Applying Strategic Reasoning for Accountability Ascription in Multiagent Teams

Vahid Yazdanpanah[1,*] , **Sebastian Stein**[1] , **Enrico H. Gerding**[1] , **Nicholas R. Jennings**[2]
[1]University of Southampton
[2]Imperial College London
v.yazdanpanah@soton.ac.uk, {ss2, eg}@ecs.soton.ac.uk, n.jennings@imperial.ac.uk

Abstract

For developing human-centred trustworthy autonomous systems and ensuring their safe and effective integration with the society, it is crucial to enrich autonomous agents with the capacity to represent and reason about their accountability. This is, on one hand, about their accountability as collaborative teams and, on the other hand, their individual degree of accountability in a team. In this context, accountability is understood as being responsible for failing to deliver a task that a team was allocated and able to fulfil. To that end, the semantic (strategic reasoning) machinery of the Alternating-time Temporal Logic (ATL) is a natural modelling approach as it captures the temporal, strategic, and coalitional dynamics of the notion of accountability. This allows focusing on the main problem on: *"Who is accountable for an unfulfilled task in multiagent teams: when, why, and to what extent?"* We apply ATL-based semantics to define accountability in multiagent teams and develop a fair and computationally feasible procedure for ascribing a degree of accountability to involved agents in accountable teams. Our main results are on decidability, fairness properties, and computational complexity of the presented accountability ascription methods in multiagent teams.

1 Introduction

For developing human-centred Trustworthy Autonomous Systems (TAS), accountability reasoning plays a key role as it contributes to: assessing the reliability of task allocations, ensuring verifiably safe and responsible human-agent collectives, and measuring the extent of each individual agent's contribution to potential failures [Yazdanpanah *et al.*, 2021b].

Accountability, as the task-oriented form of *responsibility*, is understood as being responsible for failing to deliver an allocated task [Yazdanpanah *et al.*, 2021a]. This notion relates to but is distinguishable from the epistemic notion of *blameworthiness* (as being responsible for knowingly causing an outcome) and the normative notion of liability/culpability (as being responsible for causing a normatively undesirable outcome) [van de Poel, 2011; Alechina *et al.*, 2017; Chockler and Halpern, 2004]. We abstract from such neighbouring notions, as well as the exact procedure of task allocation, and merely focus on the notion of accountability.

Supporting the reliability and verifiability of AI systems are key to ensure a trustworthy performance of autonomous systems in human-agent collectives, i.e., ensuring a desirable behaviour of TAS [Jennings *et al.*, 2014; Abeywickrama *et al.*, 2019]. Then, measuring the extent of agents' contribution to potential failures—e.g., undelivered tasks—constitutes their degree of accountability for such undesirable state of affairs [van de Poel, 2011].

In addition to technological importance, addressing the accountability ascription problem—by determining accountable teams and ascribing a degree of accountability to involved agents in a fair and computationally feasible fashion—also contributes to comply with ethical AI guidelines [EC: The High-Level Expert Group on AI, 2019]. It enables determining who is to account for a (potentially undesirable) system behaviour and fosters the societal alignment of autonomous systems [Russell, 2019; Office for Artificial Intelligence - GOV.UK, 2020; Kalenka and Jennings, 1999].

In relation to the concept of *explainability* in autonomous systems [Miller, 2019; Belle, 2017], as the capacity to describe *why* a particular behaviour is materialised, *accountability* is focused on determining *who* and *to what extent* should account for it [van de Poel, 2011]. In the responsibility reasoning literature, accountability is understood as (i.e., is a form of) task-oriented responsibility. In the prospective form, one allocates a task τ to (an agent or) agent group α and sees them accountable to bring it about. Then in the retrospective form of accountability (which is the main focus of this work), if τ remains unfulfilled, α is to account for $\neg\tau$. This is in particular a challenging problem in situations where tasks are allocated to agent *groups* or coordinated *teams*. Then, observing that a task is not delivered makes it clear that a team is to account for it; but the the extent/degree of accountability of each individual is not well-defined. This problem corresponds to what is known as a *responsibility voids* in the literature on moral responsibility [Braham and van Hees, 2011] and the retrospective dimension of *task coordination* in mul-

[*]Contact Author: v.yazdanpanah@soton.ac.uk

tiagent systems [Yazdanpanah *et al.*, 2020].

Although responsibility voids are well-studied in the philosophical literature [Braham and van Hees, 2011; van de Poel *et al.*, 2012] and multiagent systems research [Yazdanpanah *et al.*, 2019; Friedenberg and Halpern, 2019], various aspects of their task-oriented dual, i.e., accountability voids in multiagent teams, are less-explored. A notion of accountability is used in [Baldoni *et al.*, 2019] for engineering business processes and in [Baldoni *et al.*, 2020] to reason about organisational robustness; but they do not capture accountability voids in multiagent teams.

For the first time, this paper presents a verifiable notion of accountability in ATL semantics, approaches the problem of accountability voids in multiagent teams, and develops algorithmic logic-based methods to resolve them. We show the applicability of our methods and present formal results on decidability and computational complexity of the presented solution concepts. Our accountability ascription techniques contribute to developing verifiably safe and responsible autonomous system and support their integrability to form trustworthy human-agent collectives.[1]

2 Accountability Analysis in ATL Semantics

In this section, we present the intuition behind our work using a running example, analyse various conceptual aspects, and recall key formal notions.

2.1 Conceptual Analysis

Imagine a vaccination project that demands 6 units of vaccine (each unit sufficient for vaccinating 1000 patients) to be delivered and injected while we have vaccine delivery agents a_1, a_2, and a_3 with the capacity to deliver 5, 3, and 2 units of vaccine, respectively; and injection specialist agents a_4, a_5, and a_6, respectively capable of injecting 1, 5, and 5 units of vaccine. To fulfil this project, one needs to allocate the tasks to capable teams of agents. The task allocation process itself is well-studied in the multiagent systems context [Macarthur *et al.*, 2011] and is beyond the focus of this work. Task allocation can be done in an efficient manner (e.g., by allocating tasks to a minimal team of agents) or in a more resilient fashion (e.g., by considering backup teams and allocating each task to more than one capable team). For instance, a_1, a_3, a_4, and a_5 can collectively handle the project as they are able to efficiently fulfil both the delivery task as well as the injection task in this project. In a more resilient allocation (which also requires some form of coordination), delivery and injection tasks in this project can be allocated also to the backup team a_1, a_2, a_4, and a_6. This team overlaps with the main team while a_2 and a_6 can substitute for tasks originally allocated to a_3 and a_5.

