Fischer, Joel E. and Greenhalgh, Chris and Jiang, Wenchao and Ramchurn, Sarvapali D. and Wu, Feng and Rodden, Tom (2016) In-the-loop or on-the-loop? Interactional arrangements to support team coordination with a planning agent. Concurrency and Computation: Practice and Experience . ISSN 1532-0634 (In Press)

CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. 2016; **00**:1–23
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cpe

In-the-loop or on-the-loop? Interactional Arrangements to Support Team Coordination with a Planning Agent

Joel E. Fischer[1] *, Chris Greenhalgh[1], Wenchao Jiang[1], Sarvapali D. Ramchurn[2], Feng Wu[3], Tom Rodden[1]

[1] *The Mixed Reality Laboratory, School of Computer Science, University of Nottingham, UK. E-mail: joel.fischer@nottingham.ac.uk, chris.greenhalgh@nottingham.ac.uk, tom.rodden@nottingham.ac.uk*
[2]*Agents, Interaction, and Complexity Group, Dept. of Electronics and Computer Science, University of Southampton, UK. E-mail: sdr1@soton.ac.uk*
[3] *School of Computer Science and Technology, University of Science and Technology of China, China. E-mail: wufeng02@ustc.edu.cn*

SUMMARY

In this paper we present the study of interactional arrangements that support the collaboration of headquarters (HQ), field responders and a computational planning agent in a time-critical task setting created by a mixed-reality game. Interactional arrangements define the extent to which control is distributed between the collaborative parties. We provide two field trials, one to study an "on-the-loop" arrangement in which HQ monitors and intervenes in agent instructions to field players on demand, and the other to study a version that places headquarters more tightly "in-the-loop". The studies provide and understanding of the socio-technical collaboration between players and the agent in these interactional arrangements, by conducting interaction analysis of video recordings and game log data. The first field trial focuses on the collaboration of field responders with the planning agent. Findings highlight how players negotiate the agent guidance within the social interaction of the collocated teams. The second field trial focuses on the collaboration between the automated planning agent and the headquarters. We find that the human coordinator and the agent can successfully work together in most cases, with human coordinators inspecting and 'correcting' the agent-proposed plans. Through this field trial-driven development process, we generalise interaction design implications of automated planning agents around the themes of supporting common ground and mixed-initiative planning.

Received . . .

KEY WORDS: CSCW; field trial; team coordination; human-agent interaction; mixed-reality games; coalition formation; instructions; computational planning.

1. INTRODUCTION

Most disaster operations require responder teams to plan and carry out geographically distributed tasks (e.g., digging out casualties or transporting civilians) with limited resources and personnel—a timely response may be critical to save lives [1].

Deciding when and how to use available resources in such a setting can be described as a "distributed resource allocation problem under temporal constraints" [2]; to that end, multi-agent task allocation algorithms have been devised and tested in computational simulations of such tasks [2, 3, 4]. These algorithms can be used to build automated planning agents that can perform complex

*Correspondence to: The Mixed Reality Laboratory, School of Computer Science, University of Nottingham, NG8 1BB, United Kingdom. E-mail: joel.fischer@nottingham.ac.uk calculations much faster than humans (e.g., computing paths, optimising team configurations). However, these algorithms necessarily depend on abstracted models of the environment and human behaviour which might lead to task allocations that are flawed in practice, due to the contingent nature of situated action [5].

We might conjecture that a human coordinator working together with the planning agent could notice and help to deal with such emergent problems. One way in which this working together might be achieved is by placing a human coordinator "in-the-loop" between the planning algorithm and the human responders in the physical world. A variation of "in-the-loop" is "on-the-loop", in which the role of the human coordinator is less involved; perhaps best described as that of a supervisor, rather than a deciding authority. Our work studies such interactional arrangements with the goal to enable efficient interaction and collaboration between humans and agents.

In order to explore the socio-technical interactional challenges related to these human-agent arrangements we developed a technology probe in the form of a mixed-reality game called AtomicOrchid [6]. Mixed-reality games bridge the physical and the digital world [7]; they make use of pervasive technologies such as smart phones, wireless technologies and sensors with the aim of blending game events into a real world environment [8]. They have served as a vehicle to study distributed collaborative interactions across multiple devices and ubiquitous computing environments in the wild [9]. In AtomicOrchid, players in the role of field responders and headquarters (HQ) coordinators have to collaborate to save spatially distributed targets from a spreading radioactive cloud. Following an ethnomethodological orientation [10], this setting makes available the observable-and-reportable team interaction with and around the planning support system in a disaster scenario for direct observation of activity.

In this paper, we report on one field trial of an on-the-loop arrangement, and another field trial of an in-the-loop arrangement. We investigate socio-technical issues that arise in relation to automated planning support with the on-the-loop and the in-the-loop interaction design. Interaction analysis [11] is conducted based on log data and video recordings of field observations, revealing how human-agent interaction is embedded in social interaction.

We provide three contributions in this paper. First, we demonstrate a field trial-driven methodology employed to reveal socio-technical issues in relation to computational planning support; second, we present findings that suggest mixed-initiative designs that place humans *in-the-loop* may be preferable in situations with unforeseen contingencies. Third, we identify key design lessons in relation to critical mixed-initiative features such as common ground between agent and humans, and mutual awareness in planning.

In the next section we review related work and our approach. We then describe the scenario and design iterations including summary results from the field trial of the base version in section 3. We then present the field trial of the on-the-loop version in section 4, and the in-the-loop version in section 5. The presented episodes of interaction serve to identify and discuss a range of key issues around the themes of division of labour, planning support, and field-trial driven development in section 6. Finally, we conclude by summarising the lessons learnt for supporting common ground and mixed-initiative planning for designers of distributed coordination systems in section 7.

2. RELATED WORK AND APPROACH

We briefly review how our approach builds on related work on planning in disaster response, both from the point of view of computational optimisation on the one hand, and empirical studies of command-and-control settings, and CSCW systems that support workflow management on the other hand.

We also briefly review the relevant literature concerned with 'interactive automation' at the intersection of interface agents and user interface design, and outline how it relates to our mixed-reality game probe to study agent-assisted collaboration.

DOI: 10.1002/cpe

2.1. *Planning in Disaster Response*

One major concern for task planning in disaster response (DR) is how to efficiently allocate limited resources to multiple spatially distributed incidents under time pressure. To address such coordination challenges in DR operations, a number of multi-agent planning algorithms have been developed to computationally support planning in time-critical task settings [2, 3, 4]. While these algorithms can rapidly compute optimal routes, model and predict certain environmental variables (e.g., wind speed, fire spreading), they typically ignore the physical and cognitive charateristics of human field responders, such as human psychosocial condition, movement, and learning ability [12], and stress, fear, exertion or panic [13]. Hence, a key motivation in our work is to create a setting in which participants experience physical exertion and stress through bodily activity and time pressure in order to increase confidence in the veracity of observations [6]. Specifically, we adopt a serious mixed-reality game approach to study how spatially distributed responders coordinate in a time critical task setting [14].

Furthermore, socio-technical studies of command and control settings (e.g. in disaster response [15], the London Underground [16], and air traffic control [17]) have revealed the complex ways in which interaction with physical and digital (or electronic) resources is embedded in face-to-face social interaction in the control room, and have argued that taking the social organisation of the cooperative work setting into account is crucial for success [17]. Further empirical studies of CSCW systems have shown that it is vital to study technology in use to understand potential tensions raised for teamwork. In particular, field studies of workflow support systems have revealed that technologies can disrupt smooth workflow if they are not designed in a socially acceptable way [18, 19]. This paper follows the tradition of the empirical CSCW studies to investigate interaction and cooperative work in situ, in order to identify implications for technology support.

2.2. *Interactive automation support*

A review of the interaction design literature yields studies that have found that the potential benefits of automation support may not always be realised, and can be offset by unwanted consequences [20, 21]. These negative consequences can include over-reliance on automation, loss of situation awareness, and loss of skills needed to perform the automated functions manually in case of automation failure [22]. It is this recognition of the potential problems of automation that raises important challenges for the design of the interface(s) between the human and the computational support.

To this end, one significant design strategy is ‘mixed-initiative’, which refers to a flexible interaction strategy where both human and software agent can contribute to the task, with each party contributing to the task according to its strengths [23]. In the most general case, each party’s role is not pre-determined, but opportunistically negotiated as the problem is being solved. So at one time the software agent might have the initiative, controlling the interaction while the human ‘monitors’ the execution (i.e., the human is on-the-loop), while at other times the human may drive the interaction, with the software agent in a supporting role (i.e., the human is in-the-loop) [24]. A number of algorithms, interfaces and applications have been devised that facilitate mixed-initiative planning and control [25, 26].

In this work, in order to study the interactional challenges that arise in these arrangements, we integrated a planning agent in the AtomicOrchid game probe. To study how on-the-loop and in-the-loop arrangements play out in practice, the interaction layer between players and agents can be configured in different ways through modifications to the game interface. Through agent integration and iterative interface design, we created three versions of ‘probes’, two of which we evaluate in this paper in some depth.

2.3. *Research focus: practical action and reasoning*

The mixed-reality game probe is used to conduct observational studies, which allow us to unpack human-system interaction in the different interactional arrangements (on-the-loop vs. in-the-loop). Our foremost analytic orientation is Ethnomethodology [10], a perspective that focuses on the accomplishment of practical action and practical reasoning by the members of a setting. Specifically, we use Interaction Analysis to unpack naturally occurring talk and activity, with the aim of uncovering and describing something of the order and organisation by which people and interact with each other and with the things around them [11].

