Fostering Multi-Agent Cooperation through Implicit Responsibility

Daniel E. Collins[0000−0002−1075−4063], Conor Houghton[0000−0001−5017−9473], and Nirav Ajmeri[0000−0003−3627−097X]

University of Bristol, Bristol, UK
{daniel.collins,conor.houghton,nirav.ajmeri}@bristol.ac.uk

Abstract. For integration in real-world environments, it is critical that autonomous agents are capable of behaving responsibly while working alongside humans and other agents. Existing frameworks of responsibility for multi-agent systems typically model responsibilities in terms of adherence to explicit standards. Such frameworks do not reflect the often unstated, or implicit, way in which responsibilities can operate in the real world. We introduce the notion of *implicit responsibilities*: self-imposed standards of responsible behaviour that emerge and guide individual decision-making without any formal or explicit agreement.
We propose that incorporating *implicit responsibilities* into multi-agent learning and decision-making is a novel approach for fostering mutually beneficial cooperative behaviours. As a preliminary investigation, we present a proof-of-concept approach for integrating *implicit responsibility* into independent reinforcement learning agents through reward shaping. We evaluate our approach through simulation experiments in an environment characterised by conflicting individual and group incentives. Our findings suggest that societies of agents modelling *implicit responsibilities* can learn to cooperate more quickly, and achieve greater returns compared to baseline.

1 Introduction

When tasked with navigating complex social decision-making scenarios alongside humans and other agents, it is important that agents can balance potential incentive conflicts, and find ways to perform their allocated role effectively whilst acting in a manner that is considered responsible and ethical by human standards [4, 11]. Existing works have outlined various facets of responsibility in multi-agent systems (MAS) [12].

Responsibility. A general definition of responsibility, outlined in [12], involves the expectation for an agent or group of agents, A, to realise a future state, φ, of the environment [5, 8].

Explicit Responsibility. Typically, responsibilities are modelled in terms of standards of behaviour that are prescribed "top-down", such as accountability for the fulfilment of allocated tasks or sanctionability for the violation of a social norm [12]. In this paradigm, agents are responsible to the extent that they adhere to an explicit system of rules. Similarly, responsibility can be imposed through explicit agreements or commitments between agents [1, 6]. We group these treatments as *explicit responsibility*, which can always be described by "A is responsible for φ under z", where z represents the explicit source of the responsibility, which may be enforced top-down, agreed upon peer-to-peer, or otherwise entered into knowingly.

Example 1 (Explicit Responsibility). Alice adopts a puppy in the UK. By adopting the puppy, Alice has agreed to an explicit duty of care; they are aware that they are accountable for the welfare of the dog under UK law, and that adopting and subsequently neglecting a dog would violate social convention. If Alice proceeds to neglect the puppy, they may be subject to legal repercussions, or disapproval and alienation from family and friends.

Implicit Responsibility. In contrast to *explicit responsibility*, relatively little attention has been given to aspects of responsibility that emerge without any imposed standards or explicit agreement between parties. Self-imposed responsibilities can play an important role in ethical decision making amongst people. Affective responses to different scenarios and outcomes can reinforce an individual sense of responsibility, motivating subsequent cooperation and altruistic behaviour. Individual differences in these affective responses can give rise to variations in self-motivated responsible behavior between people. Understanding this type of responsibility and how it can lead to alignment and misalignment of individual perceptions of responsibility in society is important for citizen-centric design of MAS. We extend the conceptual framework of *explicit responsibility* in MAS by introducing the notion of *implicit responsibility*: a self-imposed responsibility for bringing about some φ, that emerges bottom-up, and is internally motivated and voluntarily assumed without any explicit mandate, commitment or expectation.

Example 2 (Implicit Responsibility). Alice comes across a stunned pigeon near their home. Alice reasons that the pigeon will likely be in danger if left in its current state, and that they could carefully transfer the pigeon to a cardboard box and leave it to rest in a safe quiet area to recover. Alice is driven to help the pigeon by an internal sense of responsibility, although there is no explicit expectation to do so.

