Human-Agent Collaboration for Disaster Response[1]

Yi Xia

November 1st, 2018

Ramchum, S. D., F. Wu, W. Jiang,. J. E. Fischer, S. Reece, S. Roberts, T. Rodden, C. Greenhalgh, and N. R. Jennings (2015). Human-Agent Collaboration for Disaster Response, *Journal of Autonomous Agents and Multiagent Systems*, 30(1):82-111

Outline

- Introduction
- The Disaster Scenario
- Team Coordination Algorithm
- AtomicOrchid Platform
- Field Trials and Results
- Conclusions

Outline

- **Introduction**
  - Human team coordination for disaster response
  - Agent-based planning
  - Challenges for human-agent collaboration
  - Disaster simulation & games
  - Decision-theoretic multi-agent planning
- …

Disaster Response

- Problem Formulation
  - First responders
  - Search and rescue task
  - Dynamic and uncertain environment
  - Resource and task allocation
- Human Team Coordination
  - Interdependencies between activities [2]
  - Spatially distributed incidents & resources [3]
  - Optimize coordination to minimize failures [4]

Disaster Response

- Challenges that align with ad hoc teamwork
  - Create a shared understanding [5]
  - Develop situation awareness [6]
  - Align cooperative action through on-going communication [4]

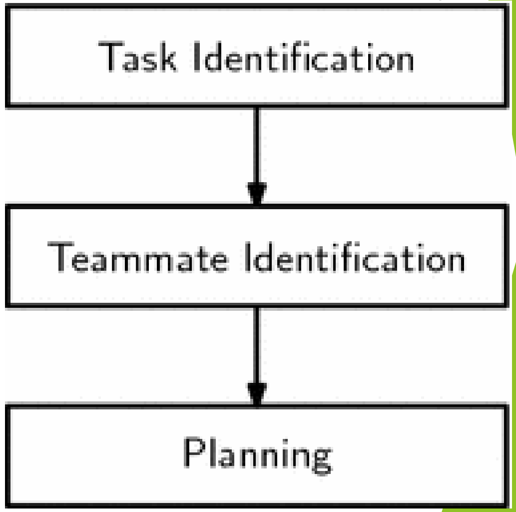

Figure 1

Agent-Based Planning

- Decentralized coordination [7, 8]
  - Decision based on local knowledge
- **Coordination by a central authority** [9]
  - Complete knowledge of the system
- Coalition formation [10]
  - Synergy based task completion

Challenges for Human-Agent Collaboration Planning

- Transfer-of-control policies [11]
- Evaluate strategies of agent support [12]
- No real world studies
- Overlooked environmental dynamics
- Ignored subtleties of human interactions and perception
- No worst case or average case of simulations

Disaster Simulation & Games

- Agent based computational simulation
- Considering human psychosocial characteristics
- Mixed-reality game approach
  - Situated in the real world
  - Captures realistic cognitive and physical stress
  - Behavioral observation

Multi-Agent MDP (MMDP)

- Sequential decision making
- Dependence between task completion
- Monte-Carlo Tree Search for online planning

Outline

- …
- **The Disaster Scenario**
  - Real world model
  - Computational model
  - Human-agent collaboration
  - The optimization problem
- …

Real World Disaster Model

- Consultations with emergency response organizations [a,b]
- Design of decision making challenges
  - Hazard avoidance
  - Path planning
  - Team coordination
- Disaster Scenario
  - Radioactive fueled satellite crash into a sub-urban area
  - Debris damaging building and threatening civilians
  - Radioactive cloud contamination
  - Emergency services to search and rescue

a. http://www.rescueglobal.org.
b. http://www3.hants.gov.uk/emergencyplanning.htm.