Our focus in this work is not on the allocation itself but on the *accountability ascription* problem: verifying who are the *accountable* teams if a task-oriented (vaccination) project fails and on determining each agent's *degree of accountability* for such an outcome. Although we abstract from the allocation process, it is crucial to note that the accountability ascription problem follows the ascription process (in a temporal order) and relates to the properties that the allocation satisfies. In particular, if the allocation process merely gives each task to a single agent, there is no need to determine a degree of accountability as tasks are directly linked to an accountable agent, i.e., each agent is fully accountable for failed tasks that were allocated to her. However, in real-life applications (e.g., in our vaccination project) single agents may be incapable of delivering the tasks, thus it is necessary to allow allocating tasks to agent teams. While allocating tasks to teams provides more flexibility, any task failure leads to so called "*accountability voids*" and what is known as the "*problem of many hands*" [Braham and van Hees, 2011; van de Poel *et al.*, 2012]—where a team is clearly accountable but the degree of accountability of each member is not well-defined. We deem that having a clear understanding of, and computationally tractable methods for ascribing, individual's degree of accountability is key for defining justifiable sanctioning measures and, in turn, coordinating the behaviour of multiagent teams towards desirable ones.

2.2 ATL Notions and Formal Preliminaries

To model Multiagent Systems (MAS) and reason about their behaviour, we use standard Concurrent Game Structures (CGS) and employ the syntax of the Alternating-time Temporal Logic (ATL), adopted from [Alur *et al.*, 2002]. The ATL language and CGS, as its semantic machinery, allow representing and reasoning about the temporal modalities of tasks and accountability. In addition, ATL is implementable using well-established model checking tools [Lomuscio *et al.*, 2017] and is expressive for specifying team-level capacities (e.g., in contrast to similar logics like the Computation Tree Logic). Having modalities to reason about the strategic capacity of groups of agents, and not only individuals, make ATL and the machinery of CGS natural choices as they allow modelling team-level accountability and support the transition towards individual-level abilities—crucial for resolving accountability voids.

Formally, we model a MAS as a CGS $\mathcal{M} = \langle \Sigma, Q, Act, \Pi, \pi, d, o \rangle$ where:

- $\Sigma = \{a_1, \dots, a_n\}$ is a finite, non-empty set of n agents;
- Q is a finite, non-empty set of *states*;
- Act is a finite set of atomic *actions*;
- Π is a set of atomic propositions (with $p \in \Pi$ as a generic proposition);
- $\pi : \Pi \mapsto 2^Q$ is a propositional evaluation function (determining propositions that hold in a state);
- $d : \Sigma \times Q \mapsto \mathcal{P}(Act)$ a function that specifies the sets of actions available to agents at each state;

[1] Note that accountability is related to but different from the normative and legal notion of *liability* [Hart, 2008; Yazdanpanah *et al.*, 2021a] — which is not the focus of this work. We argue that distinguishing various forms of responsibility and developing operational tools to reason about them in AI systems are key to ensuring the trustworthy behaviour and safety of such systems. In this work, we focus on reasoning about accountability in multiagent teams which itself can be a base, but not the only requirement, for ascribing liability in a given context and with regard to a set of legal rules and regulative norms.

- o is a transition function that assigns the outcome state $q' = o(q, \alpha_1, \dots, \alpha_n)$ to state q and a tuple of actions $\alpha_i \in d(a_i, q)$ that can be executed by Σ in q.

In a CGS $\mathcal{M}$, state formulae $\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi$ $(p \in \Pi)$ specify properties of states and path formulae $\psi ::= \bigcirc\varphi \mid \varphi\mathcal{U}\varphi \mid \Box\varphi$ specify temporal properties over sequences of states. $p \in \Pi$ is a proposition (an atomic property that may be valid in a state in Q), $\neg$ and $\wedge$ are standard logical operators, $\bigcirc\varphi$ means that φ is true in the next state of $\mathcal{M}$, $\psi\mathcal{U}\varphi$ means that ψ has to hold at least until φ becomes true; and $\Box\varphi$ means that φ is always true. We denote $\neg\Box\neg\varphi$ by $\diamond\varphi$. This modality refers to the truth of φ in some point in time in future and is known as the "existence" or "sometimes-in-future" modality. In our work, we specify *tasks* as path formulae and (task) *projects* as a set of tasks (examples will be provided later). ATL is generic for specifying temporal properties over infinite sequences. However, due to the temporally finite nature of tasks in real-life applications (e.g., most tasks have a deadline), we follow [De Giacomo and Vardi, 2015] and introduce a finite notion of *history*, and base our accountability reasoning on such finite traces. In the following, to improve readability, we directly refer to elements of a specific (also known as pointed) CGS $\mathcal{M}$, e.g., as $\mathcal{M}$ is fixed, we write Q instead of Q in $\mathcal{M}$.

Successors, Computations, and Histories: For two states q and q', we say q' is a *successor* of q if there exist actions $\alpha_i \in d(a_i, q)$ for $a_i \in \Sigma$ in q such that $q' = o(q, \alpha_1, \dots, \alpha_n)$, i.e., agents in Σ can collectively guarantee in q that q' will be the next system state. A *computation* of a CGS $\mathcal{M}$ is an infinite sequence of states $\lambda = q_0, q_1, \dots$ such that, for all $k > 0$, we have that q_k is a successor of q_{k-1}. We refer to a computation that starts in q as a *q-computation*. We denote the k'th state in λ by $\lambda[k]$, and $\lambda[0, k]$ and $\lambda[k, \infty]$ respectively denote the finite prefix $q_0, \dots, q_k$ and infinite suffix $q_k, q_{k+1}, \dots$ of λ. Finally, we say a finite sequence of states $q_0, \dots, q_n$ is a *q-history* if $q_n = q$, $n \geq 1$, and for all $0 \leq k < n$ we have that q_{k+1} is a successor of q_k. We refer to any q_k on a history h as a member of h.