In Example 2, a situation emerges in which Alice feels implicitly responsible for the fate of another entity. Even if Alice does not assume the responsibility for assisting the other entity as a goal, they are nevertheless aware that they are capable of providing that assistance, and the consequences of not doing so. Failure to help may confer a negative affective state, motivating Alice to help in similar scenarios in the future.

Contributions. In this work, we introduce the notion of *implicit responsibility* in MAS. We present a novel approach for promoting cooperation within the framework of multi-agent reinforcement learning (MARL) by operationalising *implicit responsibility* for reward shaping. We investigate our approach by conducting simulation experiments in a constrained task environment designed to incorporate well-defined *implicit responsibilities.* We compare the learning of cooperative behaviour by *implicit responsibility* agents to *baseline* reinforcement learning agents that do not shape rewards. We find that agents that model *implicit responsibility* learnt cooperative strategies faster, and demonstrate improved performance on the task compared to *baseline* agents.

2 Operationalising Implicit Responsibility in MAS

In MARL, reward shaping is the process of modifying an agent's reward function by introducing additional "pseudo-rewards" to guide agents towards learning specific patterns of behaviour that may not be adequately incentivised by the original reward function. Shaping rewards according to violation or satisfaction of *implicit responsibility* provides a novel framework for learning desirable behaviour. For a pair of agents A, B, A has an *implicit responsibility*, $R^t_{A,B}(\varphi_B)$, for realising a future state of the environment, φ_B, if at some time, t, the environment state, s^t satisfies all of three conditions

1. *Existence of Dependency*, $\psi_{A,B}(1)$ - Agent B's ability to achieve their goals in a future state $s \in \varphi_B$ is contingent on the actions or resources of A.
2. *Capability to Influence*, $\psi_{A,B}(2)$ - Agent A possesses the capacity to address the needs of B and bring about φ_B through its actions or resources.
3. *Awareness or Capability of Perception*, $\psi_{A,B}(3)$ - Agent A can perceive or is capable of perceiving conditions (1) and (2) even if B does not communicate this explicitly.

These conditions describe circumstances in which the realisation of some φ_B, in which B can pursue their goals without assistance, is not possible through the actions of B alone, or from the influence of the dynamics of the environment itself.

2.1 Foraging Survival Simulation Environment

We designed a multi-agent grid-world environment that incorporates well-defined opportunities for *implicit responsibilities*, as an evaluation test-bed. The environment is illustrated in Figure 1.

Setup In this environment, a population of agents, I, navigate an M by N grid-world with the goal of collecting berries. Initially, each agent $i \in I$ starts from a random empty position, and $|I|$ berries are placed at random empty positions so that the number of berries is equal to the number of agents.

Agent attributes Agents have two attributes which relate to their survival in the environment: (1) energy and (2) health. These are represented by the integers $e_i \in \mathbb{Z}^+ : e_i \in [0, E]$ and $h_i \in \mathbb{Z}^+ : h_i \in [0, H]$ respectively. Agents are initialised with $e_i = E$ and $h_i = H$.

Attribute decay Agents are in one of three possible states at any time, based on their attributes: (1) *Healthy*: $(e_i > 0, h_i = H)$, (2) *Helpless*: $(e_i = 0, h_i > 0)$, and (3) *Dead*: $(e_i = 0, h_i = 0)$. While agents are *Healthy*, e_i decays by one per time step. When $e_i = 0$, agents become *Helpless*, and h_i begins to decay by one per time step. Agents can only take actions while *Healthy*. If the agent transitions into the *Dead* state, $h_i = 0$, the agent is removed from the simulation for the remainder of the episode.