Computational Model

- G : a grid overlaid the disaster space
- $(x, y) \in G$: coordinates for objects and human on the grid
- $l \in [0,100]$: radioactivity level induced by the radioactive cloud
- $G' \in G$: safe zones to drop off assets and casualties
- $I = \{p_1, \dots, p_i, \dots, p_n\}$: set of FRs
- $T = \{t_1, \dots, t_i, \dots, t_m\}$: set of tasks
- $h_i \in [0, 100]$: health level for FR_i, decreased by $0.02 \times l$ per second
- $\theta_i \in \Theta$: type of responder p_i, which determines the capabilities
- C_j : set of responders that can complete task t_j iff $\Theta_{t_j} \subseteq \{\theta_i | p_i \in C_j\}$

Human-Agent Collaboration

- FRs are coordinated from a headquarters (HQ)
- HQ is headed by a human coordinator H
- H is assisted by an agent-based planning agent PA
- PA receive input from, and direct FRs
- PA handles planning, task rejections and re-planning
- PA gives instructions directly to FRs
- H has the overriding power at any point

The Optimization Problem

MMDP with uncertainties $\mathcal{M} = \langle I, S, \{A_i\}, P, R \rangle$

- $S = S_r^G \times S_{p_1} \times \cdots \times S_{p_n} \times S_{t_1} \times \cdots \times S_{t_m}$
  - State of the radioactive cloud: $S_r^G = \{l_{(x,y)} | (x,y) \in G\}$
  - State for each responder's health level, position and current task: $S_{p_i} = \langle h_i, (x_i, y_i), t_j \rangle$
  - State for each task's status and position: $S_{t_j} = \langle \mathtt{st_j}, (x_j, y_j) \rangle$
- $P = P_r \times P_{p_1} \times \cdots \times P_{p_n} \times P_{t_1} \times \cdots \times P_{t_m}$
  - Probability of the radioactive cloud spread: $P_r(s'_r | s_r)$
  - Probability of responder's action transition: $P_{p_i}(s'_{p_i} | s, a_i)$
  - Probability of task being completed: $P_{t_j}(s'_{t_j} | s, \mathbf{a})$
- Reward, penalties and cost.

The Optimization Problem

$$V^{\pi}(s^{\tau}) = R(s^{\tau}, \pi(s^{\tau})) + \sum_{s^{\tau+1} \in S} P(s^{\tau+1} | s^{\tau}, \pi(s^{t})) V^{\pi}(s^{\tau+1})$$

- A policy for the MMDP is a mapping from states to joint action
- Quality of a policy is measured by expected value
- Goal is to find the optimal policy that maximizes the expected value with initial state
  $\pi^{*} = \arg\max_{\pi} V^{\pi}(s^{0})$

Outline

- …
- **Team Coordination Algorithm**
  - Computational threshold
  - Task planning
  - Path Planning
- …

Computational Threshold

- MMDP often results a very large search space
  - 8 responders, 17 tasks, 50 x 55 grid: more than 2×10^{400} states
- Approximate optimal solution
  - Top level: form teams
  - Lower level: find paths to tasks
  - Online planning for reachable states only
    - 9^8 states for 8 responders vs. $(50 \times 55)^8$

Team Coordination Algorithm

Algorithm 1: Team Coordination Algorithm

Input: the MMDP model and the current state s.
Output: the best joint action $\mathbf{a}$.

//The task planning
$\{t^i\} \leftarrow$ compute the best task for each responder $p_i \in I$;
foreach $p_i \in I$ **do**
    //The path planning
    $a_i \leftarrow$ compute the best path to task t^i ;
return a

Task Planning

- Each FR p_i of a specific type θ_i defined the task she can perform
- Each task t requires a set of types Θ_t
- If p_i is incapable of performing a task, then $I \rightarrow I \setminus p_i$
- If a task is complete, then $T \rightarrow T \setminus t_k$
- Define the value of a team as $v(C_{jk})$
- Goal: task assignment that maximizes all team values $\sum_{j=1}^{m} v(C_j)$

Team Value Calculation [13]

Algorithm 2: Team Value Calculation

Input: the current state s, a set of unfinished tasks T, and a set of free FRs I.
Output: a task assignment for all FRs.

