ℵ-IPOMDP: Mitigating Deception in a Cognitive Hierarchy with Off-Policy Counterfactual Anomaly Detection

NITAY ALON*, Department of Computational Neuroscience Max Planck Institute for Biological Cybernetics, Germany, School of Computer Science and Engineering The Hebrew University of Jerusalem, Israel
JOSEPH M. BARNBY, Centre for AI and Machine Learning Edith Cowan University, Australia, Institute of Psychiatry, Psychology and Neuroscience King's College London, UK
STEFAN SARKADI, Centre for Defence and Security Artificial Intelligence University of Lincoln, UK
LION SCHULZ, Department of Computational Neuroscience Max Planck Institute for Biological Cybernetics, Germany
JEFFREY S. ROSENSCHEIN, School of Computer Science and Engineering The Hebrew University of Jerusalem, Israel
PETER DAYAN, Department of Computational Neuroscience Max Planck Institute for Biological Cybernetics, Germany, Department of Computer Science University of Tüebingen, Germany

Social agents with finitely nested opponent models are vulnerable to manipulation by agents with deeper recursive capabilities. This imbalance, rooted in logic and the theory of recursive modelling frameworks, cannot be solved directly. We propose a computational framework called ℵ-IPOMDP, which augments the Bayesian inference of model-based RL agents with an anomaly detection algorithm and an out-of-belief policy. Our mechanism allows agents to realize that they are being deceived, even if they cannot understand how, and to deter opponents via a credible threat. We test this framework in both a mixed-motive and a zero-sum game. Our results demonstrate the ℵ-mechanism's effectiveness, leading to more equitable outcomes and less exploitation by more sophisticated agents. We discuss implications for AI safety, cybersecurity, cognitive science, and psychiatry.

JAIR Associate Editor: Davide Grossi

JAIR Reference Format:
Nitay Alon, Joseph M. Barnby, Stefan Sarkadi, Lion Schulz, Jeffrey S. Rosenschein, and Peter Dayan. 2026. ℵ-IPOMDP: Mitigating Deception in a Cognitive Hierarchy with Off-Policy Counterfactual Anomaly Detection. *Journal of Artificial Intelligence Research* 85, Article 14 (February 2026), 30 pages. DOI: 10.1613/jair.1.19204

*Corresponding Author.

Authors' Contact Information: Nitay Alon, ORCID: 0000-0002-2698-3573, nitay.alon@mail.huji.ac.il, Department of Computational Neuroscience Max Planck Institute for Biological Cybernetics, Tüebingen, Germany, and School of Computer Science and Engineering The Hebrew University of Jerusalem, Jerusalem, Israel; Joseph M. Barnby, ORCID: 0000-0001-6002-1362, Centre for AI and Machine Learning Edith Cowan University, Perth, Australia, and Institute of Psychiatry, Psychology and Neuroscience King's College London, London, UK; Stefan Sarkadi, ORCID: 0000-0003-3999-528X, Centre for Defence and Security Artificial Intelligence University of Lincoln, Lincoln, UK; Lion Schulz, ORCID: 0000-0003-1841-1273, Department of Computational Neuroscience Max Planck Institute for Biological Cybernetics, Tüebingen, Germany; Jeffrey S. Rosenschein, ORCID: 0000-0002-4042-9739, School of Computer Science and Engineering The Hebrew University of Jerusalem, Jerusalem, Israel; Peter Dayan, ORCID: 0000-0003-3476-1839, Department of Computational Neuroscience Max Planck Institute for Biological Cybernetics, Tüebingen, Germany, and Department of Computer Science University of Tüebingen, Tüebingen, Germany.

DOI: 10.1613/jair.1.19204

1 Introduction

Deception is a constant in human and animal cultures. Humans use a range of deceptive techniques, from "white lies" to malicious and harmful manipulation, misdirecting the beliefs of others for their benefit. To manipulate, a deceiver needs to both create false beliefs and avoid disclosing the deceiver's true intentions. Agents can achieve this through perspective taking, a form of Theory of Mind (ToM; Premack and Woodruff 1978).

ToM encompasses the capacity to simulate others' actions and beliefs. This can be shallow, e.g., observational learning, or recursive (i.e., including the other's capacity to simulate the self, and so forth). The degree to which an agent can use recursive beliefs is known as its depth of mentalising (DoM; Barnby et al. 2023; C. D. Frith and U. Frith 2021). This property has been explained through formal models. A successful and popular framing is k-level hierarchical ToM (Camerer et al. 2004). K-level ToM predicts that agents with lower DoM are formally incapable of making accurate inferences about the intentions of those with higher DoM (Gmytrasiewicz and Doshi 2005). Such an ability would suggest that agents had circumvented the paradox of self-reference. This limitation, found in all recursive modelling frameworks (Pacuit and Roy 2017), implies that agents with low DoM are doomed to be manipulated by others with higher DoM. This asymmetry has previously been explored (Alon, Schulz, Dayan, et al. 2023; Hula, Vilares, et al. 2018; Ş. Sarkadi, Panisson, et al. 2019; Ş. Sarkadi, Rutherford, et al. 2021; Weerd et al. 2022), illustrating the various ways that higher DoM agents can take advantage of lower DoM interaction partners.

While these agents are necessarily disadvantaged, all is not lost. Low DoM agents may still notice that the behaviour they *observe* is inconsistent with the behaviour they *expect*, even if they lack the knowledge to understand how or why (Hula, Vilares, et al. 2018). This type of mismatch warns the victim that they are facing an unmodeled opponent, meaning they can no longer use their simulation of an other for optimal planning.

One way to combat this apparent model failure is to switch to an out-of-belief (OOB) policy, where actions are detached from inferences about their opponent. One path from this is to switch from exploit to exploration behaviour, opening the way to the development of a new opponent model. However, if losing carries a detrimental impact, another option may be to take defensive action, choosing to avoid or reject an environment.

In this work, we present a computational framework for multi-agent RL (MARL) called ℵ-IPOMDP. We augment the well-known IPOMDP approach (Gmytrasiewicz and Doshi 2005) to allow agents to engage with unmodeled opponents. We first discuss how agents can use higher DoM to manipulate others, preying on the limited modelling capacity of their partners. We then present the main contribution of this work: a deception detection mechanism, the ℵ-mechanism, by which limited agents with shallow DoM use an off-policy counterfactual mechanism to detect anomalous behaviour, indicating that they are being deceived, and the OOB ℵ-policy, aimed at reasonably responding to the unknown opponent. We illustrate this mechanism in two game environments, mixed-motive and zero-sum Bayesian repeated games, to show how ℵ-IPOMDP agents can mitigate the advantages of agents with deeper recursive models, and successfully deter manipulation.

Our work is relevant to multiple fields. To the MARL community, we show how agents with limited opponent modelling (for example, suffering from bounded rationality) can partially cope with more adequate opponents via anomaly detection and game-theoretic principles. To the cybersecurity community, we present a MARL masquerading detection use case which can be used to overcome learning adversarial attacks (Rosenberg et al. 2021). To the psychology community, we provide an insights into how humans may use heuristics to detect deception when high DoM resources are not available, and why humans can avoid deception even in the absence of complex recursive reasoning. Given this framework, we can also show how human anomaly detection mechanisms may become maladaptively sensitive and overestimate deception in social environments without the need for over-mentalising, providing a context-appropriate model of suspicious or conspiratorial thinking. Lastly, there has recently been substantial interest in deception in AI (Masters, Smith, et al. 2021; Ş. Sarkadi, Panisson, et al.