Berry collection Agents collect berries by moving to their positions. When an agent collects a berry, the agent receives a reward r_b, and a new berry is generated at a random

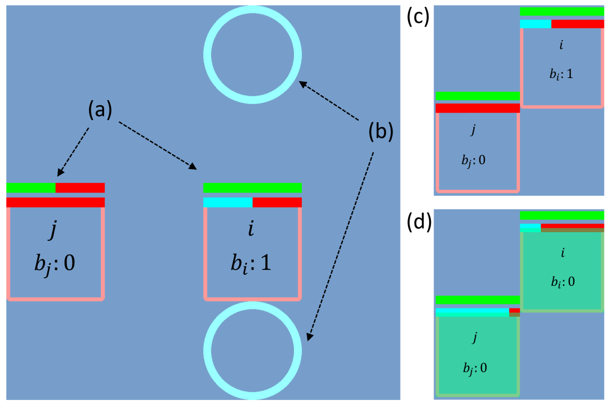

Fig. 1: (Left) Two agents i and j (a) navigate a 4×4 grid world and collect berries (b). The agents health h_i, h_j and energy e_i, e_j are indicated by indicated by the upper and lower bars above the agents respectively, and the number of stored berries is indicated by b_i, bj. (Right) (c) In the illustrated scenario, j has $e_j = 0$, and no stored berries, and i has $e_i > 0$ and one stored berry. (d) In the next time step, i throws their stored berry to j, illustrated by the green shading, and e_j is restored.

unoccupied position. If an agent dies, the next berry collection will not trigger a new berry to be generated. This ensures that there is only one berry per living agent in the environment.

Berry inventory Agents *store* collected berries in an inventory. The number of stored berries is $b_i \in \mathbb{Z}^+ : b_i \in [0, B]$, where B is the inventory capacity.

Berry consumption Agents *consume* stored berries to fully restore e_i and h_i. If an agent has $b_i > 0$ when $e_i = 0$, the agent automatically *consumes* a stored berry. Agents therefore have an *effective energy* of $e'_i = e_i + E * b_i$.

Agent actions Agents have five discrete movement actions for navigating the environment:*up*, *down*, *left*, *right*, and *stay*. Additionally, agents have a *throw* action which passes a stored berry to the agent, j, with the lowest *effective energy*, e'_j. If $b_i = 0$, or if all other agents are *dead*, the *throw* fails and the berry remains in the agents inventory. If an agent successfully throws a berry, their energy does not decay in that time step.

Decision module Agents automatically *consume* a berry if: (1) $h_i < H$ and $b_i > 0$ at the start of a time step, (2) $h_i < H$ and i has just been passed a berry by another agent, or (3) $b_i = B$ and i has just collected a new berry.

Agents have an immediate incentive to act in self-interest by collecting berries as quickly as possible. However, the *Throw* mechanic allows *Healthy* agents to cooperate by paying a cost to revive *Helpless* agents and prevent their death. We can introduce a long-term incentive for mutual cooperation which outweighs the immediate incentive for self-interest through careful choice of environment parameters, (M, N), E, H, B and $|I|$. In Appendix B, we choose environment parameters for our experiment such that mutual cooperation can facilitate longer survival times, and thus greater overall returns.

2.2 Reward Shaping using Implicit Responsibility Conditions

We now apply the conditions described in Section 2 for formation of *implicit responsibility* to our environment. For two agents $i, j \in I$, let φ_j be the set of states in which j is *Healthy*, such that $s_j^t \in \varphi_j$ if $h_j^t = H$. $R_{i,j}^t(\varphi_j) = R_{i,j}^t$ then describes whether i has an *implicit responsibility* towards j at time t for realising φ_j if all three conditions (*Existence of Dependency*, *Capability to Influence*, *Awareness or Capability of Perception*) are met.