$\{C_{jk}\} \leftarrow$ compute all possible teams of I for T ;
foreach $C_{jk} \in \{C_{jk}\}$ **do**
  //The N trial simulations
  for $i = 1$ *to* N **do**
    $(r, s') \leftarrow$ run the simulation starting at state s until task k is completed by the FRs in C_{jk} ;
    if s' *is a terminal state* **then**
      $v_i(C_{jk}) \leftarrow r$;
    else
      $V(s') \leftarrow$ estimate the value of s' with MCTS ;
      $v_i(C_{jk}) \leftarrow r + \gamma V(s')$;
  $v(\bar{C}_{jk}) \leftarrow \frac{1}{N} \sum_{i=1}^{N} v_i(C_{jk})$;
return the task assignment computed by Eq. 3.

$$v(C_{jk}) = \overline{v(C_{jk})} + c\sqrt{\frac{2N(s)}{N(s, C_{jk})}}$$

Coordinated Task Allocation

$$
\begin{array}{lll}
\max\limits_{x_{jk}} & \sum_{j,k} x_{jk} \cdot v(C_{jk}) & \\
\text{s.t.} & x_{jk} \in \{0,1\} & \\
 & \forall j, \sum_k x_{jk} \leq 1 & \text{(i)} \\
 & \forall i, \sum_{j,k} \delta_i(C_{jk}) \leq 1 & \text{(ii)}
\end{array}
$$

- Each task is assigned to at most one team
- Each FR is assigned to only one task
- Standard Mixed Integer Linear Program (MILP) that can be solved using off-the-shelf solvers
- Easily adaptable to FR's requests

Path Planning

- Single-agent MDP
- State space only involves radioactive cloud and FR’s status
- Transition function only considers radioactive cloud spreading and changes of status
- Reward function considers cost of moving and radioactive exposure
- Solved using Real-Time Dynamic Programming (RTDP) [14]

Outline

- …
- **AtomicOrchid Platform**
  - Interfaces
  - Architecture
  - Demonstration
- …

AtomicOrchid Platform

- A location based mobile game based on the disaster scenario
- FRs: medic, fire-fighter, soldier and transporter
- Targets
  - Victim – requires medic and fire-fighter
  - Animal – requires medic and transporter
  - Fuel – requires soldier and fire-fighter
  - Other resources – requires soldier and transporter

AtomicOrchid Player Interface

Figure 2 Mobile responder tool

Integrating the PA

- PA takes the tame status as input and generate plans for each FR
- The AtomicOrchid requests plans from PA, located on a separate server
- Re-planning is triggered by:
  - Completion of task
  - Explicit rejection from a FR

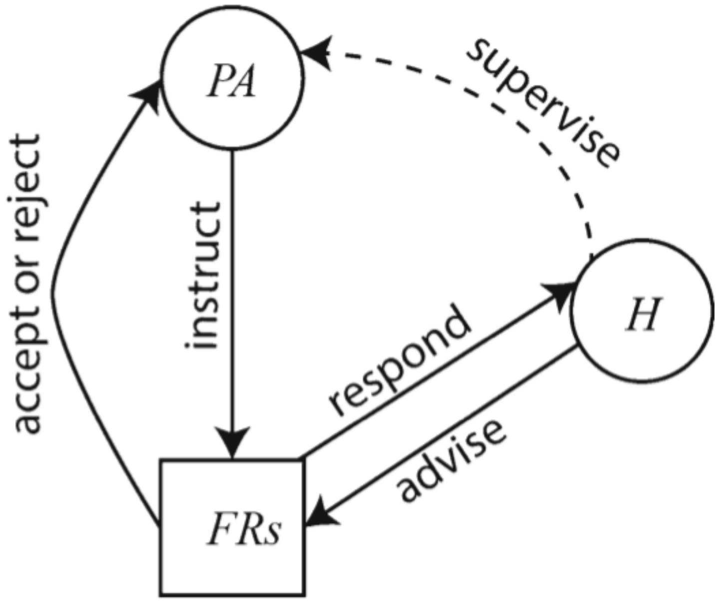

Figure 3 Interactions between PA, FR, and H

Demonstration

https://www.youtube.com/watch?v=1U3ENY6KhWY

Outline

- …
- **Field Trials and Results**
  - Game settings
  - Results
- **Conclusions**