Strategies and Outcomes: A *strategy* for an agent $a \in \Sigma$ is a function $\zeta_a : Q \mapsto Act$ such that for all $q \in Q$, we have that $\zeta_a(q) \in d(a, q)$. For a group of agents $\Gamma \subseteq \Sigma$, a *collective strategy* $Z_\Gamma = \{\zeta_{a_i} \mid a_i \in \Gamma\}$ is an indexed set of strategies, one for every $a_i \in \Gamma$. Then, $out(q, Z_\Gamma)$ is defined as the set of q-computations that agents in Γ can enforce by following their corresponding strategies in Z_Γ.

Formulas of the language $\mathcal{L}_{ATL}$ are defined by the following syntax, $\varphi, \psi ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid \langle\langle\Gamma\rangle\rangle \bigcirc \varphi \mid \langle\langle\Gamma\rangle\rangle \varphi\mathcal{U}\psi \mid \langle\langle\Gamma\rangle\rangle \Box \varphi$ where $p \in \Pi$ is an atomic proposition, and $\Gamma \subseteq \Sigma$ is a typical group of agents. Informally, $\langle\langle\Gamma\rangle\rangle \bigcirc \varphi$ means that Γ has a strategy to ensure that the next state satisfies φ; $\langle\langle\Gamma\rangle\rangle \varphi\mathcal{U}\psi$ means that Γ has a strategy to ensure ψ while maintaining the truth of φ; and $\langle\langle\Gamma\rangle\rangle \Box \varphi$ means that Γ has a strategy to ensure that φ is always true. The semantics of ATL is defined relative to a CGS $\mathcal{M}$ and state q and is given below:

- $\mathcal{M}, q \vDash p$ iff $q \in \pi(p)$;
- $\mathcal{M}, q \vDash \neg\varphi$ iff $\mathcal{M}, q \nvDash \varphi$;
- $\mathcal{M}, q \vDash \varphi \wedge \psi$ iff $\mathcal{M}, q \vDash \varphi$ and $\mathcal{M}, q \vDash \psi$;
- $\mathcal{M}, q \vDash \langle\langle\Gamma\rangle\rangle \bigcirc \varphi$ iff exists a strategy Z_Γ such that for all computations $\lambda \in out(q, Z_\Gamma)$, $\mathcal{M}, \lambda[1] \vDash \varphi$;
- $\mathcal{M}, q \vDash \langle\langle\Gamma\rangle\rangle \varphi\mathcal{U}\psi$ iff exists a strategy Z_Γ such that for all computations $\lambda \in out(q, Z_\Gamma)$, for some i, $\mathcal{M}, \lambda[i] \vDash \psi$, and for all $j < i$, $\mathcal{M}, \lambda[j] \vDash \varphi$;
- $\mathcal{M}, q \vDash \langle\langle\Gamma\rangle\rangle \Box \varphi$ iff exists a strategy Z_Γ such that for all computations $\lambda \in out(q, Z_\Gamma)$, for all i, $\mathcal{M}, \lambda[i] \vDash \varphi$.

Moreover, given a formula φ, we denote by $[\![\varphi]\!]_\mathcal{M}$ the set of states in which φ holds.

3 Verifiably Accountable Teams

We assume a function $T^d_\mathcal{M}(\Gamma, \varphi, q)$ that has direct access to the matrix of already allocated tasks and returns 1 if a given state formula φ is expected to be delivered by a team $\Gamma \subseteq \Sigma$ by a particular state $q \in Q$, and returns 0 otherwise. Then $T^{d,\varphi}_{\mathcal{M},q}$ denotes the set of such teams. (In this formulation, we represent single agents as singleton groups.) Accordingly, to have a track of the point of allocation we use $T^a_\mathcal{M}(\Gamma, \varphi, q)$ in which returning 1 means that in q, team Γ received the task to fulfil φ. Then $T^{a,\varphi}_{\mathcal{M},q}$ denotes the set of such teams. If a project P is allocated to Γ, then $T^a_\mathcal{M}(\Gamma, \varphi_i, q) = 1 : \forall \varphi_i \in P$. Here, superscripts "$a$" and "$d$" are a part of the function names to distinguish whether q is the state in which a task is allocated (a) or expected to be delivered (d). For instance, if the task of delivering a vaccines box (denoted by v) is allocated to agent group $\{Alice, Bob\}$ in state q_1 and expected to be fulfilled by state q_4 then $T^a_\mathcal{M}(\{Alice, Bob\}, v, q_1) = 1$ (determining the point of task allocation) and $T^d_\mathcal{M}(\{Alice, Bob\}, v, q_4) = 1$ (determining the expected point of task delivery). These auxiliary notions allow defining the task-oriented notion of accountability as follows.

Definition 1. *In a multiagent system modelled by CGS $\mathcal{M}$, let φ be a state formula, q be a state, $h = q_0, \dots, q_n (q_n = q)$ be the materialised q-history, and Γ be a team of agents. We say Γ is weakly q-accountable for φ based on h iff:*

1. $q \notin [\![\varphi]\!]_\mathcal{M}$*,*
2. $T^d_\mathcal{M}(\Gamma, \varphi, q) = 1$*, and*
3. *there exist $q_i, q_j (i \leq j) \in h$ such that $T^a_\mathcal{M}(\Gamma, \varphi, q_i) = 1$ and Γ has a strategy in q_j to ensure that $\mathcal{M}, q \vDash \varphi$.*

Moreover, a team Γ is q-accountable for φ based on h iff it is weakly q-accountable and there exist no weakly q-accountable $\Gamma' \subset \Gamma$ for φ. Analogously, Γ is (weakly) q-accountable for a project P based on h iff it is (weakly) q-accountable for all $\varphi \in P$.