For our environment, the condition $\psi_{i,j}^t(1)$ for *Existence of Dependency* is true if j has no energy or berries, but is not yet *Dead*.

$$\psi_{i,j}^t(1) = \begin{cases} 1, & \text{if } e_j^t = 0, \text{ AND } b_j^t = 0, \text{ AND } h_j^t > 0 \\ 0, & \text{otherwise} \end{cases}$$

The condition $\psi_{i,j}^t(2)$ for *Capability to Influence* is true if i has enough energy and berries to throw one to j, and i will not run out of energy as a result of the throw. Let ω_i^t be the *Spare Effective Energy* of i at t, e.g. the effective energy of i that would remain after throwing a berry, $\omega_i^t = e_i^t + E \cdot (b_i^t - 1)$. Let k_i^t be the shortest Manhattan distance between i and any berry at t. If $k_i^t < \omega_i^t$, i can throw a berry and have enough energy remaining to reach another.

$$\psi_{i,j}^t(2) = \begin{cases} 1, & \text{if } k_i^t < \omega_i^t \\ 0, & \text{otherwise} \end{cases}$$

For $\psi_{i,j}(3)$, *Awareness or Capability of Perception*, we assume full-observability of the environment for all agents, therefore i always has sufficient information to know if $\psi_{i,j}(1)$ and $\psi_{i,j}(2)$ are true, thus $\psi_{i,j}(3)$ is true by default.

Once formed, an *implicit responsibility* is maintained until the next time step in which any of the individual conditions are broken. If a responsibility is formed at a time t and maintained until any condition is broken at some later time t', the responsibility is violated if the state $s^{t'}$ does not belong to φ_j. Otherwise, if $s^{t'} \in \varphi_j$, the responsibility is satisfied. Algorithm 1 in Appendix A describes our method for shaping rewards by applying penalties, p, for violating an *implicit responsibility*.

3 Simulation Experiments

We conduct preliminary simulation experiments using the environment described in Section 2.1 with the parameters outlined in Appendix B, Table 1. We simulate and compare societies comprising pairs of agents, which are trained using Deep Q-Learning as described in Appendix C, with hyper-parameters in Table 2. We train a *baseline* agent society using only extrinsic rewards signals from berry collection, and an *implicit responsibility* agent society using both extrinsic rewards and additional penalties for violation of *implicit responsibilities* using our reward shaping algorithm (Section A, Algorithm 1). To evaluate our *implicit responsibility* agents, we compare the length of each episode during training to those achieved by *baseline* agents. Episode length tell us the total survival time of an agent society, indicating the performance of the agents during training.

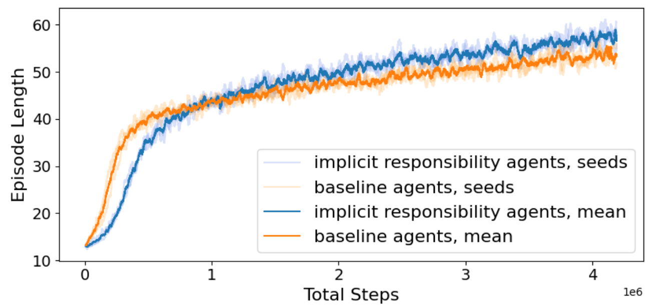

Fig. 2: Episode length (moving average, window size = 1000) vs total environment steps elapsed during training. The mean across three random seeds is shown alongside each individual seed.

4 Discussion

Figure 2 shows the training curves for *baseline* agents and *implicit responsibility* agents across three random seeds. For each episode during training, the episode length is plotted against the the total number of time steps that have elapsed prior to the episode during training. In the early stages of training, *baseline* agents achieve greater survival times than *implicit responsibility* agents. However, after roughly 10^6 steps, *implicit responsibility* agents demonstrate greater survival times on average. These results are a promising indication that shaping rewards according to *implicit responsibility* can improve the speed at which reinforcement learning agents learn to exploit mutually beneficial cooperation behaviours. However, there are several limitations which must be addressed. Firstly, we only evaluate under one set of environment parameters and learning hyper-parameters. It is possible that the benefits of our approach are less significant when we compare to baseline under an optimised training protocol, or in societies of more than two agents. Further experimentation would be needed to validate our findings and assess scalability.

Further, we only test in one environment, which we designed to include easily defined scenarios for *implicit responsibility* to arise, and in which cooperation is globally beneficial. In doing so, we were able to test our approach by shaping rewards according to rules representing an idealised and thus *explicit* model of *implicit responsibility* for that environment. For application to unseen and more complex environments, agents must be designed such that they are able to approximate these rules independently. Causal attribution of responsibility and blameworthiness for outcomes are non-trivial problems [9, 12], posing a challenge for reward function design.