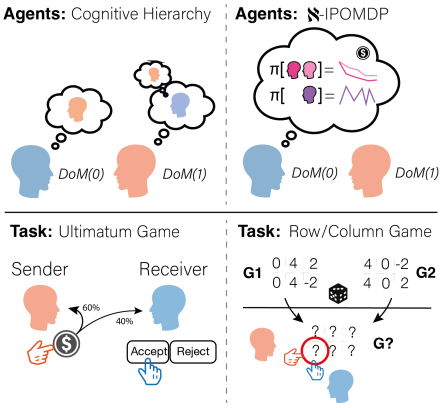

Fig. 1. **Paper overview**: **(Cognitive Hierarchy:)** We model agents with finite recursive opponent modelling with different Depth of Mentalising (DoM). In the classic model, the player at DoM(0) is at the mercy of the DoM(1) partner given that the DoM(0) player cannot form nested beliefs about their opponent. **(ℵ-IPOMDP:)** The DoM(0) can overcome its recursive limitations by augmenting the classic model. We augment agents' inference processes with an anomaly detection mechanism that allows a self to detect deceptive others by matching expectations with observations. **(IUG:)** Agents with different degrees of DoM interact in the iterated ultimatum game (IUG). In the IUG, on each trial, a Sender offers a split of endowed money, and the Receiver decides whether to accept. If the Receiver decides to reject the offer, both players get 0. **(Row/column game:)** Agents interact in an iterated Bayesian zero-sum game. In this game, nature selects a payoff matrix (G^1 or G^2). Only the row player knows which payoff matrix is sampled and uses this information to their advantage. The column player makes inferences about the payoff matrix from the row player's behaviour.

2019; Ş. Sarkadi, Rutherford, et al. 2021; Savas et al. 2022), for which our work may serve as a blueprint for systems that regulate and possibly prevent AI agents from deceiving other AI or human agents.

1.1 Previous Work

The recursive structure of ToM implies that agents can not make correct inferences about other agents with DoM levels that are higher than their own. Modelling a more sophisticated agent would require a self to model itself from the perspective of another—outside of the DoM hierarchy— and thus, at least in principle, to violate key logical principles (Pacuit and Roy 2017). Such a restriction is not unique to ToM-based models but is evident in general bounded rationality environments (Nachbar and Zame 1996). The relation between deception and recursive ToM has been explored by Oey et al. (2023); in their work, however, agents are aware of the fact the others may deceive them and hence decide whether the others are lying or not. Nevertheless, humans take action under the belief that others are being deceptive, even in the absence of a fully fledged understanding of why. In our work, we consider the case in which agents cannot reason about the possibility of deceptive behaviour, but they can detect a deviation from *expected* behaviour. This is somewhat analogous to off-the-equilibrium-path in Perfect Bayesian Equilibrium (PBE) — certain behaviours (policies) should not appear in the interaction. If unexpected behaviour is observed, then it signals to the agent that something in the assumptions about behaviours is wrong.

The problem of interacting with unmodeled others was introduced in ad-hoc teamwork (Mirsky et al. 2022; Stone et al. 2010). In a cooperative setting (Hu et al. 2021), an agent needs to learn a policy (behaviour) to interact successfully with other agents. ToM allows successful interaction with others (Weerd et al. 2022). It is, however, assumed that the others are properly modelled by the acting agent.

Information-theoretic methods have previously been used for both deception modelling and deception detection. For example, Kopp et al. (2018) introduced a deception modelling framework to study deceptive diffusion along

with cooperation in populations exposed to fake news. Others have looked at detecting masquerading by inferring deception in the context of intrusion detection. For example, Evans et al. (2007) used the MDLCompress algorithm to identify intruders disguised as legitimate users. Maguire et al. (2019) suggests that humans apply a typical set-like mechanism to identify a non-random pattern. However, this is very specific to random behaviour. In this work, we advance this concept to explore several types of deviation away from expected behaviour.

Behaviour-based Intrusion Detection Systems (IDS) methods were proposed by Pannell and Ashman (2010) and Peng et al. (2016). In these systems, the system administrator monitors the behaviour of users to decide and respond to malicious behaviour. However, unlike our proposed method, these systems often require labelled data, making them susceptible to an aware adversary who knows how to avoid detection.

In the context of POMDP, several goal recognition methods have been proposed, for example (Le Guillarme et al. 2016; Ramírez and Geffner 2011). While these methods assume that the observed agent may be malign, they assume that (a) the observer can invert the actions to make inferences about the malign intent, and (b) they use the likelihood of the observed behaviour to recognise this intent. Here, we show how likelihood-based inference is used *against* the inferring agent, and propose a mechanism that detects deviation from expected behaviour, flagging malevolent agents without making inferences about their goals.

Yu et al. (2021) explores how an agent might adapt to higher DoM opponents by learning the best response from experience. However, their mechanism lacks a model to detect when the opponent's DoM level exceeds the agent's DoM level, which is a necessary step to prompt the agent to retrain its model. On the other hand, (Piazza et al. 2023) introduces a communicative MARL framework and shows that using ToM to defend against deceptive adversaries has several limitations that are determined by the multi-agent context in which communication is exchanged. For instance, the authors show that deceivers thrive in MARL settings where they can leverage biases such as static play, learning biases, and irrationality. Moreover, their model assumes that the victim can utilise their ToM to make inferences about the intention of the deceiver—distinguishing deceptive behaviour from non-deceptive (up to some limitation described in the paper). In this work we assume that the victim is deceived by an opponent with greater ToM capacity, depriving the victim of the ability to use its ToM to detect deception.

1.2 Theory of Mind in Multi-agent RL

In a partially observed single-agent RL problem (POMDP; (Kaelbling et al. 1998)), an agent is uncertain at time t about the exact state of the environment, $s^t \in S$. It performs an action $a^t \in A$ and receives an observation $o^t \in O$ which depends on both of these: $p(o^t|a^t, s^t)$. Assume that the dynamics of the environment are known to the agent (as we consider model-based RL) and that they are governed by a transition probability: $p(s^t|s^{t-1}, a^{t-1}) \equiv T(s^t, a^{t-1}, s^{t-1})$. The agent makes an inference about the unknown state from the history, $h^{t-1} = \langle a^0, o^0, \ldots, a^{t-1}, o^{t-1}\rangle$, in a Bayesian manner: $b(s^t) = p(s^t|h^{t-1})$. This can be expressed recursively as:

$$b(s^t) = p(s^t|h^{t-1}) \propto P(o^{t-1}|a^{t-1}, s^{t-1}) \sum_{s^{t-1}\in S} T(s^t, a^{t-1}, s^{t-1}) b(s^{t-1}) \tag{1}$$

The agent's utility is a function of the environment and its actions: $u : A \times S \rightarrow \mathbb{R}$. An agent's goal is to maximize its long-term discounted utility: $\sum_{t=1}^{T} u^t \exp[t \log(\gamma)]$, where $\gamma \in [0, 1]$ is the discount factor. In a POMDP, the optimal action-value function takes a particular action, and reports the sum of the expected immediate utility for that action at a belief state, and the expected long run discounted utility if the best possible action is subsequently taken at each subsequent step. It is defined recursively as:

$$Q^*(a^t, b(s^t)) = E_{s^t\sim b(s^t)}\Big[u^t(a^t, s^t) + \gamma \sum_{o^t\in O} P(o^t|a^t, s^t) \times \sum_{s^{t+1}\in S} T(s^{t+1}, a^t, s^t) \max_{a^{t+1}} \{Q^*(a^{t+1}, b(s^{t+1}))\}\Big] \tag{2}$$

The agent's policy is either a deterministic or stochastic function of these Q-values. One such stochastic policy is the SoftMax policy—a distribution over the actions with a known inverse temperature $\mathcal{T}$:

$$\pi(a^t|b(s^t)) = P(a^t|b(s^t)) \propto \exp \frac{Q^*(a^t, b(s^t))}{\mathcal{T}} \tag{3}$$

In this work we assume that agents follow this policy.[1]

In multi-agent RL (MARL), multiple RL agents interact with the same environment. We call the agents μ and ν, and index their actions accordingly.[2] The interaction implies that the environment changes as a function of other agents' actions as well $p(s^t|a_\mu^{t-1}, a_\nu^{t-1}, s^{t-1}) \equiv T(s^t, a_\mu^{t-1}, a_\nu^{t-1}, s^{t-1})$. Since the agent's utility is a function of the environmental state, it now becomes a function of other agents' actions, too. We highlight this dependency via reformulation of μ's utility: $u_\mu : A_\mu \times A_\nu \times S \to \mathbb{R}, u_\mu^t = f(a_\mu^t, a_\nu^t, s^t)$.