Informally, a team is (weakly) q-accountable for φ only if it is not the case while the team was tasked and able to see to it that φ holds in q. Distinguishing the weak form of accountability is to realise who are the core members of a team, minimally accountable for a failed task. To have a reasonably fair degree of accountability, we later focus on this minimal group. Moreover, note that tasks in a project may be fulfilled through a path and not necessarily jointly in a particular state. This can be reduced to the availability of a path (and accordingly a strategy) such that tasks are satisfied through the path. If joint delivery is required in a domain, tasks can be bounded together in a conjunctive form and defined as a single task—and not as different members of a project. In general, tasks in a project can be delivered sequentially. The temporal modality inside each task specification determines the temporal requirements on when it should be delivered.

3.1 Accountability Reasoning in Practice

Our vaccination scenario can be modelled as the 3-state partial CGS presented in Figure 1. (We say partial as it depicts only some of the states, necessary for our accountability reasoning, and not all of the possible states.) As discussed, various task allocation processes can be used. Imagine the case that in q_0, the delivery task with a temporal expectation that vaccines should be delivered immediately, i.e., $\bigcirc \varphi_D$, is allocated to $\{a_1, a_3\}$ and then in q_D, the task to inject vaccines immediately, i.e., $\bigcirc \varphi_I$, is allocated to $\{a_4, a_5\}$.

Given these allocations, if we reach to q_1, we have the history $h = q_0, q_1$ and the delivery team $\{a_1, a_3\}$ is q_1-accountable for φ_D as they were tasked to and able to deliver all the 6 units of vaccine. However, in this situation, no team is q-accountable for φ_I as the task to inject was specifically allocated in q_D which is not a state in the materialised history. However, the allocation process could be less granular and (instead of micro-managing each task in particular states and efficiently allocating them to only one group) have allocated the project $P = \{\bigcirc \varphi_D, \bigcirc\bigcirc \varphi_I\}$ to Σ in q_0. This means that all the agents in Σ are expected to ensure that vaccines are delivered in an immediate next state after q_0 and then are all injected in one immediate state further. Then in q_1, team Σ would be weakly q-accountable for both non-delivery and non-injection. Next, we present the generalised form of such properties on the relation between (weak) accountable teams on the task level and project level.

3.2 Properties

As discussed, verifying if a team is accountable for a particular path formula φ is conditioned to whether they were tasked to fulfil φ. Thus, to verify accountability for a task, it is crucial to consider that the allocation process may give a task to more than one team (with the aim to have a level of resiliency). We refer to the number of distinguishable teams that are tasked to fulfil φ by its *degree of resilience* (as a result of introducing redundancy) and denote it by $\mathcal{DR}(\varphi)$. For instance, if the task to deliver the required units of vaccine (φ_D) is allocated to two teams, then $\mathcal{DR}(\varphi_D) = 2$.

Proposition 1. *If Γ_1 and Γ_2 are q-accountable for φ and $\mathcal{DR}(\varphi_D) = 1$, then $\Gamma_1 = \Gamma_2$.*

Proof. The minimality condition for accountability implies that Γ_1 and Γ_2 have no excess members such that one team can be a subset of another one. They can either fully overlap or be distinct teams. Considering that the degree of resilience is 1, the former is the case. □

The proposition shows that accountability is a strong concept as it requires the team to be a *minimal* weakly accountable team of agents. As a corollary we have:

Corollary 1. *If Γ is q-accountable for φ then $\Gamma' \supset \Gamma$ is weakly q-accountable for φ.*

Next, we show that a degree of resilience $\mathcal{DR}(\varphi) = k$ implies having k teams weakly q-accountable for φ based on h if the allocation process is *suitable* in the sense of [Yazdanpanah *et al.*, 2020]. Formally, suitability indicates that if $T^a_{\mathcal{M}}(\Gamma, \varphi, q) = 1$ then $\mathcal{M}, q \models \langle\!\langle \Gamma \rangle\!\rangle \varphi$.

Proposition 2. *Let Γ be a q-accountable team for φ based on h. Given a valid task allocation, if $\mathcal{DR}(\varphi) = k$ then there exist $k - 1$ teams $\Gamma' \neq \Gamma$ weakly q-accountable for φ based on h.*

Proof. The suitability of the allocation process implies that other $k - 1$ teams have a strategy to see to it that φ is the case. The minimality condition cannot be satisfied necessarily as a suitable allocation process may give a task to a team and its super team(s). They posses a strategy to fulfil the task but do not necessarily satisfy the minimality condition. Thus, in accordance to Corollary 1, we have the result on these teams being weakly accountable but not necessarily accountable for φ. □

Note that this result holds even if the teams received the task in question in different states through the history (and not necessarily in the same state). We leave further dynamics of task allocation with the notion of accountability and studying how the coherency aspects of task allocation affects the accountability ascription problem to further research.

Moving to the project level accountability, we have:

Proposition 3. *If Γ is weakly q-accountable for P, then there exist a q-accountable team for all $\varphi \in P$.*

Proof. In case Γ is q-accountable for the project, then it is the unique minimal team and accordingly q-accountable for all the involved tasks. But in case it is a weakly q-accountable team, it has a strategy to fulfil every φ from a point of allocation through the history. However, it is not minimal. For each φ, eliminating excess members guarantees the existence of minimal team. □

3.3 Decidability

In this section, we show that determining if a team is accountable for a task is a decidable problem by proving the following theorem in a constructive way.

Theorem 1. *The accountability verification problem in ATL-modelled multiagent systems is decidable.*

To prove decidability, we give Algorithm 1 that, given a multiagent system model $\mathcal{M}$, a task allocation (accessible via $T^d_{\mathcal{M}}(\Gamma, \varphi, q)$ and $T^a_{\mathcal{M}}(\Gamma, \varphi, q)$), a q-history $h = q_0, \dots, q_l (q = q_l)$, and a task φ (path formula), returns the set of weakly q-accountable teams for φ based on h.

In this procedure, to verify if a team Γ (among the teams that were expected to bring about φ) is accountable, we go through the states of history h and use ATL model-checking from [Alur *et al.*, 2002] to determine whether Γ was capable of fulfilling the task. Note that we are taking a generic approach as the procedure does not rely on a specific degree of resilience and relaxes the assumption that the allocation process was a *suitable* one (in the sense of [Yazdanpanah *et al.*, 2020]).

