

Facilitate Communication

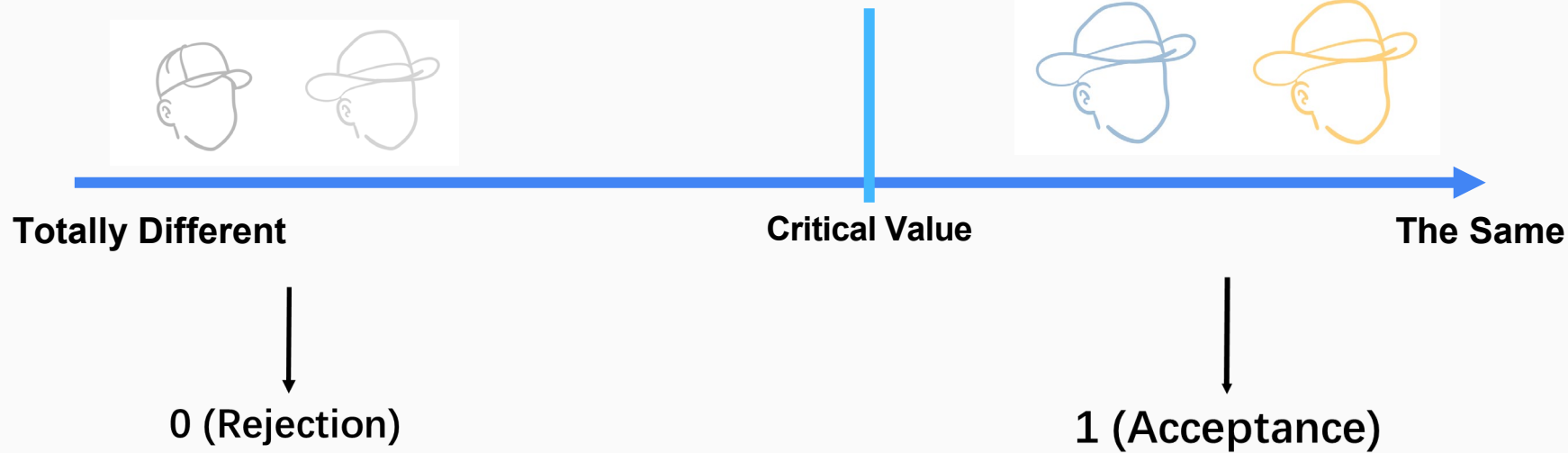
A mechanism for increasing intergroup contact.

Thomas Yiliang Ge 12/11/2024



Context

- Intergroup Contact provides an opportunity to gain new information and reduce intergroup prejudice.
- To increase the communication intention, requires:
 - a) For any group, there should be at least **one Type A** agent and at least **one Type B** agent.
 - b) The utility function of social planner satisfies: $U(S) = \min\{m_i\}$, where m_i represents the similarity degree of agent in group i . Find $\max\{U(S)\}$.



High similarity $\overset{?}{\Rightarrow}$ High Intention

		Type B	
		<i>Initiate</i>	<i>Respond</i>
Type A	<i>Initiate</i>	(u_1, u_1)	(u_2, u_2)
	<i>Respond</i>	(u_2, u_2)	(u_3, u_3)