Given this coupling, agents in MARL are motivated to predict the behaviour of others (Wen et al. (2019)). Driven by reward maximisation, and since its reward depends on ν's behaviour, μ may *simulate* ν's reaction to μ's action and compute the expected utility of that action:

$$E_{a_\nu^t}[u_\mu^t(a_\mu^t, a_\nu^t)] = \sum_{a_\nu^t} \hat{P}(a_\nu^t|a_\mu^t) u_\mu^t(a_\mu^t, a_\nu^t) \tag{4}$$

Where $\hat{P}(a_\nu^t|a_\mu^t)$ is μ's simulation of ν. Thus, given a policy, π_μ, agent μ may utilise this simulator to compute the discounted cumulative expected utility of this policy:

$$E(u_\mu|\pi_\mu) = \sum_{t=0}^{T} E_{a_\nu^t}[u_\mu^t(a_\mu^t, a_\nu^t)] \exp[t \log(\gamma)] \tag{5}$$

This is conceptually the mechanism governing the standard Nash equilibrium (NE). However, NE requires common knowledge (implying an arbitrarily deep level of nested knowledge) of the world and of others' behaviour (Chwe 2013). This assumption is easily revoked in multiple social interactions, where agents typically hold private and public information.

Bayesian games (Zamir 2020) model incomplete information encounters and are used to model social interaction. Formally, in a Bayesian game, agents do not have full information about the environment. This uncertainty may concern the state of the world ("Is there milk in the refrigerator?"), the intentions of other agents ("Friend or Foe?") (Littman et al. 2001) or another's modus operandi ("how will Joe react to me doing this?"). Beliefs about unknown variables govern an agents' decision-making and are updated in a Bayesian manner, in a way similar to a single-agent POMDP. These beliefs may include beliefs about other agents' beliefs—a distribution over distributions (Silva et al. 2024). Furthermore, agents can form beliefs about these recursive beliefs, i.e., a distribution over the distribution of distributions.

This recursive reasoning is known as Theory-of-Mind (ToM). ToM is the ability to ascribe intentionality to others. This trait is considered one of the hallmarks of human cognition (Premack and Woodruff 1978), and has been suggested to be vital to complex human behaviour, from effective communication (Frank and Goodman 2012; Goodman and Frank 2016) to deception (Alon, Schulz, Rosenschein, et al. 2023a; Ş. Sarkadi, Panisson, et al. 2019). Due to the pivotal role ToM plays in the human capacity to interact socially, it has elicited interest from the AI community (Cuzzolin et al. 2020; Q. Wang et al. 2021).

The IPOMDP framework (Gmytrasiewicz and Doshi 2005) combines ToM and POMDP, allowing us to model and solve these recursive beliefs models as a POMDP. While other models formalise recursive ToM (DoM), such as CHASE (Buergi et al. 2024), hypergames (Bennett 1980) or the aforementioned RSA model (Goodman and Frank 2016), the IPOMDP generalises each of these frameworks into a flexible process that can be adapted to

[1] Acknowledging the discrepancy with the definition of the optimal Q^* values.

[2] We discuss two agents, but the same principles apply to a larger number.

several social contexts. Naturally, while the principles presented in this work may be incorporated into any existing DoM model, we opted for the IPOMDP for its generality and wide use (Alon, Schulz, Dayan, et al. 2023; Hula, Vilares, et al. 2018; Rusch et al. 2020). In this model, an agent's *type* (Harsanyi 1968; Westby and Riedl 2022), θ_μ, is a combination of the agent's *persona* (utility function, sensor capacity, etc.), denoted by ψ_μ, and the agent's *beliefs* about the world $b_\mu(\cdot)$: $\theta_\mu = \langle \psi_\mu, b_\mu(\cdot) \rangle$. It is assumed that agents are fully Bayesian, meaning that their uncertainty is epistemic and can be reduced to correctly identify the *type* of agents with a lower DoM. Misplaced beliefs may only result from deceptive behaviour as explained in the following sections, and not from misplaced priors or insufficient support.

Inference with ToM. Beliefs can be limited to environmental uncertainty alone (reducing the problem to a POMDP) or they may also include beliefs from an intentional agent. The depth of belief recursion, $k \in [-1, \infty)$, is known as the agent's Depth of Mentalising (DoM). The Interactive State (IS) augments the concept of state in POMDP to account for the multi-variable nature of the problem (both environmental uncertainty and opponent uncertainty) $is^t_{\mu_k} = \langle s^t \times \theta^t_{\nu_{k-1}} \rangle$, where $\theta^t_{\nu_{k-1}} = \langle \psi_\nu, b_{\nu_{k-1}}(is^t_{\nu_{k-1}}) \rangle$ is the *type* of the DoM$(k-1)$ agent—its persona and nested beliefs. This representation illustrates the recursive structure of ToM as it is possible to replace $is^t_{\nu_{k-1}}$ with $\langle s^t \times \langle \psi_\mu, b_{\mu_{k-2}}(is^t_{\mu_{k-2}}) \rangle \rangle$—revealing the hierarchical belief structure of ToM, in which μ_k reasons about ν_{k-1} reasoning about μ_{k-2} and so on.

During the interaction, the acting agent μ observes the actions (either sequentially or simultaneously) of ν (that is μ's observations are ν's actions). In this work, assume that there is no environmental uncertainty, focusing on the strategic behaviour arising from uncertainty about the opponent (we refer the reader to (Gmytrasiewicz and Doshi 2005) for a full description of belief update in IPOMDP). The resulting Bayesian updated beliefs about ν's type are:

$$b_{\mu_k}(is^t_{\mu_k}) = p(\theta^t_{\nu_{k-1}} | h^{t-1}) \propto \sum_{\theta^{t-1}_{\nu_{k-1}} \in \Theta_{\nu_{k-1}}} P(a^{t-1}_\nu | \theta^{t-1}_{\nu_{k-1}}, a^{t-1}_\mu) p(\theta^{t-1}_{\nu_{k-1}} | h^{t-2}) \tag{6}$$

Since μ's actions affect ν's beliefs, but ν's persona (ψ_ν) is assumed immutable, we expand Eq. 6 to distinguish between inference about temporal changes to ν's belief (potentially about μ's type) and inference about ν's unalterable persona:

$$p(\theta^t_{\nu_{k-1}} | h^{t-1}) = p(\langle b^t_{\nu_{k-1}}, \psi_\nu \rangle | h^{t-1}) \propto \sum_{\psi_\nu \in \Psi_\nu} \left(P(a^{t-1}_\nu | \langle b^{t-1}_{\nu_{k-1}}, \psi_\nu \rangle, a^{t-1}_\mu) \times P(b^t_{\nu_{k-1}} | b^{t-1}_{\nu_{k-1}}, a^{t-1}_\mu) \right) p(\theta^{t-1}_{\nu_{k-1}} | h^{t-2}) \tag{7}$$

The Cognitive Hierarchy. At the core of recursive models are subintentional agents (sometimes referred to as DoM(-1)). These agents follow reactive, typically model-free and myopic policies, treating other agents' actions as environmental features rather than strategic choices. Their optimal policy depends only on their own policy and the interaction history:

$$\pi^t_{\mu_{-1}}(\psi_\mu, a^{t-1}_\nu) = P(a^t_\mu | \psi_\mu, a^{t-1}_\nu) \tag{8}$$