Next, we present computational complexity results for accountability verification.

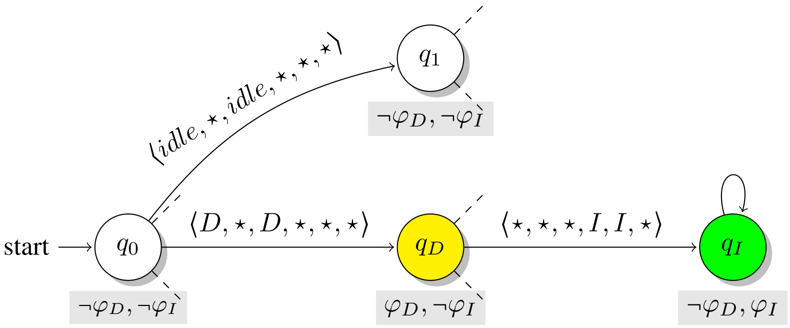
q_1
$\neg\varphi_D, \neg\varphi_I$
$\langle idle, \star, idle, \star, \star, \star \rangle$
start
q_0
$\neg\varphi_D, \neg\varphi_I$
$\langle D, \star, D, \star, \star, \star \rangle$
q_D
$\varphi_D, \neg\varphi_I$
$\langle \star, \star, \star, I, I, \star \rangle$
q_I
$\neg\varphi_D, \varphi_I$

Figure 1: We model the scenario as $\mathcal{M} = \langle \Sigma, Q, \Pi, \pi, Act, d, o \rangle$ where: $\Sigma = \{a_1, \dots, a_6\}$; $Q = \{q_0, q_1, \dots, q_D, q_I\}$; $\Pi = \{\varphi_D, \varphi_I\}$ (φ_D represents the completed delivery of vaccines (yellow states) and φ_I that they are injected (green states)); $Act = \{D, I, idle\}$ to represent two possible actions of delivery (D) and injection (I) as well as inaction (idle); $d(a_i, q) = \{D, idle\}$ ($1 \le i \le 3$) and $d(a_j, q) = \{I, idle\}$ ($4 \le i \le 6$) for all $q \in Q$; and both π and o are as represented in the automaton. For readability, we coloured the states, used $\star$ to refer to any action available to an agent, and used dashed lines to represent other paths that may result form unrepresented actions.

Algorithm 1: Accountability Verification

```
   Input: Model $\mathcal{M}$; $q \in Q$; task $\varphi$, set $T^{d,\varphi}_{\mathcal{M},q}$, and history
          $h = q_0, \dots, q_l (q = q_l)$.
   Result: $Acc^{h,\varphi}_{\mathcal{M},q}$, the set of weakly $q$-accountable
           teams for $\varphi$ based on $h$.
1  $Acc^{h,\varphi}_{\mathcal{M},q} \leftarrow \varnothing$;
2  if $q \notin [\![\varphi]\!]_{\mathcal{M}}$ then
3      forall $\Gamma \in T^{d,\varphi}_{\mathcal{M},q}$ do
4          for $i = 1$ to $l$ do
5              if $\mathcal{M}, q_{l-i} \models \langle\!\langle \Gamma \rangle\!\rangle \varphi$ (standard, see [Alur et
                al., 2002]) then
6                  $Acc^{h,\varphi}_{\mathcal{M},q} \leftarrow Acc^{h,\varphi}_{\mathcal{M},q} \cup \{\Gamma\}$;
7              end
8          end
9      end
10 end
11 return $Acc^{h,\varphi}_{\mathcal{M},q}$;
```

3.4 Complexity

In this section, we establish the complexity of the presented accountability verification (Algorithm 1). One of the main advantages of accountability reasoning using ATL semantics is that its (task) formulae, in turn the process for verifying accountability, can be model-checked in deterministic linear time [Alur *et al.*, 2002; Bulling *et al.*, 2010].

Theorem 2. *Accountability verification in ATL-modelled multiagent systems is P-complete, and can be done in time $\mathcal{O}(l.\mathcal{DR}(\varphi).|\mathcal{M}|.|\varphi|)$, where $|\mathcal{M}|$ is given by the number of transitions in $\mathcal{M}$, l is the length of the history, and $\mathcal{DR}(\varphi)$ is the degree of resilience for φ.*

Proof. The complexity of the model checking part (line 5 in Algorithm 1) is provided by the complexity of model checking ATL [Alur *et al.*, 2002] which is polynomial an can be done in $\mathcal{O}(|\mathcal{M}|.|\varphi|)$. In Algorithm 1, we call this model-checking for every member of $T^{d,\varphi}_{\mathcal{M},q}$ (with the cardinality equal to $\mathcal{DR}(\varphi)$) through all the states of history h with length l. □

And for project-level accountability verification, we have the following result.

Proposition 4. *Verifying if a team Γ is accountable for a project P is $|P|$ times the task complexity (Theorem 2) for verifying the longest $\varphi \in P$.*

This shows a desirable tractability for verifying project-level accountability as it requires $|P|$ calls to Algorithm 1.

4 A Fair Degree of Accountability

Accountability voids are situations in which a task is unfulfilled and a (non-singleton) team of agents or, more problematically, various teams of agents are found to be accountable for it. For instance, imagine that we allocate the task to deliver vaccines (Figure 1) to $\{a_1, a_2\}$ and also to $\{a_1, a_3\}$ (to have a degree of resilience equal to 2 on this task). Then, as the system evolves, if we realise that the vaccines are not delivered, we will have two accountable teams. Then the question is: "to what extent each of the team members are accountable?" As agent a_1 is a member of both of the accountable teams, it seems unreasonable to see all of the three agents equally accountable and ascribe accountability—eventually, potential sanctioning measures or penalties—in a uniform way.

In this section, we present a novel rule-based method, with a tractable complexity, for ascribing accountability.[2] This method corresponds to Marginal Contribution Networks (MC Nets) [Ieong and Shoham, 2005], and in turn provides a computationally tractable way to compute a degree of accountability that satisfies the Shapley-based notion of fairness [Shapley, 1953].