Our interest in this paper is how socio-technical interaction is organised around the computational planning support, hence our focus is both on the action on the ground, as well as in the control room. We recorded both system logs and video of interaction in the field for analysis. To capture the distributed, concurrent nature of the interaction, four researchers with camcorders shadowed the field player teams, and one researcher recorded the action in the HQ. A replay tool was used to synchronise and analyse triangulated game events, player positions, and concurrent video recordings. These were then catalogued to identify key decision points in teaming and task allocation, which served to index sequences (episodes) of interest (cf. [27]). Interesting distinct units of interaction were then transcribed and triangulated with log files and field video for deeper analysis; the results of which we present in this paper.

3. ATOMICORCHID - SYSTEM DESIGN

In this section, we outline the system design of the mixed-reality game probe AtomicOrchid. We created AtomicOrchid in order to study team coordination, interaction, and communication in a disaster scenario. In brief, AtomicOrchid simulates a radioactive incident. Participants of the game play both the role of responders 'on the ground', and coordinators in the control room. The interactive system provides situation awareness capabilities that enable monitoring of players, tasks, radioactivity, and communication via text messaging. A planning agent is integrated into the system in order to support the teaming and task allocation of field responders.

In this section, we outline the game scenario, the iterative development rationale, a description of the planning agent integrated into the system, and we provide some more detail on the system evolution, including functionality and interface description.

3.1. Game scenario

The game, 'AtomicOrchid', is a location-based game based on a fiction of an explosion which creates an expanding and moving cloud of radioactive gas. The majority of players are "on the ground" and play the role of first responders; we refer to these as 'field players'. Two players are based in a nearby 'headquarters' (HQ) and play the role of coordinators. Within the physical game area there are several 'targets' and a small number of 'safe zones'. The goal of the game is for the field players to evacuate as many targets as possible to the safe zone(s) before the radiation cloud covers the playing area. Field players have limited 'health' which declines when they are in or near the virtual radiation cloud. If they are exposed to too much radiation field players will become 'incapacitated' ('die'). Field players need to communicate frequently with HQ, as only HQ can see the entire cloud, while field players only have a numeric 'reading' for their current location.

Within the game each field player is assigned a specific type or role: medic, transporter, soldier or fire fighter. Each target also has a specific type (animal, fuel, uranium and victim) and can only be evacuated by a two-person team with the right combination of roles. For example a soldier and a transporter are required to pick up and carry fuel to safety. One of the key challenges of the game is therefore to form appropriate transient two-person teams of field players to evacuate specific targets.

3.2. Iterative development

We progressively developed and refined AtomicOrchid and the planning agent support in three iterations. Each version focused on supporting a particular relation of the interactional arrangements (see Figure 1). In the first iteration, we developed a base version of coordination support without integrating a planning agent. The system's design focus is on supporting the collaboration between and among field responders and HQ by providing real-time text messaging and 'situational awareness' interfaces, e.g., real-time monitoring of players, tasks, and cloud. In the on-the-loop

DOI: 10.1002/cpe

version, we integrated a planning agent into the system, focused on supporting the field responders directly. The planning agent automatically generates a plan and allocates tasks to field players (hence, the HQ is merely on-the-loop). The third (in-the-loop) version is aimed at providing a stronger role for the HQ, by providing an interface that lets the control room mediate between planning agent and field responders. For each version, field trials are conducted and analysed; the findings of the first and second version have then been turned into design implications for the following version.

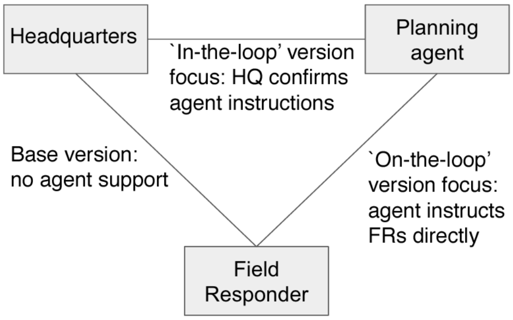

Headquarters
Planning agent
`In-the-loop' version focus: HQ confirms agent instructions
Base version: no agent support
`On-the-loop' version focus: agent instructs FRs directly
Field Responder

Figure 1. Interactional arrangement in AtomicOrchid, and the focus of each version.

3.3. *Planning agent*

In the field trials of the on-the-loop and the in-the-loop version, the player teams are supported by a software agent that acts as a 'planner'; this is in contrast to the base version [28], in which the field responders and HQ were entirely responsible for planning. The planning agent assigns evacuation tasks to field responders by making use of locations of targets and safe zones, a predictive model of the radiation cloud, and the current location and health of field responders in order to minimise their travelling distance, and maximise the number of targets rescued. A plan produced by the planning agent is a set of 'task assignments', i.e. a request for two specific field players (with particular roles) to evacuate a certain target to a specific safe zone. In the on-the-loop version the agent's plan is communicated directly to the field players. In the in-the-loop version, the agent's plan is initially made available only to the HQ players; they can check the plan and edit it if they wish; once HQ has approved the allocations they are sent to the field players.

Following the mixed-initiative principles set out in section 2.2, the design rationale is to augment, rather than to replace human decision making where each party contributes to the task according to its strength. Therefore, the human retains the capability to reject the agent's task assignments to acknowledge the uniquely human ability – unavailable to the agent – to deal with contingencies that arise in the course of action (e.g., humans may be tired, or they may have encountered a road block, etc)[†]. Note that for a plan that involves multiple responders coordinating to perform a task, having only one of the responders reject the plan means that the allocation of other responders has to be recomputed from scratch to preserve the efficiency of the planning process. Doing so can be computationally time consuming. We propose a solution to this in what follows.

To provide more technical detail, the planning agent runs a real-time multi-agent coordination algorithm to solve the coordination problem in two steps: 1) task assignment, and 2) path planning. The algorithm models the coordination problem in AtomicOrchid using a Multi-Agent Markov Decision Process (MMDP). The goal of solving MMDPs is to find the optimal policy that maximizes the number of completed tasks with minimum costs, although due to the large state space and the real-time requirement a working solution can only be approximate [29]. The model not only

[†] Note that technically, from the agent's point of view there is no difference between a 'rejection' triggered by field responders in the on-the-loop version, and a by the HQ in the in-the-loop version.

DOI: 10.1002/cpe

takes into account environmental parameters (locations, distances, cloud etc.), and actor parameters (responder role, health, etc.), but also whether tasks have been rejected. In more detail, our algorithm computes a set of plans conditioned on all possible plan rejections from the responders (i.e., combinations of rejections from individual responders), which reflect responders' preference for the plan. If the current plan is rejected, an alternative plan will be selected based on the set of rejections received. To compute such plans, our algorithm applies a Two-Pass Planning process. In the first pass, the best policy for the underlying MMDP without rejections is computed, and, in the second pass, the rejections are handled using the policy computed by the first pass. By doing so, the planner agent can quickly respond to the rejection event and generate a better plan that is more acceptable to the responders. Further technical details of the planning agent can be found elsewhere [29, 30].

3.4. Baseline version without planning agent

The system design, in particular the interfaces between the human team and the planning agent have evolved through the three iterations described. We only have space to briefly summarise the results from the field trial of the first iteration – the baseline version without the planning agent – the details of which have been presented elsewhere [28].

In the base version of AtomicOrchid without a planning agent, the HQ is manned by 2-3 coordinators. All of the coordinators are provided with a web-based coordination interface. The interface gives them an overview of the game status and enables them to communicate with the field responders who carry a phone running the Mobile Responder App. The user interfaces are similar to the interfaces shown in figure 2, but without the agent/task allocation elements.

We ran two AtomicOrchid Game sessions to field-trial the base version. The size of the game area on the local university campus is 400 by 400 meters, with little traffic. The terrain of the game area includes grassland, a lake, buildings, roads, footpaths and lawns. There are two drop off zones and 16 targets. An earlier pilot study showed that this was a challenging, yet not overwhelming number of targets to collect in a 30 minute game session. There were four targets for each of the four target types. The pattern of cloud movement and expansion was the same for both game sessions.

3.4.1. Implications for design The result of interaction analysis from video recordings of game action showed that team planning was dominated by local (face-to-face) coordination between field players in a situated manner. The field players teamed up with their teammates and selected task by utilising available resources such as local conversation, the mobile interface, and messaging remote players. The HQ was observed to successfully provide awareness of the "danger zone" to the field teams through remote messages. However, HQ had little direct influence on the planning and actions of field teams. One potential reason could be the lack of communication between HQ and field responders. The observations led to a set of design requirements to improve the usability of the system:

1. **Geospatial referencing** We found that players struggled to communicate the locations of targets and their planned routes. Although the targets' locations are displayed and shared on the map, players reference a particular target by referring to nearby landmarks or road crossing and directions (north, east). Time was wasted in such clarification of geo-referencing. Designers need to think carefully about how the presentation layer of such systems may be augmented with information that facilitates geo-spatial referencing (e.g., grids, labelling etc.) to facilitate human in addition to machine readability.
2. **Freshness of messages** We found that some messages in the communication channel become irrelevant quickly due to the changing task environment. Reading out-dated messages gives players false information about game status and can lead to dangerous actions. To reduce confusion stemming from outdated information, additional functionality is required, for example to flag messages as out-dated, to retract messages that are no longer valid, or to highlight more up-to-date messages.
3. **Acknowledgement of messages** In most cases, field responders did not acknowledge or respond to messages sent by the HQ. This was particularly problematic for instructions from