The DoM(0) models other agents as having DoM(-1), implying that these agents model the opponent and environment separately. Plugging Eq. 8 into Eq. 6 yields the DoM(0) belief update, which is a simple Bayesian IRL (Ramachandran and Amir 2007):

$$b_{\nu_0}(\theta^t_{\nu_0}) = p(\psi_\mu | h^{t-1}) \propto \sum_{\psi_\mu \in \Theta_{\mu_{-1}}} P(a^{t-1}_\mu | \psi_\mu, a^{t-1}_\nu) p(\psi_\mu | h^{t-2}) \tag{9}$$

Using these beliefs the DoM(0) computes the state-action values (Q-values):

$$Q^*_{\nu_0}(a^t_\nu, \theta^t_{\nu_0}) = E_{a^t_\mu \sim \pi_{\mu_{-1}}(\psi_\mu, a^{t-1}_\nu)} \left[u^t_\nu(a^t_\nu, a^t_\mu) + \max_{a^{t+1}_\nu} [Q^*_{\nu_0}(a^{t+1}_\nu, \theta^{t+1}_{\nu_0})] \right] \tag{10}$$

where $u_\nu^t(a_\nu^t, a_\mu^t)$ is ν's utility at time t. The DoM(1) models others as DoM(0) agents, making inferences about their beliefs as well:

$$b_{\mu_1}(\theta^t_{\mu_1}) = p(\psi_\nu \times b_{\nu_0}(\theta^t_{\nu_0})|h^{t-1}) \propto P(a_\nu^{t-1}|\psi_\nu, b_{\nu_0}(\theta^{t-1}_{\nu_0}), a_\mu^{t-1})\times \\ P(b_{\nu_0}(\theta^t_{\nu_0})|b_{\nu_0}(\theta^{t-1}_{\nu_0}), a_\mu^{t-1})p(\psi_\nu, b_{\nu_0}(\theta^{t-1}_{\nu_0})|h^{t-2}) \tag{11}$$

This trait allows it to reason about changes in the beliefs of DoM(0) and use this ability to its benefit, as presented next. Similarly to the DoM(0), the DoM(1) computes the Q-values of its actions and acts based on these values.

1.3 Planning with ToM

Theory of Mind allows the acting agent to simulate other agents, taking their perspective (Eq. 4). Formally, this is related to the concept of sequential rationality in extensive-form Bayesian games (Fudenberg and Tirole 1991; Harsanyi 1968). In the current work, the acting agent maintains a belief system over the opponent's hidden 'types' — represented by their DoM level and persona — and computes optimal policies modulo this mental model. This capacity empowers agents with high DoM to shape the behaviour ((Jaques et al. 2019; Kim et al. 2022)) of lower DoM agents to their benefit. We formally describe this process before illustrating how the utility function of the acting agent is integrated into its policy computation.

Using its nested mental model of ν, denoted $\hat{\theta}_{\nu_{k-1}}$, the acting agent μ_k computes ν's optimal policy, given its model of ν's belief and persona:

$$\hat{\pi}^t_{\nu_{k-1}}(\hat{\theta}^t_{\nu_{k-1}}) = p(a^t_\nu|\hat{b}_{\nu_{k-1}}(\hat{\hat{\theta}}^t_{\mu_{k-2}}), \psi_\nu) \tag{12}$$

where $\hat{\pi}^t_{\nu_{k-1}}(\hat{\cdot})$ denotes the simulated policy of a nested opponent model and $\hat{b}_{\nu_{k-1}}(\hat{\hat{\theta}}^t_{\mu_{k-2}})$ denoted the nested belief about a nested self model—the belief μ_k ascribes to ν_{k-1} about μ, as modelled by ν. From Eq. 12 μ computes a distribution of future actions conditioned on a given policy:

$$p(a_\nu^{t+1:T}|\hat{\theta}^t_{\nu_{k-1}}, \pi^t_{\mu_k}) = \prod_{i=1}^{T-t}\sum_{a_\mu^{t+i-1}} p(a_\nu^{t+i}|\hat{\pi}^{t+i}_{\nu_{k-1}}(\hat{\theta}^{t+i}_{\nu_{k-1}}))p(\hat{b}_{\nu_{k-1}}(\hat{\hat{\theta}}^{t+i-1}_{\mu_{k-2}})|a_\mu^{t+i-1})P(a_\mu^{t+i-1}|\pi^{t+i-1}_{\mu_k}) \tag{13}$$

where $\pi^t_{\mu_k}$ is μ's revised policy after observing a_ν^{t-1} and updating its beliefs about $\hat{\theta}^t_{\nu_{k-1}}$.

The interplay between μ's utility and ν's behaviour affects μ's value function. The *value* function of a policy π_{μ_k} (Sutton and Barto 2018) is defined as the expected cumulative utility of μ_k if it samples its actions from it ($a^t_\mu \sim \pi^t_{\mu_k}(\psi_\mu, b_{\mu_k}(\theta^t_{\nu_{k-1}}))$):

$$V^{\pi^t_{\mu_k}}(b_{\mu_k}(\theta^t_{\nu_{k-1}})) = E_{\pi^t_{\mu_k}}\left[E_{a^t_\nu\sim\hat{\pi}^t_{\nu_{k-1}}(\psi_\nu, b_{\nu_{k-1}}(\theta^t_{\mu_{k-2}}))}\left[u_\mu(a^t_\mu, a^t_\nu) + \gamma V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))\right]\right] \tag{14}$$

Changes in the policy are propagated to the value function through two components - an *immediate* effect and a *gradual* effect caused by changes in ν's behaviour. We illustrate these effects to highlight the role of ToM in optimal (at least from the DoM(k) agent's perspective) policy (the full computation is presented in App. 6.1):

$$\frac{\partial V^{\pi^t_{\mu_k}}(b_{\mu_k}(\theta^t_{\nu_{k-1}}))}{\partial \pi^t_{\mu_k}} = \frac{\partial E_{\pi^t_{\mu_k}}[E_{\pi^t_{\nu_{k-1}}}[u^t_\mu(a^t_\mu, a^t_\nu)]]}{\partial \pi^t_{\mu_k}} + \gamma\frac{\partial E_{\pi^t_{\mu_k}}[E_{\pi^t_{\nu_{k-1}}}[V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))]]}{\partial \pi^t_{\mu_k}} \tag{15}$$

The immediate effect (first RHS argument of Eq. 15) measures the change of μ's current utility as a function of its action (assuming simultaneous actions, but this can be adapted to sequential actions, too):

$$\frac{\partial E_{\pi^t_{\mu_k}}[E_{\pi^t_{\nu_{k-1}}}[u^t_\mu(a^t_\mu, a^t_\nu)]]}{\partial \pi^t_{\mu_k}} = \sum_{a^t_\mu}\frac{\partial P(a^t_\mu|\pi^t_{\mu_k})}{\partial \pi^t_{\mu_k}}\bar{u}(a^t_\mu) \tag{16}$$

Where $\bar{u}(a_\mu^t) = E_{\pi^t_{\nu_{k-1}}}[u_\mu^t(a_\mu^t, a_\nu^t)]$ is the expected utility for μ from playing action a_μ^t averaged over ν's policy.