$$u_1 = Rh(1 - Rh)^{1-\gamma} - \lambda(1 - h)h^{1-\gamma}$$

$$u_2 = h(1 - h)^{1-\gamma} - \lambda(1 - h)h^{1-\gamma}$$

$$u_3 = 0$$

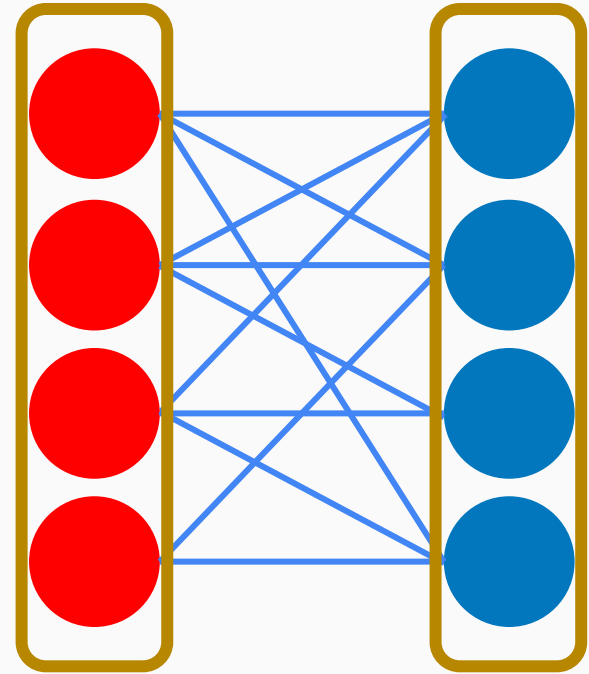
$$\begin{cases} h > \frac{\lambda}{\lambda^\gamma + \lambda} \\ h < \frac{R - R^{1-\gamma}}{R^{2-\gamma} - R^{1-\gamma}} \end{cases}$$

- Agents who are too different are more likely to suffer in communication.
- Agents who are too similar worry about unexpected losses.
- Agents who are too similar or too different from each other are not easy to communicate.



Bipartite Graph Matching

- Construct a bipartite graph: one set of vertices being A Type, the other set of vertices being B Type. Build edge between two vertices if they have an acceptable matching score.
- Perform perfect matching



Red: Type A
Purple: Type B

Bipartite Graph Matching

Input: Bipartite Graph $G = \langle L, R, E \rangle$

Output: The matching array *matched*[]

Hungarian(G) :

create Array *matched*[], *visited*[] with size $|R|$

matched[u] \leftarrow NULL for every $u \in R$

for $v \in L$ **do**:

visited[u] \leftarrow *False* for every $u \in R$

DFS(G, v)

end for

return *matched*

Input: Bipartite Graph $G = \langle L, R, E \rangle, v$

Output: Whether an alternating path starting from v exists

DFS(G, v) :

for $u \in \text{Neighbor}(v)$ **do**:

if *visited*[u] **then** continue

visited[u] \leftarrow *True*

if *matched*[u] = NULL or *DFS*($G, \text{matched}[u]$) **then**

matched[u] \leftarrow v

return *True*

end if

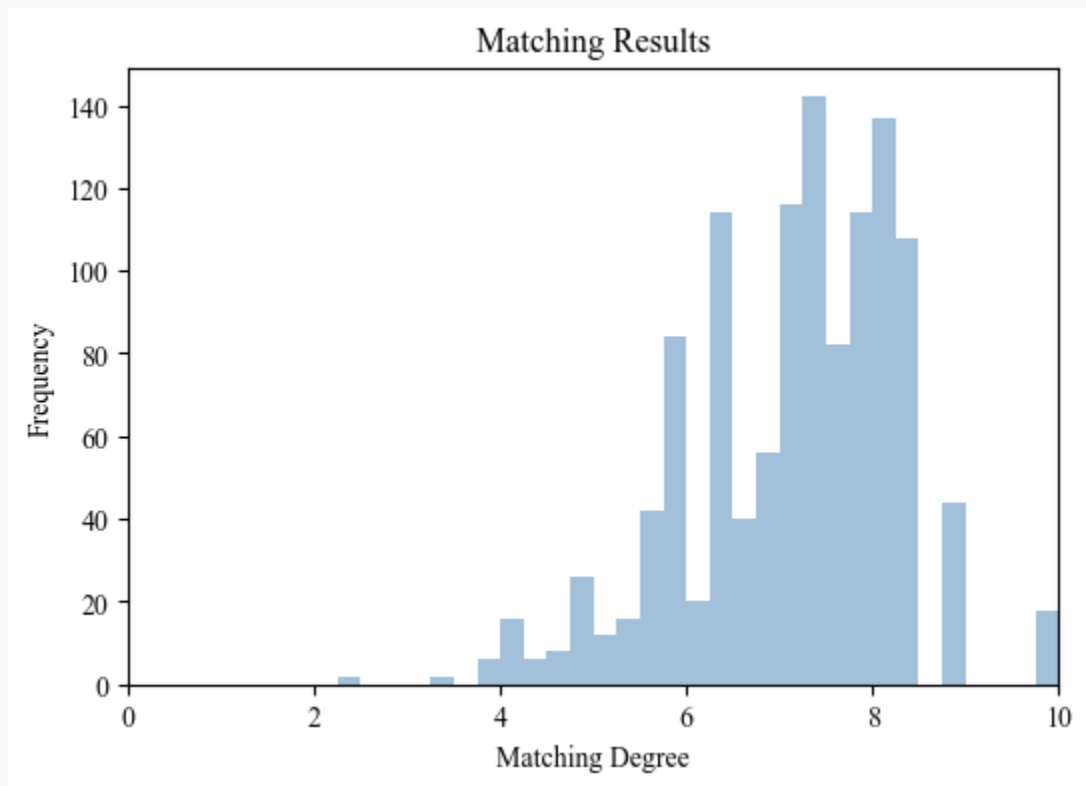
end for

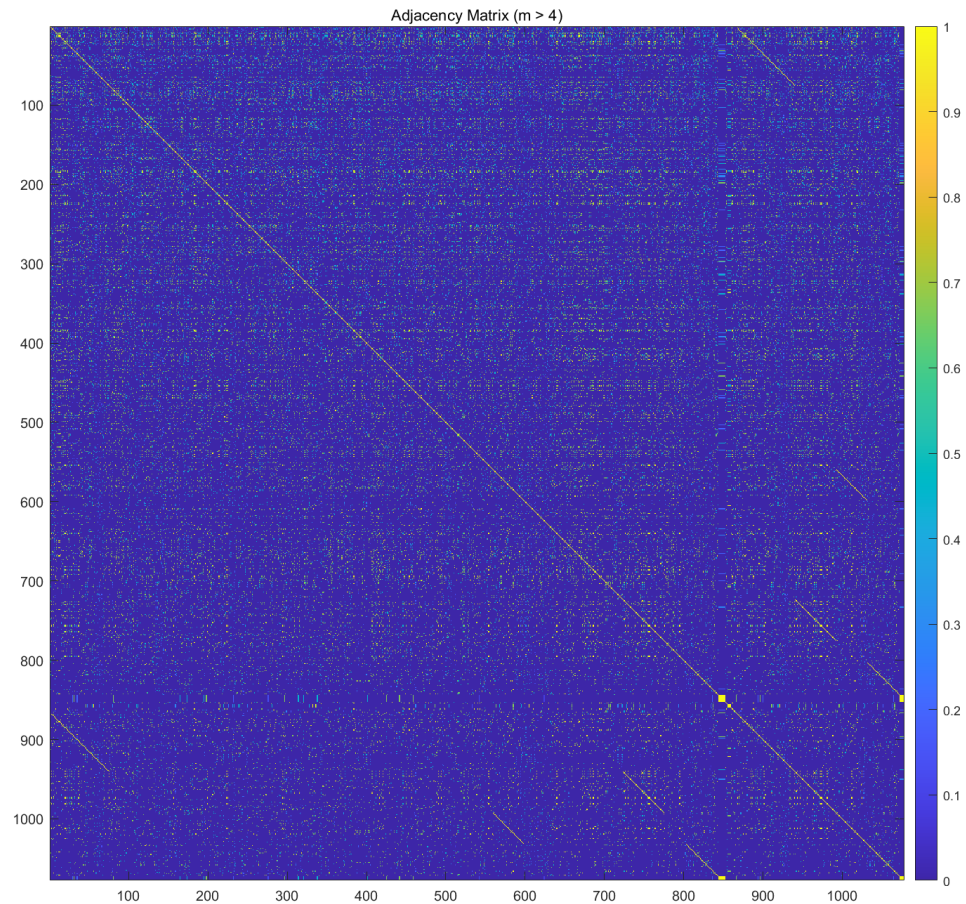
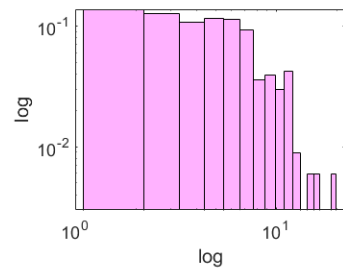
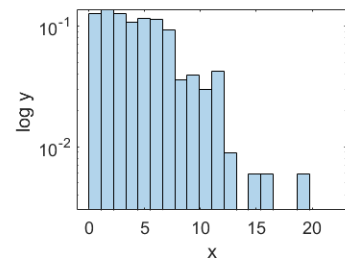
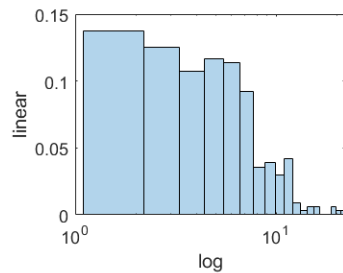
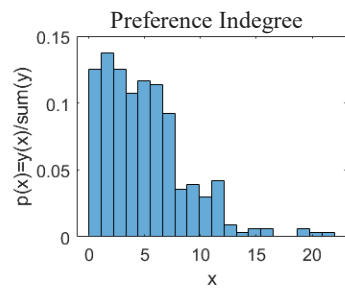
return *False*