[2] There are recent attempts to address this problem using Shapley-based cost allocation [Yazdanpanah *et al.*, 2019; Friedenberg and Halpern, 2019]. However, such approaches seem operationally infeasible due to non-tractable complexity of the standard method for computing the Shapley value. See more on related work and positioning of our contribution in Section 5.

4.1 A Rule-Based Accountability Ascription

For accountability ascription in multiagent teams, we present a two-phase procedure. For readability, we present these two phases separately (in Algorithm 2 and Definition 2). This separation also allows computing the degree of accountability of each agent in a modular way. This results in a tractable complexity as we do not need to go through all the agents and compute all the degrees. The first phase, takes the randomly indexed set $Acc^{h,\varphi}_{\mathcal{M},q}$ of accountable teams and generates a set of accountability rules.

```
Algorithm 2: Generating Accountability Rules
Input: Randomly indexed $Acc^{h,\varphi}_{\mathcal{M},q}$.
Result: $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$, the set of accountability rules for $\varphi$.
$k \leftarrow |Acc^{h,\varphi}_{\mathcal{M},q}|$;
$\mathfrak{R}^{h,\varphi}_{\mathcal{M},q} \leftarrow \varnothing$;
for $i = 1$ to $k$ do
    $rule \leftarrow \rho_i : (\Gamma_i, \varnothing) \mapsto 1/k$ ;
    $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q} \leftarrow \mathfrak{R}^{h,\varphi}_{\mathcal{M},q} \cup \{rule\}$;
end
return $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$;
```

As a result, this process generates k rules, where k is the size of $Acc^{h,\varphi}_{\mathcal{M},q}$. In Section 4.3, we will show that the presented process generates a rule-based representation of a cooperative game, use this auxiliary game to compute the contribution of individuals to accountable groups, and formulate their individual degree of accountability. In each rule of type $\rho_i : (\mathcal{P}_i, \mathcal{N}_i) \mapsto v_i$, we refer to ρ_i as the title/index of the ith rule, $\mathcal{P}_i$ as the positive set in the rule, $\mathcal{N}_i$ as the negative set in the rule, and v_i as the value of the rule. Intuitively, by assigning $1/k$ to k accountable groups (in each rule ρ_i generated by Algorithm 2), we see all groups in $Acc^{h,\varphi}_{\mathcal{M},q}$ equally accountable while the contribution of agents to such groups is the base for computing their individual degree of accountability (in Definition 2).

Definition 2. *Let $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$ be the set of q-accountability rules generated by Algorithm 2 for φ based on h. For agent $a \in \Sigma$, we say a rule ρ_r is* applicable *if $a \in \mathcal{P}_r$ and by $\omega(a)$ denote the set of rule indices that are applicable to a. Then we say agent a's degree of q-accountability for φ based on h, denoted by $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a)$, is equal to 0 if $\omega(a) = \varnothing$ and $\sum_{r \in \omega(a)} \frac{v_r}{|\mathcal{P}_r|}$ otherwise.*

Analogously, agents degree of accountability for a project is commutable based on the set of accountability rules that is generated for the project in question. In the following, we apply this notion to our vaccination example and present the fairness properties as well as the computational complexity for computing this degree.

4.2 Accountability Ascription in Practice

In the vaccination scenario, in order to ascribe degrees of accountability to agents, we first generate rules that correspond to the set of accountable teams (in $Acc^{h,\varphi}_{\mathcal{M},q}$). In this case, as both $\{a_1, a_2\}$ and $\{a_1, a_3\}$ are accountable, we will have two rules in $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q} = \{\rho_1 : (\{a_1, a_2\}, \varnothing) \mapsto 1/2, \rho_2 : (\{a_1, a_3\}, \varnothing) \mapsto 1/2\}$. Then, for all the agents the second case of Definition 2 applies as they are all a member of a positive set in a rule. Computing the share that each agent gets in its corresponding applicable rules, we have that $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a_1) = 1/2$ and $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a_2) = \mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a_3) = 1/4$.

As observed, this degree is responsive to the larger contribution of a_1 (as it could contribute to two accountable teams) and the symmetric presence of a_2 and a_3 (both contributory to only one accountable teams). Note that although a_2 and a_3 had different delivery capacities, they received similar tasks in this scenario, i.e., to cooperate with a_1 and provide the vaccine unit that a_1 could not deliver.

These desirable properties, generally known as *fairness* properties in the game-theoretic literature, are not specific to this scenario but generally valid for this degree of accountability.

4.3 Fairness Properties

The following Theorem shows that the presented degree of accountability satisfies all the Shapley-based fairness axioms [Shapley, 1953].

Theorem 3. *The presented degree of accountability $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a)$ in Definition 2 guarantees the following fairness axioms: (1) $\sum_{a_i \in \Sigma} \mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a_i) = 1$ (Efficiency); (2) for any $a_i, a_j \in \Sigma$, $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a_i) = \mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a_j)$ if for all $\Gamma \in Acc^{h,\varphi}_{\mathcal{M},q}$ we have that $a_i \in \Gamma \implies a_j \in \Gamma$ (Symmetry); (3) $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a_i) = 0$ if for all $\Gamma \in Acc^{h,\varphi}_{\mathcal{M},q}$, we have that $a_i \notin \Gamma$ (Dummy Player); (4) the summation of a_i's degree of accountability for k different tasks $\varphi_1, \ldots, \varphi_k$ is k times its degree for $\{\varphi_1, \ldots, \varphi_k\}$ (Additivity).*

To prove, we use the following lemmas and show that the presented set of rules constitute a basic Marginal Contribution Net (MC Net) [Ieong and Shoham, 2005] and that the degree corresponds to the Shapley value of agents in this MC Net. Accordingly, we have that our accountability degree satisfies the four axiomatic fairness properties that uniquely characterise the Shapley value.

Lemma 1. *$\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$ is a basic Marginal Contribution Net (MC Net) [Ieong and Shoham, 2005].*

Proof. The set of rules in $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$ correspond to the set-theoretic representation of MC Nets in [Ohta *et al.*, 2009]. □

Note that in each rule, the intersection of the positive set and the negative set is empty by definition ($\mathcal{P}_i \cap \mathcal{N}_i = \varnothing$). This allows relying on a linear computation of the Shapley value of the MC Net.

