fact that people tend to be less satisfied when their counterpart sends a "take it or leave it" offer or accepts an offer immediately [4]. While this represents the most efficient type of negotiation, prior work suggests people find such approaches quite unsatisfying; i.e., they tend to reject and feel dissatisfaction with offers in such scenarios, compared to the same offer in a different context. Algorithmic approaches to negotiation may provide greater opportunities for these kinds of efficient outcomes, which raises the question of whether such dissatisfaction would persist when using an A.I. proxy.

References

1. Adler, N.E., Epel, E.S., Castellazzo, G., Ickovics, J.R.: Relationship of subjective and objective social status with psychological and physiological functioning: Preliminary data in healthy, white women. Health psychology **19**(6), 586 (2000)
2. Bontempo, R.N., Bottom, W.P., Weber, E.U.: Cross-cultural differences in risk perception: A model-based approach. Risk analysis **17**(4), 479–488 (1997)
3. Boutilier, C., Brafman, R.I., Domshlak, C., Hoos, H.H., Poole, D.: Cp-nets: A tool for representing and reasoning withconditional ceteris paribus preference statements. Journal of artificial intelligence research **21**, 135–191 (2004)
4. Galinsky, A.D., Seiden, V.L., Kim, P.H., Medvec, V.H.: The dissatisfaction of having your first offer accepted: The role of counterfactual thinking in negotiations. Personality and Social Psychology Bulletin **28**(2), 271–283 (2002)
5. Goldman, J., Procaccia, A.D.: Spliddit: unleashing fair division algorithms. SIGecom Exch. **13**(2), 41–46 (2015)
6. Hale, J., Kim, P., Gratch, J.: Preference interdependencies in a multi-issue salary negotiation. In: Proceedings of the 22nd ACM International Conference on Intelligent Virtual Agents. IVA '22, Association for Computing Machinery, New York, NY, USA (2022)
7. Hernandez-Arenaz, I., Iriberri, N.: A review of gender differences in negotiation. Oxford Research Encyclopedia of Economics and Finance (2019)
8. Hindriks, K., Jonker, C.M., Kraus, S., Lin, R., Tykhonov, D.: Genius: negotiation environment for heterogeneous agents. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2. pp. 1397–1398 (2009)
9. Kahneman, D., Tversky, A.: Prospect theory: An analysis of decision under risk. In: Handbook of the fundamentals of financial decision making: Part I, pp. 99–127. World Scientific (2013)
10. Lee, M.K., Baykal, S.: Algorithmic mediation in group decisions: Fairness perceptions of algorithmically mediated vs. discussion-based social division. In: Proceedings of the 2017 acm conference on computer supported cooperative work and social computing. pp. 1035–1048 (2017)
11. Marks, M., Harold, C.: Who asks and who receives in salary negotiation. Journal of Organizational Behavior **32**(3), 371–394 (2011)
12. Meertens, R.M., Lion, R.: Measuring an individual's tendency to take risks: The risk propensity scale1. Journal of Applied Social Psychology **38**(6), 1506–1520 (2008). https://doi.org/https://doi.org/10.1111/j.1559-1816.2008.00357.x
13. Northcraft, G.B., Preston, J.N., Neale, M.A., Kim, P.H., Thomas-Hunt, M.C.: Nonlinear preference functions and negotiated outcomes. Organizational Behavior and Human Decision Processes **73**(1), 54–75 (1998)
14. Ouwerkerk, N.: The unintended consequences fairness brings to automated negotiation (2022)
15. Sela, A.: Can computers be fair: how automated and human-powered online dispute resolution affect procedural justice in mediation and arbitration. Ohio St. J. on Disp. Resol. **33**, 91 (2018)
16. Sternlight, J.R.: Pouring a little psychological cold water on online dispute resolution. J. Disp. Resol. p. 1 (2020)
17. Thiessen, E.M., Soberg, A.: settle described with the montreal taxonomy. Group Decision and Negotiation **12**(2), 165 (2003)