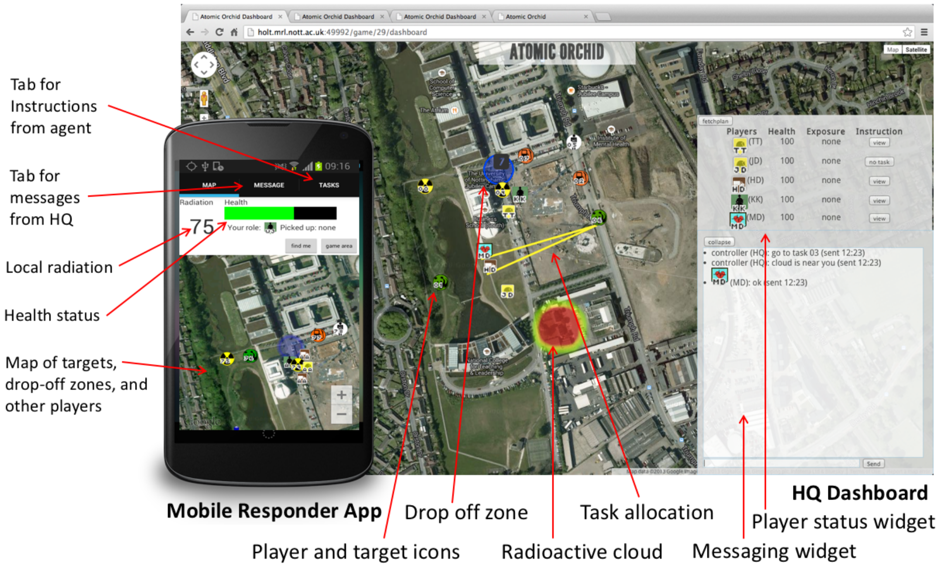
DOI: 10.1002/cpe

> HQ, as task status and field responder compliance often had to be inferred by observing their location updates on the map. This consumed HQ attention, with negative impact on HQ's overall work on state assessment and task planning. Observations in the field suggest that the physical demands (e.g., co-located team movement through terrain at speed) and cognitive demands to maintain situational awareness (e.g., monitoring of radioactivity and messages) are likely factors that explain lack of acknowledgement. User interfaces that enable and encourage field responders to quickly and easily acknowledge HQ messages should be considered for messaging in such high demand settings.

These requirements have been taken into account in the development of the the on-the-loop version.

3.5. On-the-loop version

In the second version, the game interfaces were modified according to the design requirements generated from field-trialling the base version (see Figure 2). First, messages in the messaging interface are appended with timestamps to allow players to identify their freshness. Second, targets on the digital maps are marked with a unique task number to ease geo-referencing. Third, a feedback system is built into AtomicOrchid to assist quick acknowledgement. The feedback system is part of the integration of the planning agent, which is detailed in the following section.



Players	Health	Exposure	Instruction
(TT)	100	none	view
(JD)	100	none	no task
(HD)	100	none	view
(KK)	100	none	view
(MD)	100	none	view

- controller (HQ): go to task 03 (sent 12:23)
- controller (HQ): cloud is near you (sent 12:23)
- (MD): ok (sent 12:23)

Figure 2. HQ and mobile interfaces in the on-the-loop version.

3.5.1. User interfaces As can be seen in the figure 2, the majority of the HQ dashboard is occupied by a map-based presentation of the current game status. Roles and locations of field responders are represented on the map as icons. The field responders can be uniquely identified by their initials shown on the icons. The target types and locations are also shown as icons on the map. Location and intensity of the radioactive cloud is indicated by a heatmap. Health status (health value ranges from 0 to 100) of the field responders is displayed on the right-top panel. A chatbox at the right bottom for HQ allows browsing, composing and sending messages. The messaging system follows a broadcasting model: everyone can send messages to one public channel, and the messages are visible to every player through the mobile and HQ interface. The agent's team-task allocations can be shown visually at the click of a button.

Field responders are equipped with a mobile responder app providing them with sensing and awareness capabilities (also Figure 2). There are three tabs in the responder app. The "map" tab displays a map showing locations of field responders and targets, which is similar to the map on the HQ interface, except that the cloud is not shown. The radiation level of the players' current location is displayed as a Geiger counter reading (shown as a number on the top left of the screen), which ranges from 0 to 100. Health status of the field responder is indicated by a health bar on the right side of the Geiger counter. The chatbox (similar to the one on HQ interface) is placed on the "Message" tab for the field player to receive and send messages. Finally, the "Tasks" tab shows the agent's task allocations.

3.5.2. The planning agent Apart from improvements in interface usability, crucially, we integrated a planning agent into the AtomicOrchid platform in the on-the-loop version. The planner (described above in section 3.3), is deployed on a separate server, which exposes a HTTP interface for AtomicOrchid to request plans. Each plan request issued by AtomicOrchid is appended with updated game status, which includes players' health, distribution of radioactive cloud and locations of players, and targets. Based on the updated game status, the planner will produce an optimised task allocation and return it to AtomicOrchid. The plan requests are triggered frequently in game sessions so that the task allocation can be frequently adjusted according to task execution status. In this version, plan requests (and thus re-planning) is triggered by two kinds of game events:

1. Completion of task. On successful rescue of a target, a new plan (i.e., allocation of tasks to each responder) is requested from the agent.
2. Explicit reject. On rejection of a task allocation by any of the first responders, a new plan is requested.

On receiving an instruction from the planner, the field responder can choose to either reject or accept the instruction in the 'Task' tab of the app, the rationale for which is detailed above in section 3.3. In the case of rejection, a new plan will be requested and the agent will take into account the rejection in the next iteration of task assignment. More importantly, the rejected allocation is used as a constraint within the optimisation run by the planner. For example, if two responders (a medic and a soldier) were allocated a task and the solider rejected it, the planning agent would return a new task allocation with the constraint that this soldier should not be allocated this task. Unlike the later human-in-the-loop version, the planning agent retains the control over task assignments. In this version HQ could only intervene by using the communication channel in order to study an arrangement in which the agent has a relatively stronger role.

The instructions sent to field responders are also displayed in the HQ interface for monitoring purposes. The task allocations are represented as yellow lines connecting players and their targets (Figure 2). Only one task allocation is displayed at a time when the HQ player clicks on the 'show' task button on the player status panel.

The next section provides findings from the field trial of the on-the-loop version in more detail. After that, we turn to the in-the-loop version, and its final summative evaluation in section 5.

4. "HUMAN ON-THE-LOOP" INTERACTIONAL ARRANGEMENT

This section provides an abbreviated presentation of the field trial results reported in a prior publication [31]. The field trial of this version follows the same game setup as the base version (see section 3.4). A total of 16 participants were recruited through posters and emails, and reimbursed with 15GBP for 1.5-2 hours of study. The majority were students of the local university. The procedure consisted of 30 minutes of game play, and about 1 hour in total of pre-game briefing, consent forms, a short training session, and a post-game group discussion.

Through interaction analysis of video recordings of game action and system logs, we gain insight into the division of labour between human and agent in which the agent takes over routine planning activities while the human focuses on other issues such as finding teammates, targets and choosing the best routes.

After presenting an overview of how task assignments were handled in the field trial, we present episodes that reveal how teams accomplish the tasks in the rescue mission, particularly focusing on the social organisation of interaction with and around the agent instructions.

4.1. *Overview of task assignments*

Figure 3 shows how task assignments were acted upon in the field trial. 51 assignments were created by the planner and sent to field responders. 24 were accepted, while 11 were rejected or did not receive a response, i.e., only one or none of the two involved players responded. Out of the accepted tasks, 15 were completed successfully. An additional 8 tasks were completed that had not received a response (2 of which without agent instruction).

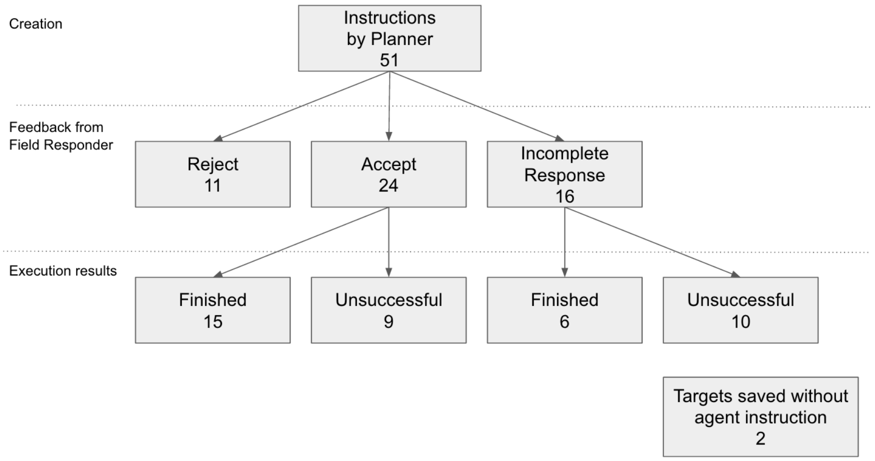


Figure 3. How instructions were handled in the on-the-loop version.

4.2. *Episodes from the field*

In the following episodes, players can be uniquely identified by their initials. Targets are denoted by their unique numeric target id. Task assignments from the agent are represented as two pairs of initials and one target id connected by a rightward arrow. For example, the notation *PC, CR* → *22* means player PC and CR are instructed to team up and go for target 22. A standard orthographic notation [11] includes non-verbal elements"((..))" and pauses in seconds, e.g., "(1.0)"; this is complemented by timestamps [0:00], and **system messages** from remote players and HQ.

4.2.1. Episode 1 - task assignment The following episode depicts a team of two dropping off a target and planning the next step.

[0:00] The team drop off a target.

PC: I think we dropped off now. OK.

[0:07] The team receives a new agent instruction: *PC, CR→22*

PC: I have a task now (3.0) ((studying screen)). I need to go with CR to 22. Are you CR?

CR: Yes.

PC: Let's go 22.

CR: We have done 22.

PC: Oh (1.0) no (2.0) 22 is there ((pointing in the direction)). Let's go ((PC starts walking, CR follows)).