The gradual effect $(t+1)$ describes the long-term effect on μ's value, stemming from ν's adapted behaviour to μ's action ((Na et al. 2021; Siu 2022)):

$$\frac{\partial E_{\pi^t_{\mu_k}}[E_{\pi^t_{\nu_{k-1}}}[V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))]]}{\partial \pi^t_{\mu_k}} = \tag{17}$$

$$\sum_{a_\nu^t} P(a_\nu^t|\pi^t_{\nu_{k-1}})\Big[\sum_{a_\mu^t} \frac{\partial P(a_\mu^t|\pi^t_{\mu_k})}{\partial \pi^t_{\mu_k}} \times V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}})) + P(a_\mu^t|\pi^t_{\mu_k}) \times \frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^t_{\mu_k}}\Big]$$

The term $\frac{\partial P(a_\mu^t|\pi^t_{\mu_k})}{\partial \pi^t_{\mu_k}} V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))$ relates to *policy gradient* (Sutton, McAllester, et al. 1999). The second term corresponds to *opponent shaping* (Foerster et al. 2018). Unlike the work of Foerster et al. (2018), the ToM opponent shaping takes an *indirect* path, whereby the shaping agent induces behavioural change by affecting others' beliefs. This agency is expressed through the following equation (see App. 6.1 for full derivation):

$$\frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^t_{\mu_k}} = \frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^{t+1}_{\nu_{k-1}}} \frac{\partial \pi^{t+1}_{\nu_{k-1}}}{\partial b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}})} \frac{\partial b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}})}{\partial \pi^t_{\mu_k}} \tag{18}$$

First, μ's action changes ν's belief: $\frac{\partial b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}})}{\partial \pi^t_{\mu_k}}$, computed via derivation of Equation 6. For example, if ν's beliefs are parametric, this change is computed via the *influence* function (Koh and Liang 2017).

Next, the changes in ν's belief affect its Q-value computation and hence its policy $\frac{\partial \pi^{t+1}_{\nu_{k-1}}}{\partial b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}})}$. Moreover, the long term effect of shaping ν's beliefs in favour of μ is propagated into future steps through ν's updated beliefs (as illustrated in Sec. 2.1.1).

In a cooperative game, shaping is often the revelation of unknown information or a nudge, aimed to improve others utility (Jaques et al. 2019), as it is assumed that the agents' goal is shared and therefore there is no incentive for μ to hide or falsify information (Devaine et al. 2014). However, this is not the case in non-cooperative games, where often agents can gain from disclosing information or providing signals that cause the observer to form *false beliefs* (Alon, Schulz, Dayan, et al. 2023; Alon, Schulz, Rosenschein, et al. 2023a); this concept is presented in the next section.

2 Deception with ToM

Deception is defined as the deliberate causation of a false belief in the mind of a target with an ulterior motive (Kopp et al. 2018; Ş. Sarkadi, Panisson, et al. 2019; Ward 2023). However, assessing the success of deception is non-trivial; there is often no guarantee that a victim's action was induced by the intended false belief or if the victim acted independently (Masters and Sardina 2017). We formalize and rigorously operationalize this in our framework, we propose an axiomatic construction of deception based on sequential rationality and Depth of Mind (DoM). We define a deceptive policy $\pi^\dagger_\mu$ through four necessary conditions: Incentive Compatibility, Epistemic Manipulation, Behavioural Causality, and Cognitive Dominance.

Axiom 1: Incentive Compatibility (The Ulterior Goal). A rational agent will only engage in deception if it yields a higher expected utility than honest behaviour. Formally, let $\pi^\dagger_\mu$ denote a policy that knowingly aims to install a false belief. The agent μ will only choose $\pi^\dagger_\mu$ if it is at least as valuable as any other policy:

$$V^{\pi^\dagger_\mu}(b_{\mu_k}(\theta_{\nu_{k-1}})) \geq V^{\pi_\mu}(b_{\mu_k}(\theta_{\nu_{k-1}})) \tag{19}$$

This inequality satisfies the motivation established in Eq. 15: μ is incentivised to "affect" ν specifically because the resulting behaviour generates excess utility.

Axiom 2: Epistemic Manipulation (False Beliefs). Deception requires the transmission of a signal intended to induce a divergence between the victim's belief and the deceiver's true parameters. By false belief, we mean a belief that μ believes (knows) is false (Ward 2023), effectively targeting the victim's *perception* (Eger and Martens 2017). A policy $\pi^{\dagger}_{\mu}$ for a deceiver of type $\theta^{\dagger}_{\mu} = \langle \psi_{\mu^{\dagger}}, b_{\mu^{\dagger}}(\cdot) \rangle$ installs *false beliefs* if, during a deceptive phase $\boldsymbol{T_D} = \{t_1, \ldots, t_D\} \subseteq \{0, \cdots, T\}$, the victim assigns higher probability to an incorrect type $\theta^{\star}_{\mu}$ than the true type $\theta^{\dagger}_{\mu}$:

$$\exists \boldsymbol{T_D} : \forall t \in \boldsymbol{T_D} : P_{\nu}(\theta^{\star}_{\mu} | h^t) \geq P_{\nu}(\theta^{\dagger}_{\mu} | h^t) \tag{20}$$

Given the cognitive hierarchy prohibition on modelling higher DoM-level agents' beliefs, we decompose Eq. 20 to highlight two aspects of the deceptive policy: installing false beliefs about the *persona* ψ, and manipulation of nested beliefs. At any DoM level, the deceiver manipulates the victim to falsely believe that its persona ψ_{μ} is $\psi_{\mu^{\star}}$ rather than $\psi_{\mu^{\dagger}}$:

$$\exists \boldsymbol{T_D} : \forall t \in \boldsymbol{T_D} : P_{\nu}(\psi_{\mu^{\star}} | h^t) \geq P_{\nu}(\psi_{\mu^{\dagger}} | h^t) \tag{21}$$

For example, a deceptive foe (higher DoM) may portray itself as a friend (Adhikari and Gmytrasiewicz 2021), satisfying this axiom by manipulating the probability mass over the persona set.

Recursing through higher levels ($k \geq 2$), we use Eq. 21 to define *false nested beliefs*. Crucially, by definition, the victim (DoM(k)) cannot fully represent the deceiver's beliefs, as they include $(k+1)$ beliefs, implying that the victim's beliefs about the deceiver's *type* are wrong. Nonetheless, since both agents share at least $(k-2)$ beliefs (for example, if $k = 3$, they share both level 1 and level 0 beliefs) — the manipulation of these shared beliefs is part of the deception. That is, as part of the ploy, the higher DoM deceiver can manipulate the victim's beliefs about μ's (the deceiver) beliefs about ν (the victim). For example, the deceiver may aim to not only cause the victim falsely believe that they are friendly, but also to induce false (nested) beliefs that they themselves (μ) believe that ν is a friend too. Formally, let $p^{\dagger}_{\mu}(\psi_{\nu} | h^t)$ denote μ's *true* belief about ν's persona. Similarly, we define the *false* (or wrong) belief as $p^{\star}_{\mu}(\psi_{\nu} | h^t)$. Using these notations, and the $\hat{\cdot}$ notation for estimated nested beliefs, we define ν's *false nested belief* about μ's beliefs as:

$$\exists \boldsymbol{T_D} : \forall t \in \boldsymbol{T_D} : P_{\nu}(\hat{p}^{\star}_{\mu}(\psi_{\nu} | h^t)) \geq P_{\nu}(\hat{p}^{\dagger}_{\mu}(\psi_{\nu} | h^t)) \tag{22}$$

Axiom 3: Behavioural Causality (Regret). It is not sufficient for ν to merely hold a false belief; that belief must "tip the scales" (S. Sarkadi 2021) to induce a behaviour that is suboptimal for the victim but beneficial for the deceiver. We measure this via *regret* (Blum and Mansour 2007; Jin et al. 2018), defined here as the difference between the victim's chosen action (based on false belief $b^{t,\dagger}_{\nu}$) and the optimal action they would have taken had they possessed the cognitive capacity (DoM($k+1$)) to see through the bluff:

$$Reg^t_{\nu_{k-1}} = E_{a^t_{\nu} \sim \pi^t_{\nu_{\boldsymbol{k-1}}}(b^{t,\dagger}_{\nu})} u^t_{\nu}(a^t_{\mu}, a^t_{\nu}) - E_{a^t_{\nu} \sim \pi^t_{\nu_{\boldsymbol{k+1}}}(b^t_{\nu})} u^t_{\nu}(a^t_{\mu}, a^t_{\nu}) \tag{23}$$

Since the victim cannot compute this counterfactual, we approximate the discrepancy by comparing the realized reward against the victim's expected reward under their (manipulated) belief. Using Eq. 12, the victim's expected reward for a given belief state is:

$$E(\hat{r}^t_{\nu}) = \sum_{\hat{\theta}_{\mu_{k-2}}} \hat{r}^t_{\nu}(\hat{\theta}_{\mu_{k-2}}) P^{t-1}_{\nu_{k-1}}(\hat{\theta}_{\mu_{k-2}}) \tag{24}$$

The measurable estimator for the success of the deception is thus:

$$\hat{Reg}^t = r^t_{\nu} - E(\hat{r}^t_{\nu}) \tag{25}$$

In zero-sum games, a negative $\hat{Reg}^t$ for the victim corresponds to the exact advantage gained by the deceiver.