Lemma 2. *For each $a_i \in \Sigma$, $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a)$ computes the Shapley value of a_i in $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$.*

Proof. In $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$, rules only consist of positive literals [Ieong and Shoham, 2005]. In such an MC Net, Shapley value of each agent is equal to the summation of its Shapley value in all the applicable rules. □

4.4 Complexity

Next, we show the desirable complexity of computing the degree of accountability.

Theorem 4. *The total running time for computing* $\mathfrak{acc}^{h,\varphi}_{\mathcal{M},q}(a)$ *is linear in the size of the input.*

Proof. We show this by first focusing on the computation of the degree itself and then the complexity for generating $\mathfrak{R}^{h,\varphi}_{\mathcal{M},q}$ as its input. To compute the degree of any agent, we are computing its Shapley value in each rule and then apply a summation for all the applicable rules. In MC Nets with positive literals, each agent's Shapely is equal to the value of the rule over the number of members in $\mathcal{P}$ [Ieong and Shoham, 2005]. We have that the upper bound for the number of rules is the degree of resilience of the task. Thus, as the Shapley in a given rule can be computed in time linear in the pattern of the rule, the total running time for computing the degree is linear in the size of the input. And we have that the preceding rule generation process goes through the set of accountable teams which is at most equal to the degree of resilience, thus does not affect the computing time. □

5 Related Work

In relation to past work, in particular to recent work on logic-based responsibility reasoning in multiagent settings [Friedenberg and Halpern, 2019; Yazdanpanah *et al.*, 2019], we focused on verifying the task-based notion of *accountability* (as a specific form of responsibility reasoning) while [Friedenberg and Halpern, 2019] study the epistemic notion of *blameworthiness* and [Yazdanpanah *et al.*, 2019] focus on the *responsibility* of agents with imperfect information. In comparison to these, we assumed perfect information and focused on task dynamics and the task-oriented notion of accountability. As discussed in [Yazdanpanah *et al.*, 2021a], task-based accountability focuses on reasoning about agents that received a task but failed to deliver it while responsibility in its generic from is about the ability to cause or avoid a state of affair and blameworthiness is concerned with agents who not only caused a situation but did it knowingly.

In this work, we used finite histories as a natural choice for modelling the temporally-bounded concept of *task*. This corresponds with the so called *provenance traces* [Tsakalakis *et al.*, 2020], commonly used for reasoning about the reasons behind a materialised situation and providing behaviour-aware explanations. Finally, we share the perspective with [Friedenberg and Halpern, 2019; Yazdanpanah *et al.*, 2019] for applicability of cost sharing methods, such as the Shapley value, for ascribing responsibility (in our case accountability). However, in comparison to the standard Shapley calculation with computationally intractable complexity, our rule-based representation resulted in a computationally tractable method for ascribing accountability degrees. Interestingly, this low-complexity accountability ascription process is also applicable to handle the complexity in imperfect information settings (e.g., in combination with [Yazdanpanah *et al.*, 2019]) as it is a module that comes after the verification of accountable teams, hence does not require any changes to their model-checking under imperfect information.

6 Conclusions

We proposed a formal account of the notion of accountability and presented ATL-based techniques to verify accountability in multiagent settings and ascribe a fair degree of accountability to individual agents. Based on a novel rule-based representation, we developed a *fair* and *computationally tractable* degree for resolving accountability voids in multiagent teams.

The results of this study and developed accountability ascription method also contribute to integrating ethics into AI systems and ensuring their safety and trustworthiness. In particular, as discussed in [Yazdanpanah *et al.*, 2021a], developing computational tools to verify and reason about different forms of responsibility and task-oriented accountability is necessary for design and development of safe and trustworthy AI. Such AI systems are expected to make autonomous decisions and, at the same time, need to make sure that their decisions are in compliance with safety concerns and ethical values. To that end, we need to enrich such systems (and their behaviour monitoring units) with the capacity to contemplate how accountabilities, for potential consequences of such decisions, are to be ascribed. Furthermore, embedding accountability reasoning into AI systems contributes to providing transparency on who is, and to what extent they are, accountable for potentially undesirable behaviour of a given AI-based product [Winfield and Jirotka, 2018].

In this work, we focused on teams of agents assuming their intra-team capacity to coordinate towards the fulfilment of tasks. An interesting extension would be to consider the feasibility of *team formations* based on the value of inter-agent interactions [Beal *et al.*, 2020]. This way, we can reason about feasible team structures in presence of inter-agent incompatibilities and study their dynamics with the accountability ascription problem in multiagent teams.

Acknowledgements

This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through the Trustworthy Autonomous Systems Hub (EP/V00784X/1), the platform grant entitled "AutoTrust: Designing a Human-Centred Trusted, Secure, Intelligent and Usable Internet of Vehicles" (EP/R029563/1), and the Turing AI Fellowship on Citizen-Centric AI Systems (EP/V022067/1).

References

[Abeywickrama *et al.*, 2019] Dhaminda B Abeywickrama, Corina Cirstea, and Sarvapali D Ramchurn. Model checking human-agent collectives for responsible AI. In *Proceedings of Robot and Human Interactive Communication*, pages 1–8, 2019.

[Alechina *et al.*, 2017] Natasha Alechina, Joseph Y. Halpern, and Brian Logan. Causality, responsibility and blame in team plans. In *Proceedings of AAMAS-2017*, pages 1091–1099, 2017.

[Alur *et al.*, 2002] Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-time temporal logic. *J. ACM*, 49(5):672–713, 2002.

[Baldoni *et al.*, 2019] Matteo Baldoni, Cristina Baroglio, Olivier Boissier, Roberto Micalizio, and Stefano Tedeschi. Accountability and responsibility in multiagent organizations for engineering business processes. In *Proceedings of EMAS*, pages 3–24, 2019.

[Baldoni *et al.*, 2020] Matteo Baldoni, Cristina Baroglio, and Roberto Micalizio. Fragility and robustness in multiagent systems. In *Proceedings of EMAS*, pages 61–77, 2020.