[0:28] The team complete the task (drop off target 22).

DOI: 10.1002/cpe

At the beginning of this episode, the team (PC, CR) drops off a target at a drop off zone. Player PC vocalises that they have finished the task ("I think we dropped off now. OK"). After about 7 seconds, PC says she received a new task allocation from the agent ("I have a task now"). PC confirms the initials of the other player (CR), and suggests CR to join her to go for target 22. The action is consistent with the agent instruction (*PC, CR → 22*), suggesting that PC has read the instruction and decided to follow it. CR said that they have already finished target 22 ("We have done 22"), which indicates he is confused about the current task allocation. PC resolves the confusion by pointing in the direction of 22 and repeating to go for it. Later, the team successfully drop off target 22 as instructed by the agent.

The episode shows how an agent instruction is brought up and followed by a team in a relatively straightforward manner. The instruction was delivered immediately after the drop off of a previous target (7 seconds after). PC successfully locates the new target in the instruction and leads the team to pick it up. Although CR is confused at first, PC manages to rectify CR's mistake and they finish the task successfully.

This episode is a typical case of task assignment to existing teams, i.e. the agent sent a new task to a team immediately after they finished their previous task. Out of a total of 51 agent instructions, 23 fall into this category. The rate of compliance is high for these cases of task assignment to existing teams (21 out of 23; 91%).

4.2.2. Episode 2 - team reformation Unlike episode A, sometimes the agent instruction implies players need to disband and form new teams after finishing their previous task, in order to enact the computationally optimal plan. 10 out of 51 agent instructions fall into this category. The compliance rate of instructions that require reteaming (50%) is substantially lower than compliance of instructions where players can stay in the same teams (91%). The following episode depicts a typical case in which team reformation fails.

[0:00 After a target drop off, LT and SS joined PC and CR at the drop off zone.

[0:24] HQ: LT, if you think you have the stamina to run to 10 around the north of the lake do so now with a firefighter. (message A)

[0:28] New agent instruction: *NK, LT→16*

LT: They said ((reads out aloud HQ message A))

[0:35] CR: ((facing LT)): Shall we go get 10.

LT: Mine is 16.

[0:38] HQ: Avoid 17 at all costs (...) I'd avoid 10, too. (message B)

CR: ((read out HQ message B)) avoid 10 now.

[0:55] New agent instruction: *NW, LT→15*

LT: 15!

[1:24] After some deliberation, SS and PC decide to go for target 19, leaving CR and LT behind.

[1:29] NW (via chat): LT where you.

CR: ((turning to LC)) Are you LT?

LT: Yes.

CR: NW is looking for you.

LT: Yah thanks. ((turning away from CR)) Ah::. I will go towards them. ((starts walking))

CR: Okay. Do you want company?

LT: ((turning back towards CR)) Yeah.

CR and LT leave the drop off zone together to find NW.

The episode begins with a recommendation by HQ to LT to go for 10 (message A). The message is topicalised by LT, but it is soon overridden by an agent instruction (*NK, LT→16*). When CR proposes to team up with LT to go for target 10, LT declined ("mine is 16"). HQ then withdraws its previous suggestion to go for 10 in message B. Shortly after, a new instruction (*NW, LT→15*) prompts LT to read out the target number (15), but she fails to raise the other players' attention. While other group

DOI: 10.1002/cpe

members are engaged in planning next steps, LT does not engage and keeps looking around. She can be seen turning and walking back and forth. Perhaps LT is trying to locate the player NW who she had been instructed to team up with. LT does not take any action until prompted by CR ("are you LT? NW is looking for you"). Then, LT begins to walk to find her teammate. However, when she finally manages to meet up with NW two minutes later, NW has already been assigned another task.

On one hand, LT seems to feel obliged to follow the agent instructions. She turns down other teaming invitations and appears to try to look for NW in her immediate vicinity, indicating difficulty with locating teammates out of sight (despite the real-time location map). On the other hand, her body orientation displays a sense of attachment to the existing group. Her indecisive walking and turning back and forth suggests she struggles to leave. She does not leave the group to follow the instructions until prompted by someone. When CR points out NW's message, LT does not answer the message either. The episode illustrates a combination of interactional 'troubles' as a result of which the reteaming fails: being attached to the local group, struggling to locate teammates out of sight, and failing to reciprocate messages.

Further, we found the distance between instructed players to be a key factor in successful reteaming. That is to say, if instructed players are not within line of sight, the rate of non- compliance with the agent instruction is high. Taking episode B as an example, player LT was instructed to team up with a distant player twice. Neither one of the instructions was successfully implemented. Overall, there were 17 agent instructions that implied teaming with distant players; only 1 of them were actually followed by players. Players explicitly rejected 11 of them by pressing the rejection button; the other 5 were not followed without an interface action.

4.2.3. Episode 3 - task interruption

AW and KD are in the process of walking to target 44.

[0:00] AW receives new agent instruction: *AW, YF→46*

AW: New instruction 46! ((both stop walking))

KD: Do they know we are already on the task?

[0:06] AW receives a new agent instruction: *AW, LC→37*

AW: yea, but I think, Oh, no, got new instruction again, (team up with) LC.

[0:19] AW starts walking to LC, who is at drop off zone within line of sight, leaving behind KD.

KD: KD: ((reads out an old HQ message)) AW and KD you wont reach 44. Alright, Lets go to 46.

AW: ((turning back towards KD)) I dont know, I got a new task with LC.

KD: Ahh, I do not have a task.

AW turns and walks towards LC again. KD follows.

In this fragment, we can observe disagreement and negotiation about team reformation. AW receives 2 consecutive reteaming instructions from the agent, finally teaming them up with LC, while KD does not receive another instruction. KD's question ("Do they know we are already on the task?") suggests that he might think the agent is unaware of their situation, and that he disagrees with disbanding the existing team. In spite of KD's disagreement, AW declares his intention to follow the new instruction ("got new instruction again, [team up with] LC") and he turns to find LC. However, KD ignores this ("Alright, Lets go to 46"), indicating he does not agree with AW's intention to disband the team. AW interjects ("I don't know, I got a new task with LC"), and continues to walk towards LC, denying KD. As KD realizes he is without assignment ("Ah, I do not have a task"), he follows AW to find LC.

In this episode, teammates agree to reject the first task assignments. We found task interruption could be a major reason to reject new instructions. 10 out of 11 rejected instructions are associated with task interruption. In an extreme case (not pictured), one team reached an agreement to ignore any agent instructions after the agent tried to interrupt the team's on-going task.

In the end, the player that received the new instruction disagrees with his teammate's suggestion to ignore the instruction and decides to leave the current team. The team is disbanded in disagreement; the teammates spend a fair amount of time arguing whether to follow or ignore instructions, hinting at the hidden social cost of 'coalition formation' algorithms when applied to human teams.

Overall, the majority of new instructions that interrupted on-going tasks required team reformation. When tasks were interrupted, the rate of compliance (22%) is substantially lower than when teams were required to reform after a task was completed (50%). Task interruptions were also much more likely to lead to rejection of the new assignment (10 out of 11 assignments that interrupted tasks were rejected.).

4.3. *The headquarters*

HQ sent a total of 147 messages in the two sessions. We identified 50 assertives and 68 directives in two sessions through speech act analysis. The majority of assertives were focused on providing situational awareness and safe routing for the responders to avoid exposing them to radiation. E.g. "NK and JL approach drop off 6 by navigating via 10 and 09." Or "Radiation cloud is at the east of the National College".

6 out of 68 directives were directly related to task allocations and teaming, which is substantially less than the number of agent instructions (51). Among the 16 directives, HQ sent 11 direct instructions to the field players (e.g. "SS and LT retrieve 09"), while the remaining 5 are related to forward planning, (e.g., "DP and SS, as soon as you can head to 20 before the radiation cloud gets there first"). 6 of the HQ instructions are consistent with agent instruction, while 5 other HQ instructions override the agent instructions. It is worth mentioning that field players implemented only 5 out of 16 HQ instructions. In the interview, HQ reported that they felt they supported the agent rather than taking control.

4.4. *Implications for design*

Our observations reveal the tension between agent planning support and the social organisation of teamwork. The tension does not simply mean the model held by the agent is "incorrect"; it highlights potential trade-offs we need to consider in system design [18]. As a result, we propose three design implications to scaffold the division of labour when building agent-based planning support for human teams.

1. **Facilitating accountability** We found players often reach a decision with their co-located team member to reject new tasks that would split the team, but without considering its impact on other remote members. As a result of receiving a rejection, the planner agent re-plans and sends out new task allocations for everyone. Consequently, the remote team members may experience frequent task changes for unclear reasons. Therefore, we suggest the interaction design should reveal the hidden cost of certain actions (e.g., rejections) to facilitate the accountability of local decision making to remote team members, ensuring consequences of local decisions for the welfare of all teams are understood.
2. **Social cost of team reforming** The agents algorithm re-plans and reshuffles teams, in order to optimise group performance by minimising the travel distance to the targets. However, we observed that players are often unwilling to disband teams and discard ongoing tasks. Team reformation (instructed by planner agent) is frequently associated with delays caused by discussion, disagreement and task rejections. We categorise this kind of coordination overhead as the 'social cost' of team reformation. The planner agent used in this study does not have the ability to model and take into account the social cost. In general, we posit that it may be hard to model every aspects of a human team. In turn, system designers may need to consider the 'imperfection' of planner agents and design an interaction layer that can alleviate this issue.
3. **Weak role of HQ** We found that HQ struggled to influence the plan because of the lack of interface level support. Their attempts to override agent plans (through the text messaging channel) were often ignored, missed, or resulted in confusion. This observation highlights the need to provide interface level support to strengthen the role of the HQ in the planning loop.