Matching Degree

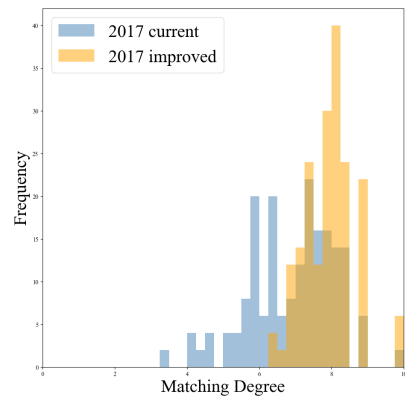
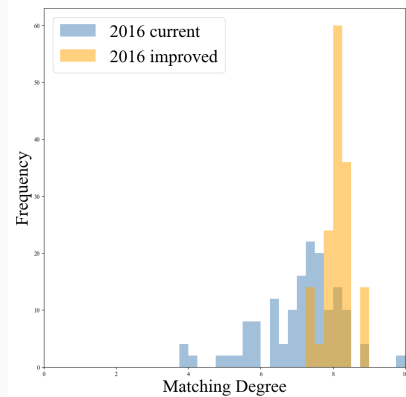
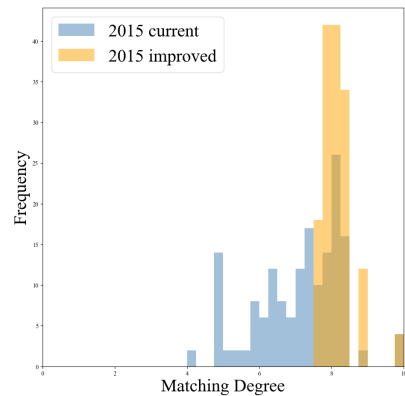
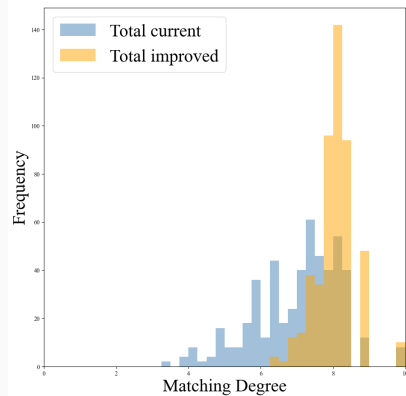
Student	Gender	Nationality	Term Session	Roommate	Atmosphere	Bedtime	Cleanliness	Outgoing	Wake Up
211	Female	International	Fall 2017	229	2	2	0	2	2
229	Female	Chinese	Fall 2017	211	-2	-1	-1	0	0
258	Female	International	Fall 2017	306	-1	-1	-1	1	0
306	Female	Chinese	Fall 2017	258	1	1	0	1	0