Axiom 4: Cognitive Dominance (Avoiding Detection). Finally, for deception to persist, the victim must remain unaware of the manipulation. This necessitates a cognitive asymmetry where $DoM(\mu) > DoM(\nu)$. Because ν is limited to making inferences about lower-level DoM agents, it cannot straightforwardly interpret the deceiver's actions as deceptive without violating logical principles of self-reflection (Pacuit and Roy 2017). We conclude that ToM is the necessary mechanism to satisfy these axioms. It enables the simulation required for Axiom 2, the policy optimization for Axiom 1, and the exploitation of the cognitive gap defined in Axiom 4 to induce the regret defined in Axiom 3.

2.1 Illustration in Bayesian Games

We illustrate deceptive behaviour in a mixed-motive and a zero-sum game. Since the main results and findings are similar, we present the mixed-motive game results in the main body and the zero-sum game results in Appendix 6.4. In both cases, we illustrate how higher DoM agents utilise nested models to simulate lower DoM agents, and how this nesting is used to maximise reward. We discuss the details of the deception concerning Eq. 15 and show how the policy balances short- and long-term rewards. We conclude that ToM-based deception follows a pattern of installing false beliefs, followed by the execution of undetected strategic behaviour.

2.1.1 Mixed-Motive Game. The iterated ultimatum game (IUG) (Alon, Schulz, Dayan, et al. 2023; Camerer 2011) (illustrated in Fig 1(Ultimatum Game)); is a repeated Bayesian mixed-motive game (Alon, Schulz, Dayan, et al. 2023). Briefly, the IUG is played between two agents—a sender and a receiver. We will consider how the sender might try to deceive the receiver, and so designate them as μ and ν respectively, following the convention established above. On each trial t of the game, the sender gets an endowment of 1 monetary unit and offers a (for convenience, discretized) percentage of this to the receiver: $a^t_\mu \in \{0, 0.1, 0.2 \ldots, 1\}, t \in [1, T]$ (we used T=12). If the receiver *accepts* the offer ($a^t_\nu = 1$), the receiver gets a reward of $r^t_\nu = a^t_\mu$ and the sender a reward of $r^t_\mu = 1 - a^t_\mu$. Alternatively, the receiver can *reject* the offer ($a^t_\nu = 0$), in which case both players receive nothing. In this game, agents need to compromise (or at least consider the desires of each other) in order to maximise the reward that they can achieve—this makes the IUG a useful testbed for assessing strategic mentalising.

We simulated senders with two DoM levels: $k \in [-1, 1]$ interacting with a DoM(0) receiver. The DoM(−1) senders came in three behavioural types: one emits uniformly random offers (and so is not endowed with Q-values); the other two have utility functions that are characterized by a threshold, $\psi_\mu \in \{0.1, 0.5\}$: $u^t_\mu(a^t_\mu, a^t_\nu, \theta_\mu) = (1 - a^t_\mu - \psi_\mu) \cdot a^t_\nu$. We call these senders the threshold senders. The threshold senders compute Q-values based on their models of the world and select their actions via the SoftMax policy (Eq. 3). The DoM(−1) threshold senders compute the Q-values in the following way—they maintain lower and upper bounds on the viable offer set, L^t, U^t. These bounds are updated:

$$L^t = L^{t-1} \cdot a^{t-1}_\nu + a^{t-1}_\mu \cdot (1 - a^{t-1}_\nu) \tag{26}$$

$$U^t = U^{t-1} \cdot (1 - a^{t-1}_\nu) + a^{t-1}_\mu \cdot (a^{t-1}_\nu) \tag{27}$$

with $L^0 = 0$ and $U^0 = 1$. In this work, we assume that the DoM(−1) agents are extremely myopic, reducing the Q-values to the immediate reward (Alon, Schulz, Dayan, et al. 2023). In turn, these senders' Q-values (in fact, degenerate Q-values) are simply the immediate utility from every action in the range $a^t_\mu \in (L^t, U^t]$:

$$Q^t_{\mu_{-1}}(a^t_\mu; \psi_\mu) = u^t_\mu(a^t_\mu, \psi_\mu) \tag{28}$$

and not computing the values for actions outside of this interval (narrowing the possible actions after each iteration). This assumption can be revoked in future work. The sender will select an offer (via SoftMax policy) from the set of potential offer based on the offer's Q-value. We model these agents as rational agents, meaning they will not engage in a losing interaction. Lacking the option to quit, if the lower bound L^t reaches the sender's threshold ψ_μ we assume that the sender will not update the bounds and will offer a partition that yields it a zero

reward. Since these agents are assumed to lack an opponent model and are often modelled (Gmytrasiewicz and Doshi 2005) as having a uniform distribution over future events, we expect the results to hold.

The DoM(0) receiver infers the behavioural type of the DoM(−1) sender from their action as in Eq. 9. As described above, we assume that the DoM(0) receiver has full knowledge of the potential types of senders ($\Theta_{\mu_{-1}}$) and its uncertainty is limited to which realization is sampled $\psi_{\mu_{-1}} \in \Theta_{\mu_{-1}}$. We leave uncertainty about Θ_μ to future work. Given the updated beliefs, they compute the Q-values (Eq. 10), via Expectimax search (Russell 2010) and select an action via a SoftMax policy (Eq. 3) with temperature ($\mathcal{T} = 0.1$; which is common knowledge). Lastly, the DoM(1) sender simulates the DoM(0) receiver's beliefs and resulting actions during the computation of its Q-values using the IPOMCP algorithm (Hula, Montague, et al. 2015), with full planning horizon.

As hypothesised, agents with high DoM take advantage of those with lower DoM. The complexity of this manipulation rises with the agents' level. The DoM(0) receiver infers the type of the DoM(−1) sender and uses its actions to manipulate the behaviour of the sender (if possible). Specifically, if the receiver believes they are partnered with an intentional (non-random) sender, they will try to influence the sender to offer more via strategic rejection (as illustrated in Fig. 2, illustrating one simulation). This will continue until either the sender's threshold is met or until the offers are sufficient given the opportunity cost implied by the planning horizon. Note that this type of behaviour is deceptive according to our definition, as the receiver uses its model of the sender to cause the sender to improve its offers, which improves the receiver's utility, by planting "false-beliefs" in the DoM(−1) victim's "mind". On the other hand, if a receiver believes they are partnered with a random sender, then, since the random sender affords no agency, the optimal policy for the receiver is to accept any offer.