5. "HUMAN IN-THE-LOOP" INTERACTIONAL ARRANGEMENT

In the final in-the-loop version, we took into account the design implications from the on-the-loop version. We were interested to see whether the aforementioned issues of accountability, and 'social cost' may be alleviated by a stronger role of HQ in the planning loop. Therefore, we enabled a 'human in-the-loop' arrangement in which HQ can mediate between the planning agent and the field players. In this arrangement, the human HQ can request task allocations from the agent at any point, and then needs to approve the generated allocations. Once the allocations are approved the task allocations are sent to the field players, who are then able to respond by accepting or rejecting their assigned task. However, in this version any task 'rejections' from field players are merely requests for the HQ to change the allocations; final task allocation remains at the HQ's discretion. As a result of the evaluation of the on-the-loop version, HQ can also communicate preferences to the agent, for example to 'keep' a certain task assignment when replanning. To clarify, we list several requirements that are necessary of the 'in-the-loop' design.

This is communicated back to the HQ players, and the HQ players can request a new plan based on field players' feedback at any point.

1. HQ should be able to review, edit and approve every instruction generated by the agent.
2. HQ should be able to decide when the agent should re-plan.
3. HQ should be able to modify plans for some of the players, leaving the agent to plan for the rest of the players.
4. HQ should be able to communicate their task assignments (or task cancellation) to field responders in a structured way.
5. HQ should be able to add task-specific information to each sent assignment.

The purpose of requirement 1-2 is to give HQ more control over the planning loop, by delegating to them the responsibility for the final planning decision. Requirement 3 enables HQ to modify the plans computed by the agent without having to take full manual control of plan generation. Requirement 4 is derived from the observations from the base version and the on-the-loop version that HQ struggled to override agent planning through unstructured text messages. New HQ and mobile interface were developed to facilitate the in-the-loop design.

5.1. Improved user interfaces

Because the on-the-loop version of the HQ interface (see Figure 2) has proved effective for monitoring the game status, the interface was kept for operation by one of the HQ players in the control room (HQ2). In addition, a new task assignment interface was developed and operated by an additional HQ player in the control room (HQ1, see Figure 4). The new task assignment interface is designed to support HQ monitoring and intervention in the plan-execution loop. The interface enables HQ to approve and edit agent-suggested task assignments and monitor player feedback.

The task assignment interface has a live map view on the left (Figure 4) which shows current player and target locations, and task assignments. The right side of the interface is occupied by the task assignment panel. The left (1) column of the panel shows 'pending' (i.e. proposed but unconfirmed) task assignments while the right column (2) shows current (confirmed) tasks. When the operator presses the plan request button (3), the agent will calculate a plan based on current task status which is then shown in the pending panel. If the player then presses the plan edit button (4) then the assignments in the pending area become editable through drag-and-drop interaction. Pressing the plan approval button approves all pending assignments, which move to the current (confirmed) area.

Figure 4 (5) shows an example of a proposed task assignment: player MP and GO are assigned to target 07. Within each confirmed task assignment (6) a feedback indicator shows the field player's response to this assignments (no response, reject, accept). The stop button terminates an assignment, for example in an emergency. A 'keep' checkbox causes the planner to retain the corresponding task assignment whenever it generates a new plan. A text messaging panel is linked to the current selected

DOI: 10.1002/cpe

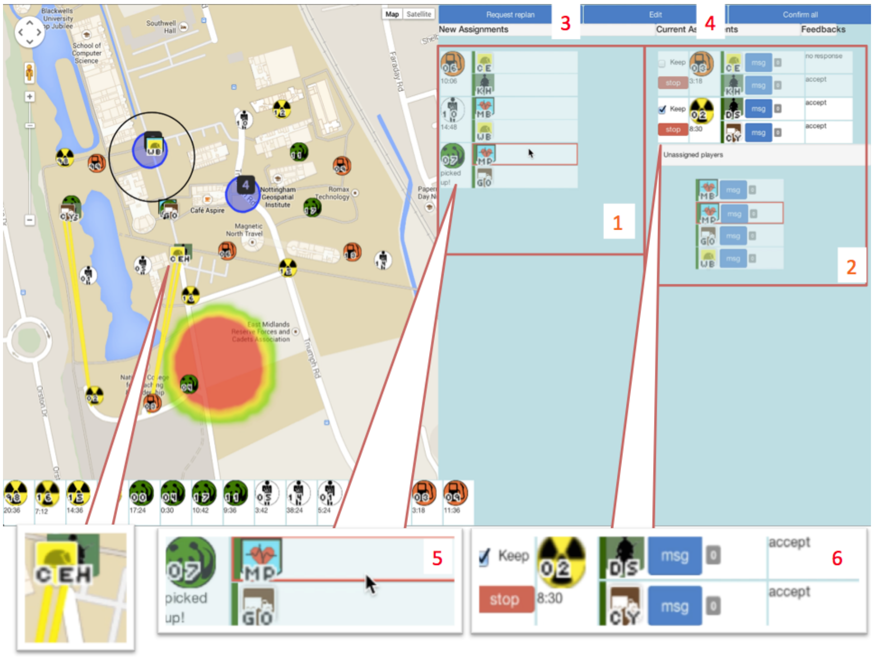
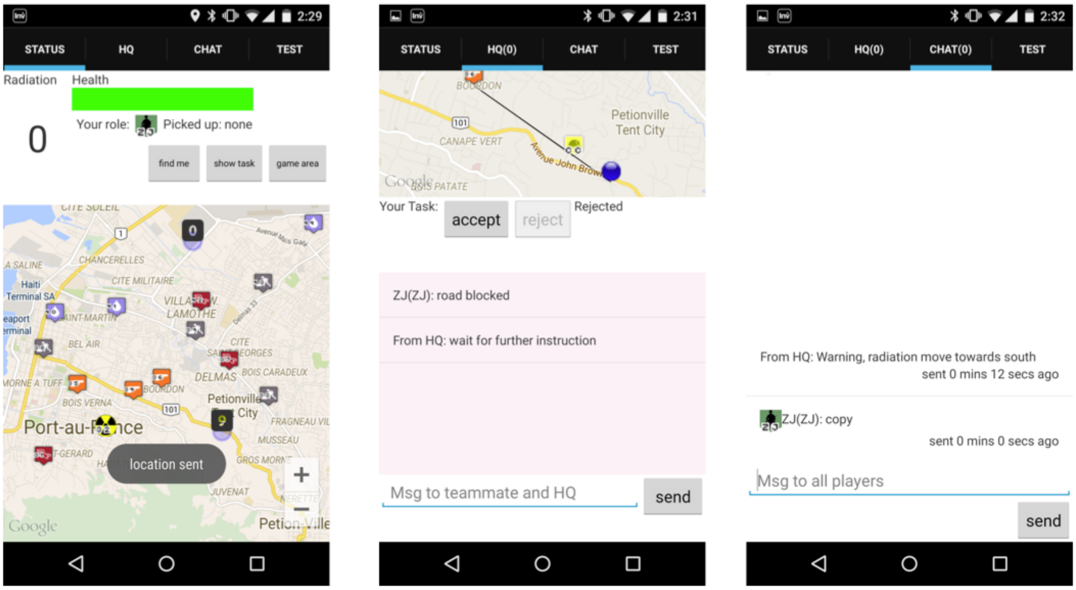


Figure 4. Task assignment interface with live map view (left) and task assignment panel (right).



ZJ(ZJ): road blocked

From HQ: wait for further instruction

Msg to teammate and HQ send

From HQ: Warning, radiation move towards south
sent 0 mins 12 secs ago

ZJ(ZJ): copy
sent 0 mins 0 secs ago

Msg to all players send

Figure 5. Mobile responder app: ‘Status’ tab (left), task / HQ chat tab (middle), and global chat (right).

task assignment and allows the two players involved in the assignment and HQ1 to exchange task-specific messages.

Compared to the on-the-loop version of AtomicOrchid, the mobile interface is largely unchanged except for the HQ task/chat tab (see middle of Figure 5). The task tab now displays a task with text description and map visualisation of the task at the top. The bottom half of the interface is a message box showing task-specific information from HQ. It should be noted that the HQ can still send broadcast information (visible to everyone), which will be displayed in the chat tab.

5.2. *Summative field trial*

We ran two AtomicOrchid sessions to trial the in-the-loop version. Each session follows the same procedure as the base version and the on-the-loop version. Detailed results of the interaction analysis is presented in the next section. Overall, 70% (28 of the 40) of the targets were evacuated in the in-the-loop version, which is similar to the on-the-loop version (71.8%).

The following subsections start with an overview of task assignments. Task assignments serve to 'index' the beginnings of potential episodes of interests in our qualitative data corpus. Selected episodes of game play are then presented in order to unpack the interactions surrounding the task assignment activities in the control room. We provide these episodes as *vivid exhibits* of how members accountably organise their team coordination in situ [32].

5.2.1. Overview of task assignments Figure 6 shows how task assignments were acted upon in the in-the-loop arrangement. Overall, the planning agent created a total of 45 task assignments with an additional 5 assignments created manually by HQ. HQ approved a total of 39 assignments. Field responders accepted most of the approved assignments (30 out of 39). Only 1 assignment is rejected by field responders, and 8 assignments did not receive a response[‡]. During task execution, occasional HQ interventions resulted in 5 task cancellations and 5 assignments being overridden.