$$m_{i,j} = 10 - \frac{\sqrt{5}}{2} \|l_i - l_j\|_2$$

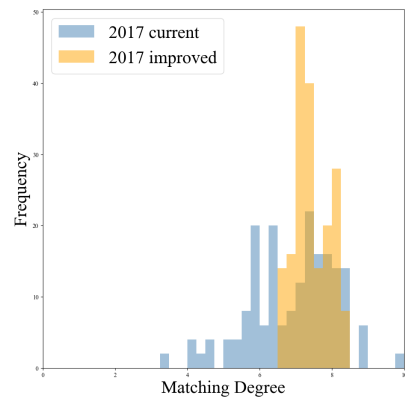
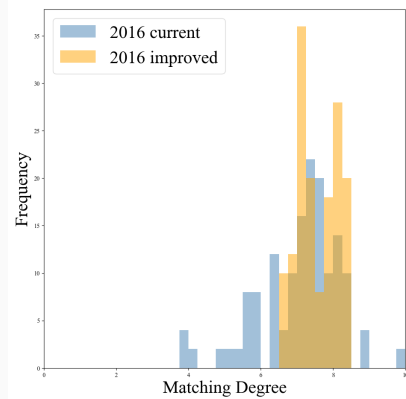
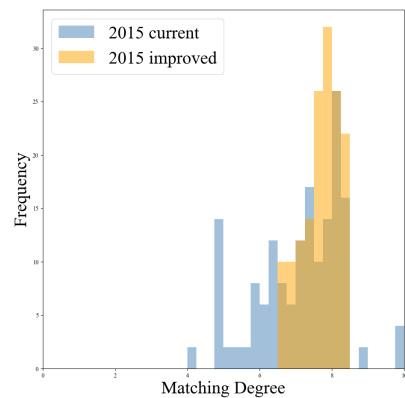
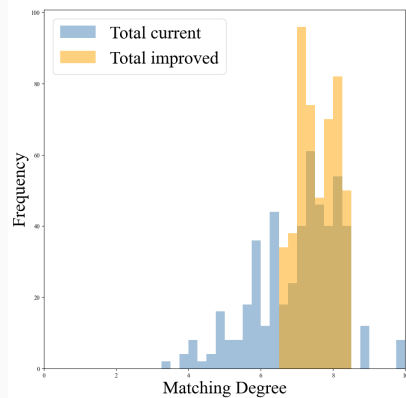




Current Algorithm v.s. Improved Algorithm



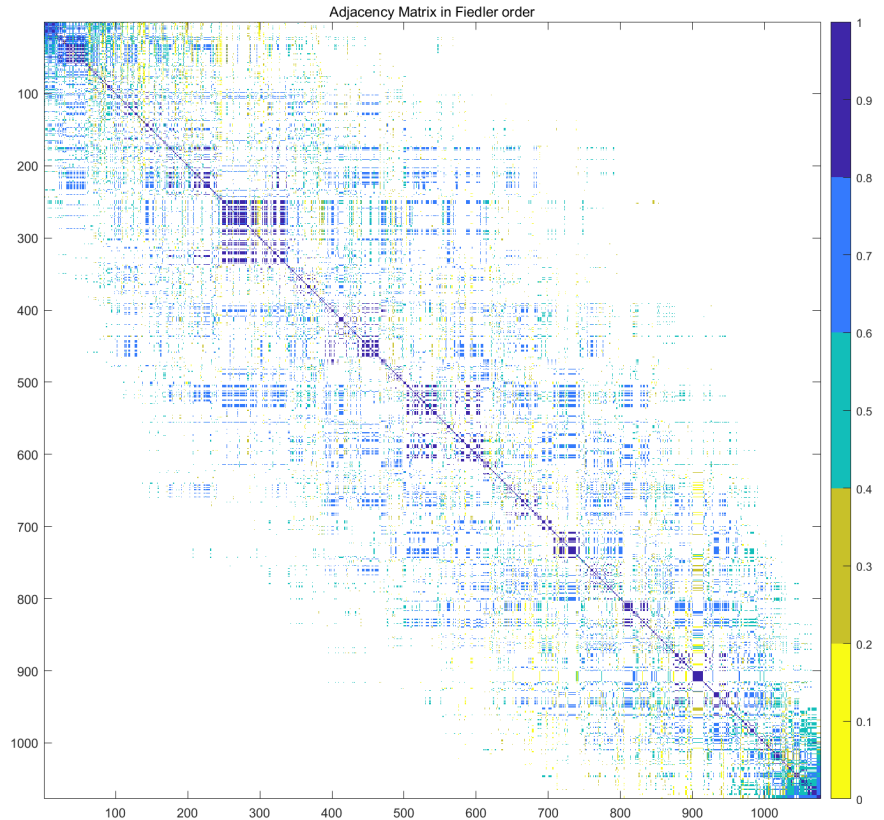
Current Algorithm v.s. Improved Algorithm