Game Setting

- 3 trials with 8 people each [15]
  - £15 for 1.5-2 h of study
  - Majority were students
- One session (A) without PA, two sessions (B,C) with PA
- 400 x 400 meter game area
- 2 safe zones and 16 targets in each session
- Each session is 30 minutes
- Data collection:
  - 20 h of video of participants
  - Transcription and triangulation
  - Most relevant types of acts:
    - Assertives speech acts for truth of proposition
    - Directives speech acts for action requirement

Game Results

One session (A) without PA, two sessions (B,C) with PA

- A is less efficient in traces while B, C have more coverage
- 8 targets rescued in A, 12 in B and 11 in C
- 14 re-planning in B and 18 in C

Game Results

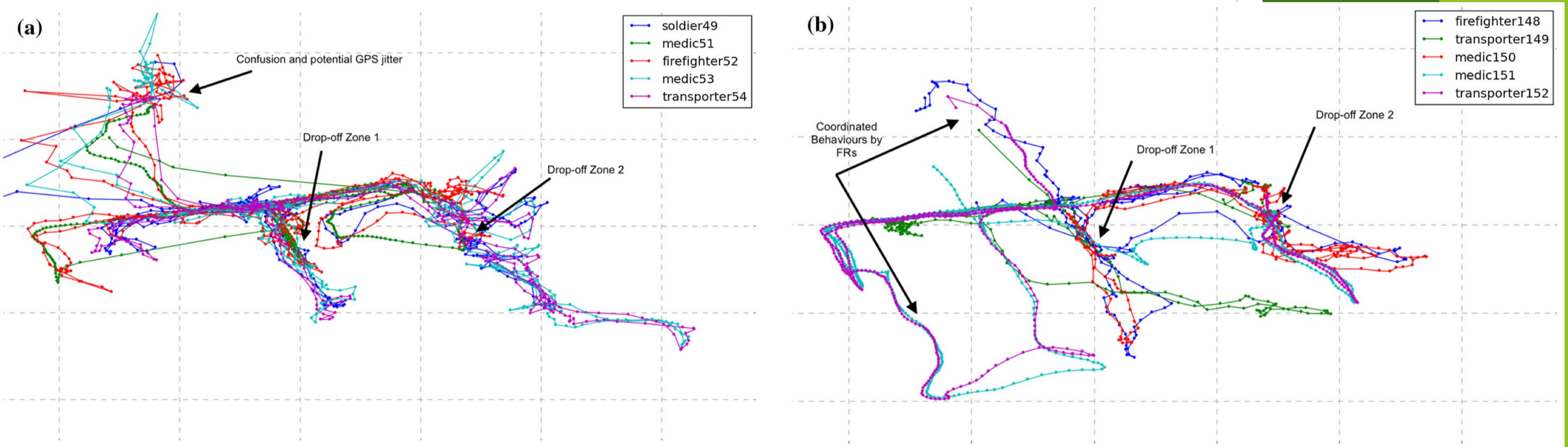

Figure 4 Traces of player movements during two games. **a** Players are seen to herd to different tasks with no clear coordination. User ids in the game are shown in the legend. **b** Players are seen to pair up with different partners at different times indicating good coordination