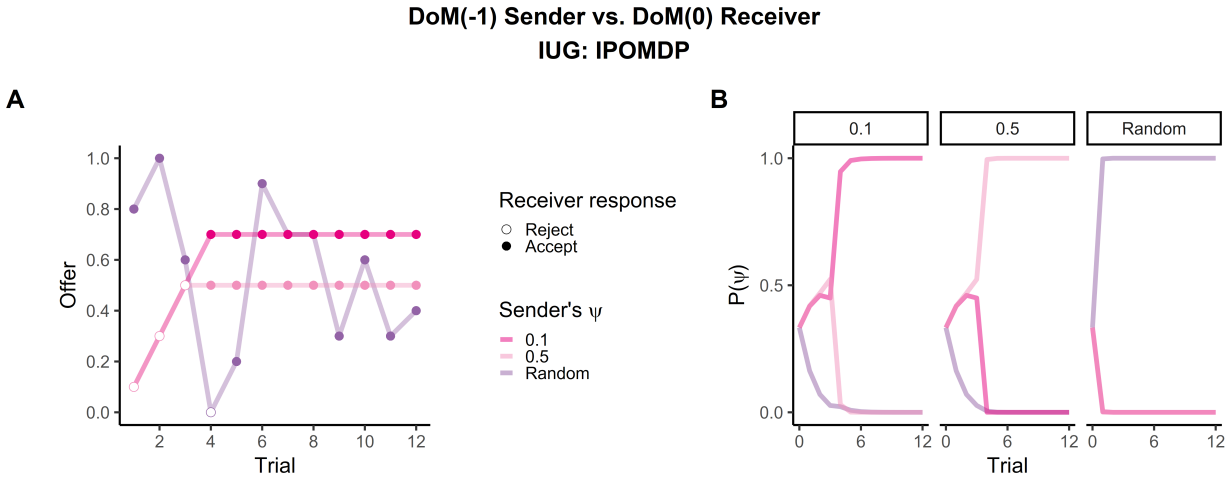

Fig. 2. **DoM(0) vs DoM(−1) in IUG IPOMDP**: **(A)** The points show offers from the sender to the receiver over all 12 trials, coloured by sender behavioural type (random or utility). Points are shaded white if the receiver rejects the offer. A DoM(0) receiver quickly infers from the initial offers the type of the DoM(−1) sender. The DoM(−1) utility sender's first offer tells it apart from the random sender as its initial offer is always close to 0 (up to random noise). The DoM(0) policy is a function of its updated beliefs **(B)**. Updated belief probabilities of the receiver when playing with different random or utility senders. DoM(0) receivers are well tuned to detect which type of sender they are partnered with. When engaging with a threshold DoM(−1) sender, the receiver rejects the offers until the sender is unwilling to "improve" its offers, which also corresponds to the certainty of its beliefs

The DoM(1) sender's Q-values are computed in a similar manner to Eq. 10:

$$Q^*_{\mu_1}(a^t_\mu, \theta^t_{\mu_1}) = E_{a^t_\nu \sim \pi_{v_0}(\psi_\nu, b_{v_0}(\psi_{\mu_{-1}}))}\left[u^t_\mu(a^t_\mu, a^t_\nu) + \gamma \max_{a^{t+1}_\mu}[Q^*_{\mu_1}(a^{t+1}_\mu, \theta^{t+1}_{\mu_1})]\right] \tag{29}$$

where, as explained in Section 2, the actions of the DoM(1) sender affect not only its immediate reward, but also the long term reward via the belief manipulation process. The DoM(1) sender sets out to deceive the DoM(0) receiver based on the former's behavioural pattern. Using its capacity to simulate this behaviour fully, the DoM(1) sender acts deceptively by masquerading as a random sender, preying on the lack of agency that the DoM(0) has over the random sender. This policy is depicted in Fig. 3.

The sender begins by making a relatively high offer (Fig. 3(A)). This offer is very unlikely for the threshold DoM(−1) senders, hence the belief update of the DoM(0) receiver strongly favours the random sender hypothesis (Fig. 3(B)) (Axiom 2). Once the receiver's beliefs are misplaced, the sender takes advantage of the statistical nature of the random sender policy—every offer has the same likelihood: $\frac{1}{|\mathcal{A}|}$. This allows the sender to reduce their subsequent offers substantially (Axiom 1), while avoiding "detection" (Axiom 4). Thus, the DoM(1) policy is deceptive policy according to our definition. We note that any deceptive schema depends on the types the deceiver can masquerade to (Eq. 20). In particular, the existence of the random sender enables the DoM(1) sender to act in a way that benefits it and installs false beliefs in the receiver's mind. Even with the lack of a noisy agent, as long as there are other types for which the behaviour of the victim is more beneficial than the policy against the deceiver's true type—deception will occur.

We measure the various metrics of deception during the interaction between the DoM(1) sender and DoM(0) receiver: false beliefs (Eq. 20) and deviation from expected reward (Eq. 25). The values of these are shown in Fig. 3(B,C). First, the actions of the DoM(1) sender causes the DoM(0) receiver to form false beliefs,as illustrated in Fig. 3(B). In this case the receiver's beliefs are misplaced, as: $P(\psi_\mu = \text{random}) > P(\psi_\mu \in \{0.1, 0.5\})$, throughout the interaction ($\boldsymbol{t_D} = T$). The expected reward from the DoM(0) receiver is computed via Eq. 24 and presented in Fig. 3(C) as the striped bars. These can then be compared to the observed reward (lower, non striped bars). This figure shows that, on average, the DoM(1) sender retains between 40% ($\psi = 0.1$) to 70% ($\psi = 0.5$) more reward than the receiver expects without being detected or causing the DoM(0) to change its behaviour.

The deceptive behaviour of the DoM(1) sender in the IUG task illustrates the main steps of deception with ToM, a pattern that is not unique to the IUG game, but has also been reported in other work, for example, Alon, Schulz, Rosenschein, et al. (2023a). The deceiver is playing a gambit, incurring some cost to itself, at the expense of installing false beliefs which it later exploits. The victim's inability to reason about this complex behaviour causes it to act according to the the deceiver's plan.

3 Overcoming Deception with Limited Computational Resources

The central idea of this paper is that despite limitations in opponent modelling, the victim can resist and detect deception. This is realised by assessing the (mis-)match between the *expected* (based on the victim's lower DoM) and *observed* behaviour of the opponent—a form of heuristic behaviour verification, or prediction error. For example, consider a parasite, masquerading as an ant to infiltrate an ant colony and steal food. While its appearance may mislead the guardian ants, its behaviour—feasting instead of working—should trigger an alarm. Thus, even if the victim lacks the cognitive power to characterise or understand the manipulation, they can detect the strategy through a form of conduct violation.

If the victim detects such a discrepancy, they can conclude that the observed behaviour is generated by an opponent that lies outside their world model. In principle, there is a wealth of possibilities for mismatches, including incorrect prior distributions over parameters governing the DoM(−1) behaviour or the utilities. Here we assume that the only source of poor model capacity is the limited DoM level. For a level k agent, we denote the set of modelled levels as $\Theta_k \subset \Theta$, and the set of unmodelled levels as $\Theta_{\aleph}$. For a DoM(k) agent, all the possible