Finally, we consider only a subset of *implicit responsibilities* that capture mutually beneficial outcomes, and thus neglects the role of altruism captured by other approaches for bottom-up learning of responsible behaviour [2, 3, 10].

Bibliography

[1] Dastani, M., van der Torre, L., Yorke-Smith, N.: Commitments and interaction norms in organisations. Autonomous Agents and Multi-Agent Systems (JAAMAS) **31**(2), 207–249 (Mar 2017)

[2] Deshmukh, J.: Emergent responsible autonomy in multi-agent systems. In: Proc. AAMAS. pp. 3029—3031. (May 2023)

[3] Deshmukh, J., Adivi, N., Srinivasa, S.: Resolving the dilemma of responsibility in multi-agent flow networks. In: Proc. PAAMS. pp. 76–87. (Jul 2023)

[4] Murukannaiah, P.K., Ajmeri, N., Jonker, C.M., Singh, M.P.: New foundations of ethical multiagent systems. In: Proc. AAMAS. pp. 1706–1710. (May 2020)

[5] van de Poel, I.: The Relation Between Forward-Looking and Backward-Looking Responsibility. In: Moral Responsibility: Beyond Free Will and Determinism, pp. 37–52. Library of Ethics and Applied Philosophy, Springer (2011)

[6] Singh, M.P.: Norms as a basis for governing sociotechnical systems. ACM Transactions on Intelligent Systems and Technology (TIST) **5**(1), 21:1–21:23 (Dec 2013)

[7] Szita, I., Lőrincz, A.: The many faces of optimism: a unifying approach. In: Proc. ICML. pp. 1048–1055. ACM (2008)

[8] Triantafyllou, S.: Forward-Looking and Backward-Looking Responsibility Attribution in Multi-Agent Sequential Decision Making. In: Proc. AAMAS. pp. 2952–2954 (May 2023)

[9] Triantafyllou, S., Radanovic, G.: Towards Computationally Efficient Responsibility Attribution in Decentralized Partially Observable MDPs. In: Proc. AAMAS. pp. 131–139. (May 2023)

[10] Wang, J.X., Hughes, E., Fernando, C., Czarnecki, W.M., Duenez-Guzman, E.A., Leibo, J.Z.: Evolving intrinsic motivations for altruistic behavior (Mar 2019), arXiv:1811.05931 [cs]

[11] Woodgate, J., Ajmeri, N.: Macro ethics for governing equitable sociotechnical systems. In: Proc. AAMAS. pp. 1824–1828. (May 2022).

[12] Yazdanpanah, V., Gerding, E.H., Stein, S., Cirstea, C., Schraefel, M.C., Norman, T.J., Jennings, N.R.: Different Forms of Responsibility in Multiagent Systems: Sociotechnical Characteristics and Requirements. IEEE Internet Computing **25**(6), 15–22 (Nov 2021)

[13] Zhu, Z., Hu, C., Zhu, C., Zhu, Y., Sheng, Y.: An Improved Dueling Deep Double-Q Network Based on Prioritized Experience Replay for Path Planning of Unmanned Surface Vehicles. Journal of Marine Science and Engineering **9**(11), 1267 (Nov 2021)

A Reward Shaping Algorithm

Algorithm 1 Reward shaping for *implicit responsibility* agents

Let i, j be any pair of agents from a population I.

Let b_i^t be the number of berries that i has stored in their inventory at time t, where $0 \leq b_i^t \leq B$ and $b_i^t, B \in \mathbb{Z}^+$

Let e_i^t be the energy of i at t, where $0 \leq e_i^t \leq E$ and $e_i^t, E \in \mathbb{Z}^+$

Let h_i^t be the health of i at t, where $0 \leq h_i^t \leq H$ and $h_i^t, H \in \mathbb{Z}^+$

Let $d_{i,j}^t$ be the Manhattan distance between i and j at t.