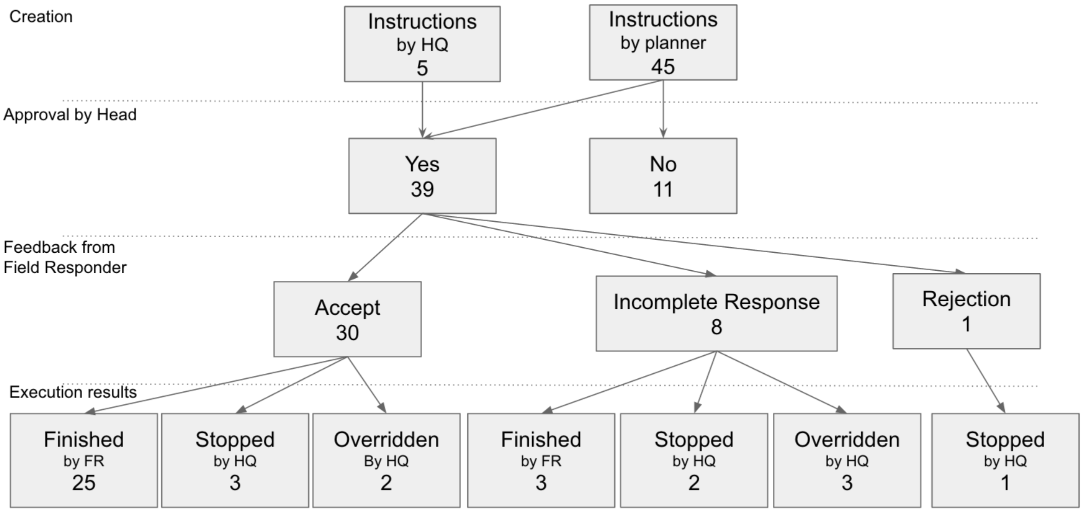


Figure 6. How instructions were handled in the in-the-loop version.

5.2.2. Responses to task assignments This section presents selected episodes of game play in order to unpack the interactions surrounding the task assignment activities in the control room. The presentation of the episodes follows the same notation as introduced in section 4.2.

5.2.3. Episode 4 - Confirming the plan As summarised above, a majority of task assignments are generated by the planning agent and approved by the HQ players. Episode 1 illustrates a typical case of task planning and approval.

```
 Team MV, XW are carrying target 43 back to drop off zone 7.
[16:45] HQ2: XW and MV.
[16:50] HQ1: (taking) 43.
[16:51] HQ2: They should be going to drop off (zone) 7 and get 36.
[16:49] HQ2: tell them to go to 36 afterwards.
```

[‡] only one or none of the two instructed players responded

[17:04] HQ1: ok this one ((pointing to target 36)).

[18:10] The team drop off target 43.

[18:22] HQ1: ((clicks re-plan))

[18:26] New agent assignments, including: MV,XW → 36.

[18:28] HQ1 36, yes ((clicks confirm)).

At the beginning of episode 1, HQ2 is drawing attention to his monitoring of MV and XW, who are confirmed by HQ1 to be carrying target 43. Given their current location, HQ2 is able to deduce ‘they should be going to drop off zone 7’, and is also able to anticipate that they should then ‘get 36’, referring to the next target assignment. As HQ1 is manning the task assignment interface that includes the task specific chat, HQ2 instructs HQ1 to ‘tell then to go to 36 afterwards’, which HQ1 confirms in turn and acknowledges by pointing at the target on his screen. A short while later, after the team dropped off the target, HQ1 requests a new plan from the agent, upon which the agent suggests team MV, XW is assigned target 36. This assignment is consistent with their previous discussion as confirmed by HQ1’s utterance ‘36, yes’. HQ1 approves the assignment by clicking ‘confirm’. The assignment is sent to the field responders, who in turn accept the assignment.

This episode depicts a typical case of unproblematic agent-supported task assignment. As figure 6 shows, 34 (39 less 5 created by HQ) out of 45 of the agent’s allocations are approved without editing. Worthy of note is that the HQ can be seen to be monitoring the field responders in their ongoing task execution by means of the interfaces provided, which enables them to *plan ahead* for the next task assignment. As a result, they do not make a timely request for new task assignments from the agent, but they have already selected an appropriate next task (‘36’), probably based on its location and requirements; this suggests that the interface is providing the HQ with sufficient information (e.g., regarding player, target and radiation) in order to come to a decision about which task to allocate. Notably, this decision is the same decision that the agent has arrived at, which confirms the HQ in their planning, and lends support to their decision making. However, HQ does not always agree with the agent’s assignment, as the following episode will show.

5.2.4. *Episode 5 - Correcting the plan*

HQ players chose to change the task assignments generated by the planning agent in 11 out of 45 cases (see figure 6); episode 2 presents one such example.

CE and KH are currently assigned target 03 but have not accepted; other players are free.

[04:18] ((HQ1 clicks ‘request plan’.))

New agent-suggested allocations arrive: *CE, KH → 06; MP, GO → 07; MB, WB → 10)*

[04:24] HQ1: What? Why am I getting? Ah:: one of these guys did not accept. (Referring to the team of CE, KH.)

[04:29] ((HQ1 clicks ‘keep’ button on assignment *CE, KH -> 03*)).

[04:33] ((HQ1 clicks ‘request plan’ again))

[04:40] New agent allocations arrive: *MB, WB → 10; MP, GO → 07.*

[04:42] ((HQ1 clicks confirm))

This episode begins with HQ requesting a new plan from the agent. The agent proposes a set of assignments, one of which (CE, KH → 06) would interrupt an ongoing task (CE, KH → 03), much to the disapproval of HQ1 (“What? Why am I getting?”). The task assignment ‘03’ had previously been sent to KH and CE; however, whilst they are ostensibly in the process of doing the task (apparent by their location and direction of movement), they have not both ‘accepted’ the task. Hence, the responders ‘look’ available to the agent, which in turn suggests a new task for KH and CE.

HQ1 realises the fact that they have not explicitly accepted the previous task (“Ah:: one of these guys did not accept.”). HQ then instructs the agent not to change the existing assignment [04:29] by use of the “keep” checkbox and requests a new plan, which is generated without the conflicting assignment. As a result, the changed plan is in turn confirmed.

It then is noteworthy that in contrast to the episode presented in section 4.2.3, the task assignment interface allows HQ1 to avoid interrupting the field responders’ current task, in that HQ1 is not only able to notice, but also able to compensate for the field players' failure to explicitly accept the task. As a result, the field players are able to continue with the previously allocated task without interruption, and oblivious to HQ's intervention in the control room. However, in contrast to this unproblematic instance of plan correction, the next episode will show that editing of the agent's allocations does sometimes not lead to desirable outcomes.

5.2.5. Episode 6 - Changing the plan At the start of this game session we can observe one of the HQ players overriding three out of four of the planning agent's allocations.

All players are together in the drop off zone, idly waiting for initial tasks.

[01:25] ((HQ1 requests initial plan))

[01:28] 4 agent-assignments arrive.

[01:29] **HQ1:** Why, it is stupid.

[01:33] ((HQ1 clicks 'edit'))

[01:53] **HQ1:** I want this one ((manually drags targets to replace agent assignments)) (.) this one (.) a:nd this one. (The three prioritized targets are very close to the radioactive cloud).

[02:03] ((HQ1 clicks 'confirm')).

[02:07] **HQ1:** ((turning to HQ2)) I think we should get the far ones first.

HQ1 requests initial task assignments for all of the field players. The planning agent provides HQ1 with a set of task assignments for approval, but HQ1 is not happy with them ("Why, it is stupid."). HQ switches into 'edit' mode and replaces 3 of the targets in the agent assignments, voicing his intention as he is performing the editing. The three manually assigned new targets are the ones that are closest to the radioactive cloud. HQ1 confirms his modification [02:03] and provides an account of his strategy to HQ2: "we should get the far ones first", probably referring to the distance of the selected targets from the field responders' current location.

The episode shows how the capability to change the agent's allocations allows HQ to implement their own strategy and priorities. The design rationale of this 'feature' was to enable human decision-making in response to situational contingencies to take precedent over the agent's rigid world model. However, things do not work out so well in this case. The modified plan turned out to be undesirable as it leads to two assignment cancellations and two players 'dying' as they attempt to rescue a target from the radioactive cloud. In the end, only one of the three modified assignments was finished successfully.

5.3. Comparative evaluation

Herein, we provide some key metrics in order to compare compliance (task acceptance) and team performance (task completion) between the on-the-loop and the in-the-loop version. Note that this comparison may be confounded by changes in the user interface made between versions, and by individual and between group differences. The objective of the statistical comparison is to be informative and to supplement the qualitative analysis, which is the main focus of our analysis.

Table I shows key metrics for both versions. Compared to the on-the-loop version, the task assignments in the in-the-loop version have relatively higher success rate: 28 out of 39 (72%) assignments are completed successfully, while only 21 out of 51 (42%) assignments were completed successfully in the field trial of the on-the-loop version.

Compared to the on-the-loop trial, the task assignments in the in-the-loop trials have relatively higher acceptance rate. 30 out of 39 (77%) assignments are accepted by the field players, while only 24 out of 51 (47%) assignments are accepted in the on-the-loop trial. An independent samples t-test indicated that acceptance rate was significantly higher for the in-the-loop version (M = 0.77, SD = 0.43), than for the on-the-loop version (M = 0.47, SD = 0.5), t(87) = 3.04, p = 0.003. Levene's test indicated unequal variances (F = 19.45, $p < 0.001$), so degrees of freedom were adjusted from 88 to 87.

In addition, an independent samples t-test shows that the completion rate of tasks in the in-the-loop version (M = .72, SD = 0.46) is also significantly higher then that in the on-the-loop version

(M = 0.42, SD = 0.5), t(85) = 3.04, p = 0.003. Again, Levene's test indicated unequal variances (F = 6.5, p = 0.012), so degrees of freedom were adjusted from 88 to 85.

In summary, the results show significant improvements from the on-the-loop to the in-the-loop version in the key evaluation metrics of acceptance and completion of task assignments.