[Beal *et al.*, 2020] Ryan Beal, Narayan Changder, Timothy Norman, and Sarvapali Ramchurn. Learning the value of teamwork to form efficient teams. In *Proceedings of AAAI-2020*, volume 34, pages 7063–7070, 2020.

[Belle, 2017] Vaishak Belle. Logic meets probability: Towards explainable ai systems for uncertain worlds. In *IJCAI*, pages 5116–5120, 2017.

[Braham and van Hees, 2011] Matthew Braham and Martin van Hees. Responsibility voids. *The Philosophical Quarterly*, 61(242):6–15, 2011.

[Bulling *et al.*, 2010] Nils Bulling, Jurgen Dix, and Wojciech Jamroga. Model checking logics of strategic ability: Complexity. In *Specification and Verification of Multiagent Systems*, pages 125–159. Springer, 2010.

[Chockler and Halpern, 2004] Hana Chockler and Joseph Y Halpern. Responsibility and blame: A structural-model approach. *JAIR*, 22:93–115, 2004.

[De Giacomo and Vardi, 2015] Giuseppe De Giacomo and Moshe Vardi. Synthesis for LTL and LDL on finite traces. In *Proceedings of IJCAI-2015*. Citeseer, 2015.

[EC: The High-Level Expert Group on AI, 2019] EC: The High-Level Expert Group on AI. Ethics guidelines for trustworthy AI. https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai, 2019. Accessed: 2021-05-01.

[Friedenberg and Halpern, 2019] Meir Friedenberg and Joseph Y Halpern. Blameworthiness in multi-agent settings. In *Proceedings of AAAI-2019*, pages 525–532, 2019.

[Hart, 2008] Herbert Lionel Adolphus Hart. *Punishment and responsibility: Essays in the philosophy of law*. Oxford University Press, 2008.

[Ieong and Shoham, 2005] Samuel Ieong and Yoav Shoham. Marginal contribution nets: a compact representation scheme for coalitional games. In *Proceedings of E-Commerce-2005*, pages 193–202, 2005.

[Jennings *et al.*, 2014] Nicholas R Jennings, Luc Moreau, David Nicholson, Sarvapali Ramchurn, Stephen Roberts, Tom Rodden, and Alex Rogers. Human-agent collectives. *Communications of the ACM*, 57(12):80–88, 2014.

[Kalenka and Jennings, 1999] Susanne Kalenka and Nicholas R Jennings. Socially responsible decision making by autonomous agents. In *Cognition, Agency and Rationality*, pages 135–149. Springer, 1999.

[Lomuscio *et al.*, 2017] Alessio Lomuscio, Hongyang Qu, and Franco Raimondi. MCMAS: an open-source model checker for the verification of multi-agent systems. *Int. J. Softw. Tools Technol. Transf.*, 19(1):9–30, 2017.

[Macarthur *et al.*, 2011] Kathryn Sarah Macarthur, Ruben Stranders, Sarvapali Ramchurn, and Nicholas R. Jennings. A distributed anytime algorithm for dynamic task allocation in multi-agent systems. In *Proceedings of AAAI-2011*, pages 701–706, 2011.

[Miller, 2019] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

[Office for Artificial Intelligence - GOV.UK, 2020] Office for Artificial Intelligence - GOV.UK. A guide to using artificial intelligence in the public sector. https://www.gov.uk/government/publications/a-guide-to-using-artificial-intelligence-in-the-public-sector, 2020. Accessed: 2021-05-01.

[Ohta *et al.*, 2009] Naoki Ohta, Vincent Conitzer, Ryo Ichimura, Yuko Sakurai, Atsushi Iwasaki, and Makoto Yokoo. Coalition structure generation utilizing compact characteristic function representations. In *International Conference on Principles and Practice of Constraint Programming*, pages 623–638, 2009.

[Russell, 2019] Stuart Russell. *Human compatible: Artificial intelligence and the problem of control*. Penguin, 2019.

[Shapley, 1953] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

[Tsakalakis *et al.*, 2020] Niko Tsakalakis, Laura Carmichael, Sophie Stalla-Bourdillon, Luc Moreau, Dong Huynh, and Ayah Helal. Explanations for AI: Computable or not? In *Web Science Companion*, pages 77–77, 2020.

[van de Poel *et al.*, 2012] Ibo van de Poel, Jessica Nihlén Fahlquist, Neelke Doorn, Sjoerd Zwart, and Lamber Royakkers. The problem of many hands: Climate change as an example. *Science and engineering ethics*, 18(1):49–67, 2012.

[van de Poel, 2011] Ibo van de Poel. The relation between forward-looking and backward-looking responsibility. In *Moral responsibility*, pages 37–52. Springer, 2011.

[Winfield and Jirotka, 2018] Alan FT Winfield and Marina Jirotka. Ethical governance is essential to building trust in robotics and artificial intelligence systems. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2018.

[Yazdanpanah *et al.*, 2019] Vahid Yazdanpanah, Mehdi Dastani, Wojciech Jamroga, Natasha Alechina, and Brian Logan. Strategic responsibility under imperfect information. In *Proceedings of AAMAS-2019*, pages 592–600, 2019.

[Yazdanpanah *et al.*, 2020] Vahid Yazdanpanah, Mehdi Dastani, Shaheen Fatima, Nicholas R. Jennings, Devrim Murat Yazan, and W. Henk Zijm. Task coordination in multiagent systems. In *Proceedings of AAMAS-2020*, pages 2056–2058, 2020.

[Yazdanpanah *et al.*, 2021a] Vahid Yazdanpanah, Enrico H. Gerding, Sebastian Stein, Corina Cirstea, m.c. schraefel, Timothy J. Norman, and Nicholas R. Jennings. Collective responsibility in multiagent settings. In *ACM Collective Intelligence Conference 2021 (CI-2021)*, April 2021.

[Yazdanpanah *et al.*, 2021b] Vahid Yazdanpanah, Enrico H. Gerding, Sebastian Stein, Mehdi Dastani, Catholijn M. Jonker, and Timothy J. Norman. Responsibility research for trustworthy autonomous systems. In *Proceedings of AAMAS-2021*, page 57–62, 2021.