Game Results

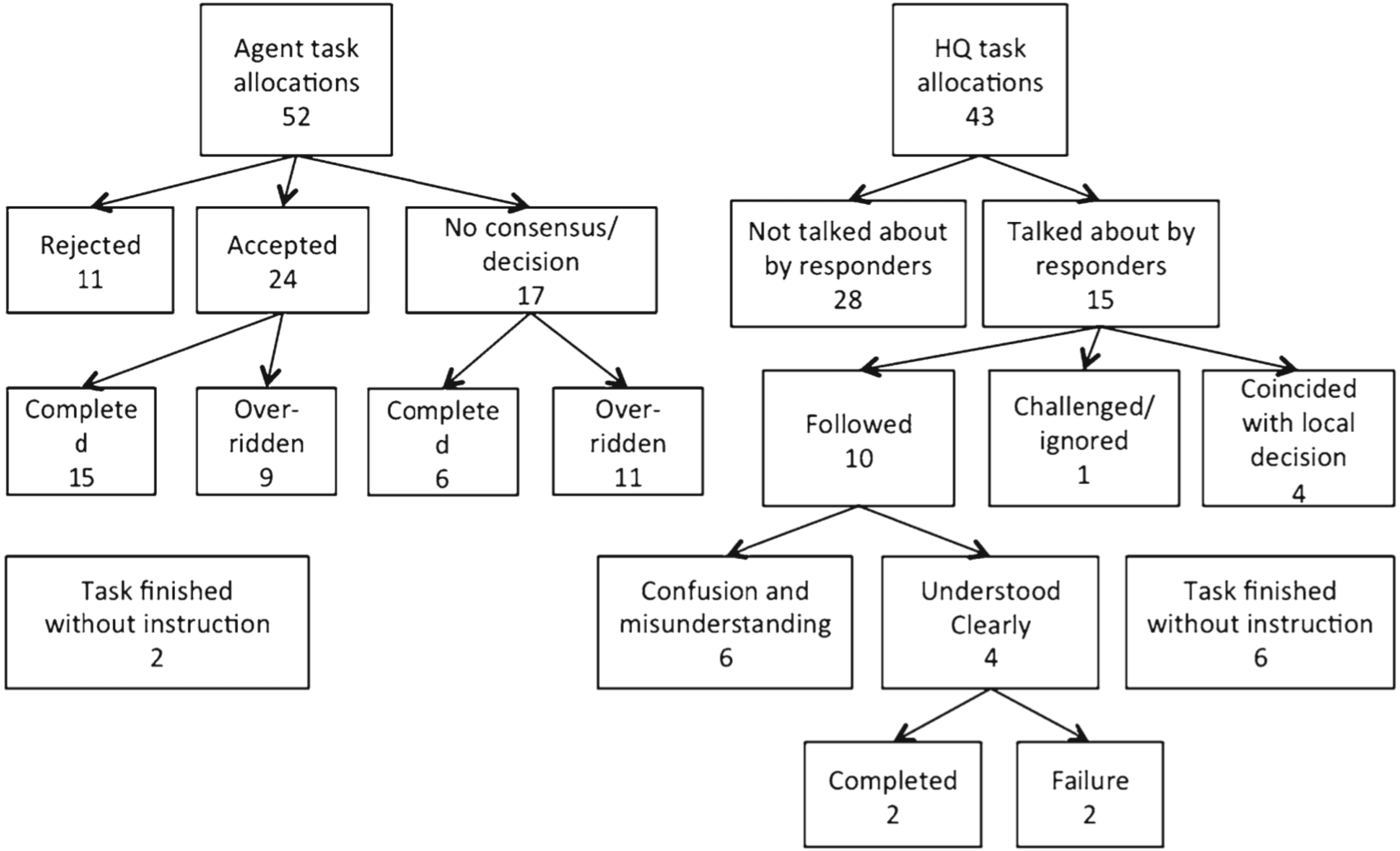

Figure 5 How task allocations were handled by FRs in the version with agent (left), and without agent (right). The 'overridden' box denotes the number of times an 'accepted' allocation in the non-agent case was not carried out to its full extent and, instead, another task was chosen by H to be completed

Game Results

Table 1 Health of FRs.

	Minimum health	Maximun health	Average health	Standard deviation
Session A	0	95	40	26.95
Session B	64	100	91	13.41
Session C	41	99	72	24.99

Table 2 Message classification

Speech acts	No agent		Agent				Total
	Session A		Session B		Session C		
	HQ	FR	HQ	FR	HQ	FR	
Directives	89	0	34	2	34	0	159
Assertives	33	6	26	16	24	16	121
Total	122	6	60	18	58	16	280

Game Results

- FRs performed better and maintained higher health levels when supported by the agent.
- Fast re-planning helps more tasks to be completed.
- H and PA demonstrates effective division of labor.