	Success rate	Failure rate	Acceptance rate
On-the-loop arrangement	21/51 (42%)	30/51 (58%)	24/51(47%)
In-the-loop arrangement	28/39 (72%)	11/39 (28%)	30/39(77%)

Table I. Result overview

6. DISCUSSION

As the core part of the analysis of the field trials we have presented detailed episodes of interaction to illustrate how collaboration was achieved in practice. We now draw out our observations on key interactional themes displayed in the data, and we reflect on the improvements between the versions.

6.1. *On field trial-driven iterations*

The results in section 5.3 show that task acceptance and completion has been significantly improved from the on-the-loop version to the in-the-loop version. Moreover, the communication between HQ and the field players has been largely unproblematic in the final version, and most targets were successfully evacuated according to plan. The outcomes seem to be considerably better than for the on-the-loop version.

In particular, the HQ players in the on-the-loop version were observed to struggle to intervene in the planning process. In a paper presented at CTS in 2014 we have argued that there is a 'hidden cost' associated with the agent's task interruption and instructions that require team reformation [31]. The episodes presented in section 4 illustrate the local interactional 'troubles' (e.g., disagreement, locating teammates) implicated by allocations that require reteaming (episode 2) and interrupt ongoing tasks (episode 3).

These findings in turn inspired the design rationale towards a stronger HQ *in-the-loop* that we hoped would alleviate some of the problems associated with 'unfiltered' agent instructions. In the *on-the-loop* arrangement, the only way for HQ to intervene in the planning is to send unstructured text messages in the broadcast channel. The fact that only 5 (out of 16) HQ instructions were acted on in on-the-loop version suggests that HQ was unable to effectively override the agent when they wanted to.

The improved task acceptance and completion rate do suggest that the performance is significantly improved in an *in-the-loop* arrangement compared to the earlier *on-the-loop* arrangement. Specifically, HQ's ability to intervene has been enhanced by the mixed-initiative task allocation interface introduced in the *in-the-loop* arrangement.

In sum, our evaluation has not only shown that task allocations computed by the planner are more likely to be accepted by field responders when there is a human in the loop who confirms or modifies each allocation according to the situation at hand, but also that this arrangement leads to a better task completion rate. More broadly, the move towards a stronger *in-the-loop* arrangement highlights the need for interfaces that provide means for humans to moderate and intervene in agent-based planning in order to respond to situational contingencies. The following sections explore the findings with regard to division of labour and further planning support.

6.2. *Working together*

Herein we reflect on the division of labour between the field responders, the HQ, and the planning agent observed in the field trials reported in earlier sections. The rationale for the planning agent's integration was to take on some of the work load involved in planning. Episode 1 demonstrates a typical case of division of labour: the agent handles planning of teaming and task assignment, freeing the field responder team to focus on navigational issues (identifying the target on the interactive map and finding directions). However, we have already lamented the trade-offs implicated by the comparatively 'weak' role of the HQ in on-the-loop arrangement, which led to the aforementioned improvements.

The field trial of the in-the-loop version showed that in many cases the communication between HQ and the field players is unproblematic, and most targets are successfully evacuated according to plan. Hence, the situation has improved considerably, and we conjecture that this is due at least in part to differences in the user interface in the in-the-loop version. Specifically, to recap, the main changes are the HQ-manned task allocation interface, and the improved Mobile Responder App. In the mobile app, the current task allocation is shown as a graphical overlay on the mobile map in the in-the-loop version, not just as a textual instruction (given by the HQ player in the base version or the planning agent in the on-the-loop version. This seems to significantly reduce the field players' confusion about their current target and team-mate and where to find them.

Furthermore, the task planning interface for the most part appears to provide an effective shared representation of the current state of the game. As well as showing current player and target locations and player health, it also makes visible the currently approved task allocations, field player responses and any new plan that has been requested or is being edited. This shared information forms the common ground between the HQ players and the planning agent.

The evaluation has demonstrated that HQ players closely monitor this view and its representation of plan execution. For example episodes 4, 5, and 6 all reveal HQ players' awareness of field player progress and current tasks. Episode 2 and 6 show awareness of the cloud's location in relation to players, and episodes 5, and 6 show HQ players engaging actively with proposed (rather than current) task assignments. We observe that the HQ players are quite capable of modifying the agent's plans when they wish to, for better (episode 5) or worse (episode 6). HQ is also able to intervene in current task allocations, which is successful in resolving the situation in episode 5.

6.3. *Support for Human Planning*

As seen in episodes 4, 5, and 6, HQ players are observed to utilise the task interface to assess current game status, while in episodes 5, and 6 we have also seen how they can modify the agent's plans. This suggests that the interface is sufficient in providing basic situational awareness for HQ players to make their own plans.

6.3.1. Changing the agent's plan The drag-and-drop based task assignment interface in the in-the-loop version also enforces various constraints on task assignment so that all plans are at least valid, i.e. well-formed. For example, each player and each target can be assigned to at most one task, and each task can only have players with the correct combinations of game roles for the target. The interface also highlights players and targets on the map when they are manipulated so that the HQ player can readily assess location and proximity when editing task assignments. However, the observations also reveal some potential for improving support for human planning.

Returning to episode 6, where the HQ player massively revises the agent's assignments (leading to undesired outcomes), one future idea is to enable the planning agent to 'comment' with regard to potential problems in the player's proposed plan. While making visible the planning agent's reasoning might have discouraged the player from changing the plan so dramatically, there will still surely be situations in which plans could or should be changed. And in future we may improve the system beyond leaving the player to "do their best". For example, the planning agent could simulate (and perhaps extend) the proposed modified plan to provide the HQ player with at least one *predictive* view of the possible outcomes of their plan.

6.3.2. Forward planning In the current system the agent performs forward planning, i.e. it considers what field players might do in the future, not just in the current/next task assignments. In future, this information could be made available to the HQ players. In episode 4 we also saw one of several examples of the HQ players also planning for future task assignments. In future, HQ could be enabled to record their own forward planning and thereby feed back into the system instead of

DOI: 10.1002/cpe

having to make a note or remembering what they were thinking when the current task is completed and they have the chance to check and intervene. Therefore, at least for some situations it might be beneficial if the agent's future plans could also be viewed, and if the HQ players also had some system-support to guide their own future thinking.

6.3.3. Other interactional troubles We have also encountered the following interactional challenges that likely generalise more broadly to related settings.

- **Complacency** describes the phenomenon whereby occasional failures of automation remain undetected by the operator. In particular, this may occur when the operator has learnt to trust the computational component, and is repetitively exposed to its outputs. This finding echoes results on 'automation bias' in the supervisory control literature [33]. Specifically, in some cases in the in-the-loop arrangement it appeared HQ approved the agent's assignments quickly without any verbal discussion, which resulted in unnecessary team reformations when editing as seen in episode 5 did not occur. However, mechanisms to counter this may turn towards more human involvement in the planning; therefore, there will be a fine balance so as not to overburden the operator.
- **The agent's hidden reasoning process** Some manually modified assignments lead to undesirable results (e.g. episode 6). Effective sharing of the agent's reasoning process and the potential consequences of its modifications may prevent some undesired assignment modifications.
- **Non-responsiveness** Although quick acknowledgement is supported in the final prototype, non-responsiveness of field players still caused trouble for HQ players. This non-responsiveness was also observed to create uncertainty in the planning. It is important to realise that this could be caused by technical communication outages as well as by human non-response. Designers may attempt to incorporate this as 'known unknowns' in the system, for example planning could be done with an estimated *probability* of a positive response.

7. CONCLUSIONS

Herein, we provide the lessons learnt that may benefit the designers of distributed coordination systems, in particular, in relation to situation awareness, computational planning support and interactional breakdowns. These may be particularly relevant for settings in which timely human decision-making is critical.

7.1. Common ground

Common ground is a critical requirement for making collaborative decisions in an effective and timely manner. Through our field trials, we identified the following features as constitutive of common ground through providing a mutual situation awareness for the participating parties (HQ, field responders, and the agent).

- **Domain-specific information models** (task allocations in this case) were critical for establishing common ground between the planning agent and players. In addition, specific message types added to the common perspective of HQ and field players.
- **Appropriate representations** that made use of domain-specific visual cues (e.g. linking tasks elements to the map) enabled alignment and consistency across views and between mobile (field) and HQ representations, and supported practical reasoning about these activities. Similarly, the task interface allowed HQ players to read, modify, and confirm the agent's task assignments.
- **Articulation of future actions** was also a key part of player's situated planning work, as evidenced by their engagement with *proposed* tasks, both within and beyond the task interface.

Our observations also align with the theoretical framework model of situational awareness proposed by [34], which argues the this needs to be supported by three levels including 1) Perception of the elements in the environment 2) Comprehension of the current situation and 3) Projection of future status.

7.2. Supporting Mixed-Initiative Planning

Some opportunities and challenges have also become evident that relate more specifically to the possibilities of mixed-initiative planning.

- **Domain-specific constraints** were seen to be an advantage of using planning support in this setting, with the task interface allowing only well-formed task assignments.
- **Making the agent's reasoning visible** would perhaps have avoided some of the situations in which responders were sent in harm's way. The challenge is how to implement this without inundating the operator with information.
- **Feedback on human proposals** might have supported the task interface's ability to arbitrarily modify task assignments, e.g. identifying possible hazards.
- **Forward planning** was being done by both the agent and the people in the HQ as they anticipated future actions, and this might have been better supported if it had been recorded and made visible, perhaps on demand.

Our observations echo work on human considerations in context-aware systems, which propose principles to support intelligibility and accountability [35]; similarly we stress that the goal for planning support systems should be to be accountable for their actions, therefore, 'what they know, how they know it, and what they are doing about it' [ibid., p. 201] needs to be legible by the people involved. Furthermore, as planning is oriented towards the future, yet produced as a contingent, situated activity [5], the interface needs to support revision and revoking of plans in situ, and furthermore provide the situational awareness essential to do so.