Let k_i^t be the shortest Manhattan distance between i and any berry at t.

Let ω_i^t be the *Spare Effective Energy* of i at t, where $\omega_i^t = e_i^t + E \cdot (b_i^t - 1)$

Let s^t represent the full environment state at time t.

Let r_i^t be the reward to i at time t.

Let p be the constant representing the penalty for violation of an *implicit responsibility*.

Let φ_j be the set of states in which j is *Independent*, such that $s_j^t \in \varphi_j$ if $h_j^t = H$.

Let $\psi_{i,j}^t(1)$ describe the condition for the *Existence of Dependency* such that

$$\psi_{i,j}^t(1) = \begin{cases} 1, & \text{if } e_j^t = 0, \text{ AND } b_j^t = 0, \text{ AND } h_j^t > 0 \\ 0, & \text{otherwise} \end{cases}$$

Let $\psi_{i,j}^t(2)$ describe the condition for *Capability to Influence* such that

$$\psi_{i,j}^t(2) = \begin{cases} 0, & \text{if } k_i^t > \omega_i^t \\ 1, & \text{otherwise} \end{cases}$$

Let $R_{i,j}^t$ be the bool representing whether i has an *implicit responsibility* towards j at time t

$$R_{i,j}^t = \begin{cases} True, & \text{if } \psi_{i,j}^t(1) = 1, \text{ AND } \psi_{i,j}^t(2) = 1 \\ False, & \text{otherwise} \end{cases}$$

// Iterate over all permutations of agent pairs $i, j \in I$

for $i \in I$ **do**

    for $j \in I : j \neq i$ **do**

        // If i was responsible before but not after the transition ...

        if $R_{i,j}^t$ AND $\neg \mathrm{R}_{i,j}^{t+1}$ **then**

            //... and if j has not reached φ_j

            if $\neg(\mathrm{s}^{t+1} \in \varphi_j)$ **then**

                // Apply penalty for violation

                $r_i^{t+1} = r_i^{t+1} - p$

B Environment Parameters

For an (M, N) grid with population $|I|$, if we do not allow agents to use the *Throw* action, and if E is less than some threshold, E^*, the energy of each agent will on average decay towards zero each time step, and all agents will eventually die even with an optimal coordinated foraging strategy. For our environment, we estimate E^* to be the average Manhattan distance between any agent and their closest berry for all possible combinations of positions of $|i|$ agents and $|I|$ berries. In practice, E^* will be slightly lower since the optimal foraging strategy would also ensure that no two or more agents target the same berry at any time. By allowing agents to *Throw* berries, the population can cooperate to survive for longer and thus achieve greater overall returns. For our experiments, we use the environment parameters shown in Table 1.

Table 1: Default environment parameters.

Parameter	**Default Value**
Grid Shape (M, N)	(4, 4)
Population Size $\|I\|$	2
Max Energy E	2
Max Health H	6
Inventory Capacity B	10
Berry Reward r_b	0.1
Violation Penalty p	-0.9

C Agent Architecture and Hyperparameters

Here we describe a schematic of the modular architecture used for our *baseline* and *implicit responsibility* agents. In our experiments, both *baseline* and *implicit responsibility* agents are trained using using independent Deep Q learning implemented with PyTorch. Agents comprise a Deep Q-Network (DQN) architecture with with two fully connected layers. We employ experience replay [13] to stabilise the learning process. Agents explore their shared environment using an epsilon-greedy [7] exploration strategy with exponential decay. Table 2 lists the hyper-parameters of the learning procedure.

Table 2: DQN hyperparameters.

Hyperparameter	**Value**
Batch Size	64
Replay Buffer Capacity	10 000
Discount Factor	0.99
Initial Exploration Rate	0.9
Final Exploration Rate	0.005
Exploration Steps	1000
Tau	0.005
Learning Rate	0.001
Loss Function	MSE
Target Network Update Frequency	500