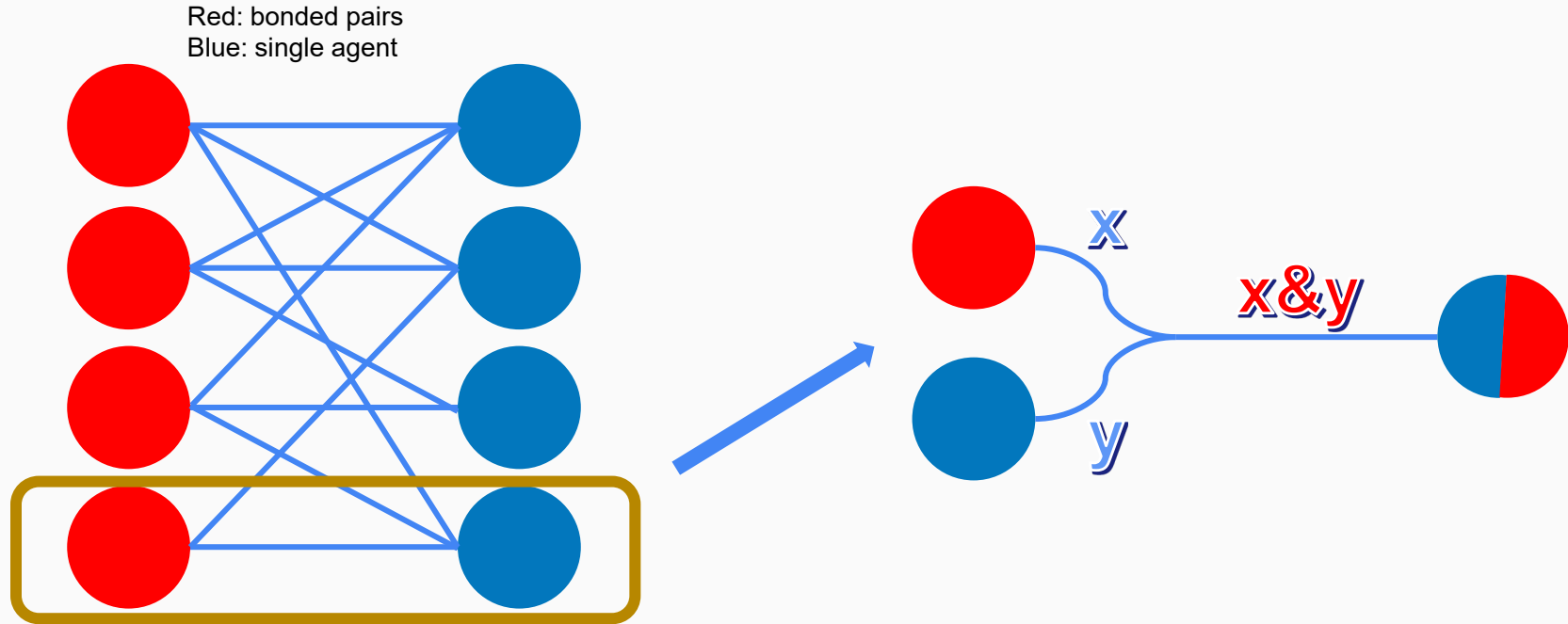
Discussion

Difficulties:

1. hard to satisfy in groups 3 agent
2. difficult to define a proper way to measure the matching degree



Potential Approach: Bounded Pair



Questions
are welcome