Paper Conclusion

- Developed a novel planning agent (using an MMDP approach)
- Planning agent can quickly re-plan based on FR's preferences and constraints
- Provided design guidelines for human-agent collaboration in human-agent collectives:
  - Adaptivity
  - Interaction simplicity
  - Flexible autonomy

My Conclusion

- After-work assessment based on FR performance to generate initial states for next scenario
- Uncertainty in centralized human coordinator
- Task assignment enforcement based on FRs current states
  - Reject to rejections
- Task priorities when defining reward function
- If team is rarely selected in previous visits, why should it have higher chance of being selected?

$$v(C_{jk}) = \overline{v(C_{jk})} + c\sqrt{\frac{2N(s)}{N(s, C_{jk})}}$$

Thank you!
Q & A

Reference

[1] Ramchurn, S. D., F. Wu, W. Jiang,. J. E. Fischer, S. Reece, S. Roberts, T. Rodden, C. Greenhalgh, and N. R. Jennings (2015). Human-Agent Collaboration for Disaster Response, *Journal of Autonomous Agents and Multiagent Systems*, 30(1):82-111

[2] Malone, T. W. & Crowston, K. (1990). What is coordination theory and how can it help design cooperative work systems? In *Proceedings of the 1990 ACM conference on computer-supported cooperative work—CSCW '90* (pp. 357–370). New York, NY: ACM.

[3] Chen, R., Sharman, R., Rao, H. R., & Upadhyaya, S. J. (2005). Design principles of coordinated multi-incident emergency response systems. *Simulation*, *3495*, 177–202.

[4] Toups, Z. O., Kerne, A., & Hamilton, W. A. (2011). The team coordination game: Zero-fidelity simulation abstracted from fire emergency response practice. *ACM Transactions on Computer–Human Interaction*, *18*(4), 1–37.

[5] Convertino, G., Mentis, H. M., Slavkovic, A., Rosson, M. B., & Carroll, J. M. (2011). Supporting common ground and awareness in emergency management planning. *ACM Transactions on Computer–Human Interaction*, *18*(4), 1–34.

[6] Bader, T., Meissner, A., & Tscherney, R. (2008). Digital map table with fovea-tablett ®®: Smart furniture for emergency operation centers. In *proceedings of the 5th international conference on information systems for crisis response and management* (pp. 679–688).

Reference

[7] Chapman, A., Micillo, R. A., Kota, R., & Jennings, N. R. (2009). Decentralised dynamic task allocation: A practical game-theoretic approach. *Proceedings of AAMAS, 2009*, 915–922.

[8] Pujol-Gonzalez, M., Cerquides, J., Farinelli, A., Meseguer, P., & Rodríguez-Aguilar, J. A. (2014). Binary max-sum for multi-team task allocation in robocup rescue. In *Optimisation in multi-agent systems and distributed constraint reasoning (OptMAS-DCR)*, Paris, France, 05/05/2014.

[9] Khan, M. A., Turgut, D., & Bölöni, L. (2011). Optimizing coalition formation for tasks with dynamically evolving rewards and nondeterministic action effects. *Journal of Autonomous Agents and Multi-agent Systems*, *22*(3), 415–438.

[10] Ramchurn, S. D., Farinelli, A., Macarthur, K. S., & Jennings, N. R. (2010). Decentralized coordination in robocup rescue. *The Computer Journal*, *53*(9), 1447–1461.

[11] Scerri, P., Tambe, M., & Pynadath, D. V. (2002). Towards adjustable autonomy for the real-world. *Journal of Artificial Intelligence Research*, *17*(1), 171–228.

[12] Lenox, T. L., Payne, T., Hahn, S., Lewis, M., & Sycara, K. (2000). Agent-based aiding for individual and team planning tasks. *Proceedings of the human factors and Ergonomics Society Annual Meeting*, *44*(1), 65–68.

Reference

[13] Kocsis, L., & Szepesvári, C. (2006). Bandit based Monte-Carlo planning. *Proceedings of ECML*, *2006*, 282–293.

[14] Barto, A. G., Bradtke, S. J., & Singh, S. P. (1995). Learning to act using real-time dynamic programming. *Artificial Intelligence*, *72*(1), 81–138.

[15] Brown, B., Reeves, S., & Sherwood, S. (2011). Into the wild: Challenges and opportunities for field trial methods. In *Proceedings of the SIGCHI conference on human factors in computing systems, CHI '11* (pp. 1657–1666). New York, NY: ACM.