7.3. Future work

Our findings should not be overgeneralised. In this work, we compared two different human-agent arrangements to study the emergent interaction. However, our goal was not to find the optimal system to solve the task allocation problem. While our results suggests that the in-the-loop arrangement was preferable to the on-the-loop arrangement, it was not without issues and there may be other arrangements and improvements that could have led to better performance, and reduced losses. Therefore, we suggest that future work could and should improve the system further. Particular aspects that could be improved further include both mixed-initiate interfaces and the computational intelligence for distributed task allocation problems. For example, further means to communicate emergent issues back to the planning agent should be considered; however, the potential gains of such features would need to be carefully considered against the additional workload for the responders.

Overall, we foresee that there usually are unforeseen contingencies that humans need to deal with; hence, we feel strongly that a consideration of how contingencies can be responded to would need to be incorporated from the outset in any future work building on the contributions of this work.

8. ACKNOWLEDGMENTS

This work is supported by the Engineering and Physical Sciences Research Council [grant number EP/I011587/1]. Transcripts of the qualitative data are provided in full in this paper, the quantitative data can be accessed at http://dx.doi.org/10.17639/nott.62 in accordance with EPSRC research data policy.

REFERENCES

1. Chen R, Sharman R, Rao HR, Upadhyaya SJ. Design Principles of Coordinated Multi-incident Emergency Response Systems. *Intelligence and Security Informatics* 2005; :81–98.
2. Ramchurn SD, Polukarov M, Farinelli A, Truong C, Jennings NR. Coalition formation with spatial and temporal constraints. *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, International Foundation for Autonomous Agents and Multiagent Systems, 2010; 1181–1188. URL http://dl.acm.org/citation.cfm?id=1838186.1838191.
3. Chapman A, Micillo R. Decentralised dynamic task allocation: a practical game: theoretic approach. *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2.*, International Foundation for Autonomous Agents and Multiagent Systems, 2009; 915–922. URL http://dl.acm.org/citation.cfm?id=1558139.
4. Scerri P, Farinelli A, Okamoto S, Tambe M. Allocating tasks in extreme teams. *Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems* 2005; :727doi:10.1145/1082473.1082584. URL http://portal.acm.org/citation.cfm?doid=1082473.1082584.
5. Suchman LA. *Plans and Situated Actions: The Problem of Human-machine Communication*. Cambridge University Press: New York, NY, USA, 1987.
6. Fischer JE, Jiang W, Moran S. Atomicorchid: A mixed reality game to investigate coordination in disaster response. *Proceedings of the 11th International Conference on Entertainment Computing*, ICEC'12, Springer-Verlag: Berlin, Heidelberg, 2012; 572–577, doi:10.1007/978-3-642-33542-6_75. URL http://dx.doi.org/10.1007/978-3-642-33542-6_75.
7. Benford S, Magerkurth C, Ljungstrand P. Bridging the physical and digital in pervasive gaming. *Commun. ACM* Mar 2005; **48**(3):54–57, doi:10.1145/1047671.1047704. URL http://doi.acm.org/10.1145/1047671.1047704.
8. Crabtree A, Tandavanitj N, Steed A, Benford S, Rodden T, Greenhalgh C, Flintham M, Anastasi R, Drozd A, Adams M, *et al.*. Orchestrating a mixed reality game 'on the ground'. *Proceedings of the 2004 conference on Human factors in computing systems*, ACM Press, 2004; 391–398, doi:10.1145/985692.985742. URL http://portal.acm.org/citation.cfm?doid=985692.985742.
9. Crabtree A, Benford S, Greenhalgh C, Tennent P, Chalmers M, Brown B. Supporting ethnographic studies of ubiquitous computing in the wild. *Proceedings of the 6th Conference on Designing Interactive Systems*, DIS '06, ACM: New York, NY, USA, 2006; 60–69, doi:10.1145/1142405.1142417. URL http://doi.acm.org/10.1145/1142405.1142417.
10. Garfinkel H. *Studies in Ethnomethodology*. Prentice-Hall, 1967.
11. Jordan B, Henderson A. Interaction analysis: foundations and practice. *The Journal of the Learning Sciences* [11]:39–101.
12. Armenakis C. Systems approach to management of disasters: methods and applications. *Natural Hazards* 2013; **66**(1):139–140, doi:10.1007/s11069-012-0197-2. URL http://dx.doi.org/10.1007/s11069-012-0197-2.
13. Drury J, Cocking C, Reicher S. Everyone for themselves? a comparative study of crowd solidarity among emergency survivors. *British Journal of Social Psychology* 2009; **48**(3):487–506.
14. Fischer JE, Flintham M, Price D, Goulding J, Pantidi N, Rodden T. Serious mixed reality gamges. *Mixed Reality Games Workshop, The 2012 ACM Conference on Computer Supported Cooperative Work (CSCW '12)*, 2012.
15. Fischer J, Reeves S, Rodden T, Reece S, Ramchurn S, Jones D. Building a birds eye view: Collaborative work in disaster response. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, 2015.
16. Heath C, Luff P. Collaboration and control : Crisis management and multimedia technology in london underground line control rooms. *Computer-Supported Cooperative Work* 1992; :69–94.
17. Richard H R Harper DZS John A Hughes. Working in harmony: an examination of computer technology in air traffic control. *Proceedings of the First European Conference on Computer Supported Cooperative Work*, 1989; 73–86.
18. Bowers J, Button G, Sharrock W. Workflow From Within and Without : Technology and Cooperative Work on the Print Industry Shopfloor Introduction : Workflow Systems and Work Practice. *Fourth European Conference on Computer-Supported Cooperative Work*, Springer Netherlands, 1994; 51–66.
19. Abbott K, Sarin S. Experiences with workflow management: issues for the next generation. *Proceedings of the 1994 ACM conference on Computer supported cooperative work.*, ACM Press, 1994; 113–120. URL http://dl.acm.org/citation.cfm?id=192886.
20. Manzey D, Onnasch L. Human Performance Consequences of Automated Decision Aids : The Impact of Degree of Automation and System Experience. *Journal of Cognitive Engineering and Decision Making* 2012; doi:10.1177/1555343411433844.
21. Parasuraman R, Manzey D. Complacency and bias in human use of automation: An attentional integration. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 2010; **52**(3):381–410, doi:10.1177/0018720810376055.Copyright.
22. Kaber D, Endsley M. Out of the loop performance problems and the use of intermediate levels of automation for improved control system functioning and safety. *Process Safety Progress* 1995; **16**(3).
23. Allen JE, Guinn CI, Horvtz E. Mixed-initiative interaction. *IEEE Intelligent Systems* 1999; **14**(5):14–23, doi:10.1109/5254.796083.
24. Horvitz E. Principles of mixed-initiative user interfaces. *Proceedings of the SIGCHI conference on Human Factors in Computing Systems.*, ACM Press, 1999; 159–166, doi:10.1145/302979.303030.
25. Burstein M, Ferguson G, Allen J. Integrating agent-based mixed-initiative control with an existing multi-agent planning system. *Proceedings Fourth International Conference on MultiAgent Systems* 2003; (January):389–390, doi:10.1109/ICMAS.2000.858489. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=858489.
26. Hardin B, Goodrich MA. On Using Mixed-Initiative Control : A Perspective for Managing Large-Scale Robotic Teams. *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, ACM Press, 2009.

27. Heath C, Hindmarsh J, Luff P. *Video in qualitative research Analysing Social Interaction in Everyday Life*. Introducing qualitative methods, Sage Publications: London, 2010.
28. Fischer JE, Jiang W, Kerne A, Greenhalgh C, Ramchurn S, Reece S, Pantidi N, Rodden T. Supporting team coordination on the ground: Requirements from a mixed reality game. *Proceedings of the 11th International Conference on the Design of Cooperative Systems*, Springer International Publishing, 2014; 49–67, doi:10.1007/978-3-319-06498-7_4. URL http://dx.doi.org/10.1007/978-3-319-06498-7_4.
29. Ramchurn SD, Wu F, Jiang W, Fischer JE, Reece S, Roberts S, Rodden T, Greenhalgh C, Jennings NR. Human agent collaboration for disaster response. *Journal of Autonomous Agents and Multi-Agent Systems* Feb 2015; :1–29doi:10.1007/s10458-015-9286-4. URL http://link.springer.com/10.1007/s10458-015-9286-4.
30. Wu F, Ramchurn SD, Jiang W, Fischer JE, Rodden T, Jennings NR. Agile Planning for Real-World Disaster Response. *International Joint Conference on Artifical Intelligence*, 2015.
31. Jiang W, Fischer J, Greenhalgh C, Ramchurn S, Wu F, Jennings N, Rodden T. Social implications of agent-based planning support for human teams. *Collaboration Technologies and Systems (CTS), 2014 International Conference on*, 2014; 310–317, doi:10.1109/CTS.2014.6867582.
32. Crabtree A, Rouncefield M, Tolmie P. *Doing design ethnography*. Springer-Verlag London, 2012.
33. Cummings ML. Automation bias in intelligent time critical decision support systems. *AIAA 1st Intelligent Systems Technical Conference*, vol. 2, 2004; 557–562.
34. Endsley M. Design of situation awareness in complex systems. *Proceedings of the Second International Workshop on symbiosis of humans, artifacts and environment. 2001.*, 2001.
35. Bellotti V, Edwards K. Intelligibility and accountability: Human considerations in context-aware systems. *Hum.-Comput. Interact.* Dec 2001; **16**(2):193–212, doi:10.1207/S15327051HCI16234_05. URL http://dx.doi.org/10.1207/S15327051HCI16234_05.