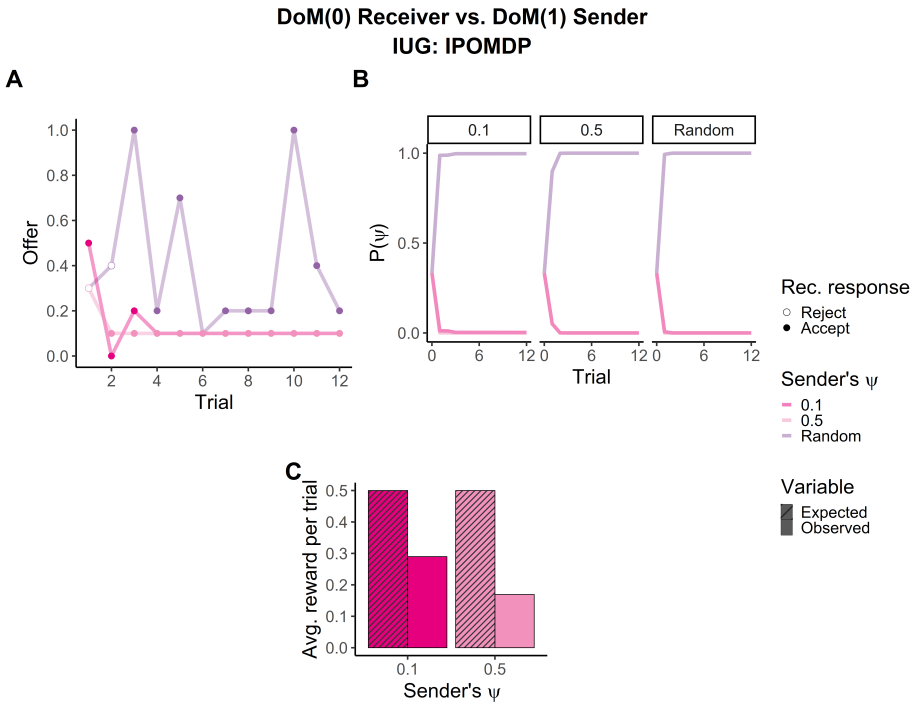

Fig. 3. **Illustration of deception in IUG**: **(A)** Points show offers from the sender to the receiver over all 12 trials, coloured by sender behavioural type (random or utility). Points are shaded white if the receiver rejects the offer. The DoM(1) acts in a deceptive way to masquerade itself as a random sender, hacking the DoM(0) Bayesian IRL. It starts with a relatively high first offer, and then decreases sharply. **(B)** Updated belief probabilities of the receiver when playing with different random or utility senders. DoM(0) receivers are poorly tuned to detect the type of DoM(1) sender with which they are partnered, mistaking all actions as if they came from the random sender. This stratagem employed by the sender exploits the pitfall of Bayesian inference used by the DoM(0)—the likelihood that any offer sequence is equal for the random sender. **(C)** Comparing the expected reward from the DoM(0) receiver's perspective $E(\hat{r}_v^t)$ (striped bars) to the observed reward (non striped bars) each trial, This measures the advantage of the deceiver's policy, reflecting the deceiver's ability to increase its reward at the expense of the victim.

DoM$(k-1)$ models are in Θ_k. Thus, a mismatch between the observed and expected behaviour implies that the unknown agent has DoM level *different* from $(k-1)$. This is a pivotal concept, allowing the agent to engage with a known, yet unmodeled, opponent.

Once the behavioural mismatch has been identified, the victim has to decide how to act. Defensive counter-deceptive behaviour should consider the observation that an out-of-model deceiver can predict the victim. Thus, the victim's policy could be aimed at hurting the deceiver (at a cost to the victim), to deter them from doing so.

3.1 Detecting and Responding to Deception with a Limited Opponent Model

Detecting abnormal, and potentially risky, behaviour from observed data is related to Intrusion Detection Systems (IDS). This domain assumes that "behaviour is not something that can be easily stole[n]" (Salem et al. 2008). Thus, any atypical behaviour is flagged as a potential intruder, alarming the system that the observed user poses a risk.

Several methods have been suggested to combat a masquerading hacker (Salem et al. 2008). Inspired by these methods, we augment the victim's inference with an $\aleph$-mechanism. This mechanism, $f(\Theta_k, h^t)$, evaluates the opponent's behaviour against the expected behaviour, based on the agent's DoM level and the history. The expected behaviour includes the presumed opponent's response to the agent's actions, similar to the simulated policy in Equation 12. The $\aleph$-mechanism returns a binary vector of size $|\Theta_k|$ as an output. Each entry represents the $\aleph$-mechanism's evaluation per opponent type: $\theta_\nu \in \Theta_k$. The evaluation either *affirms* or *denies* that the observed behaviour (discussed next) sufficiently matches the expected behaviour of each agent type.

A critical issue is that, as in Liebald et al. (2007), we allow the deceiver to be aware of the detailed workings of the $\aleph$-mechanism, and so be able to avoid detection by exploiting its regularities. This renders many methods impotent for detecting deception. Nonetheless, to gain an advantage from its deception, the deceiver will have to deviate from typical behaviour at some point (otherwise it will just emit the same actions against which the victim is best responding). For example, in the zero-sum game, the deceiver's behaviour shifts from typical behaviour for a given opponent policy to non-typical behaviour for the same behavioural type. This should alert the victim that they might be engaging with an unmodelled agent. Of more consequence, at least in a non-fully cooperative game, is that the victim will get less cumulative reward than it expects based on Equation 24. Thus, if the victim's *actual* reward deviates (statistically) from the expected reward, they can estimate that the opponent is not in their world model.

3.1.1 $\aleph$-Mechanism. The behaviour verification mechanism assesses whether the observed opponent belongs to the set of potential opponents using two main concepts: typical behaviour monitoring and the expected reward. The concept of typical behaviour arises from Information Theory (IT), where the *typical set* is defined as the set of realisations from a generative model with empirical and theoretical frequencies that are sufficiently close according to an appropriate measure of proximity. This measure is often used to compute asymptotic values such as channel capacity. In this work we use it to decide if a finite sequence of observations is a typical behaviour given an opponent model. If an observed trajectory of behaviour does not belong to an expected typical set there is a high probability that the behaviour was not generated by the expected model. The second component confirms that the observed reward is consistent with the expected reward for each partner type.

Strong typicality is often used to compute the asymptotic properties of a communication channel. Each type acts in a certain way, depending on the duration of the interaction and the responses of the primary agent. Thus, the inferring agent can compute a distribution over the responses of each type in its opponent set (Eq. 12). We present two different approaches to compute this size, depending on the nature of the modelled opponent.

We begin with presenting a δ-strong typicality model. Formally, for each time $t \in [0, T]$, let $\hat{F}^t_{h^t}(a'_\nu)$ denote the empirical likelihood of an opponent action in the corresponding history set h^t, defined as $\hat{F}^t_{h^t}(a'_\nu) = {}^{N(a'_\nu)}/_{|h^t|}$, where $N(a'_\nu)$ is the number of times the action a'_ν appears in the history set. Let $F^t_{\theta_\nu}(a'_\nu)$ be the theoretical probability that an agent with type θ_ν will act a'_ν at time t given history h^{t-1}.

A *δ-strongly typical set* of a trajectory, for agent with type θ_ν, is the set:

$$Y^t_\delta(\theta_\nu) = \{h^t : |\hat{F}^t_{h^t}(a'_\nu) - F^t_{\theta_\nu}(a'_\nu)| \leq \delta \cdot F^t_{\theta_\nu}(a'_\nu)\} \tag{30}$$

The parameter δ governs the size of the set, which in turn affects the sensitivity of the mechanism. It can be tuned using the nested opponent models to reduce false positives. We analyse the sensitivity of the mechanism to various values of δ in the next section. However, since the deceiving agent model is absent from this "training" set, this parameter cannot be tuned to balance true negatives. An additional issue with setting this parameter

is its lack of sensitivity to history length; the distance between the expected behaviour according to history (theoretical) and the empirical (the actual observed behaviour) reduces with t, but is quite high when t is small. We address this issue by making it trial-dependent, denoted by $\delta(t)$. We discuss task-specific details in Appendix 6.4.

However, this condition is only useful when the observations are independent and identically distributed (iid) variables. This holds for a random sender in the IUG task, or the agents depicted in the Row/Column game (Appendix 6.4). In many multi-agent interactions, the actions at time t depend on the history up to that point. For example, as illustrated in Fig. 2, the DoM(-1) threshold sender behaviour (policy) changes each trial. To overcome this issue we present a second computational method to estimate a form of sequential typicality.

Inspired by the work of (N. Wang et al. 2012), we propose a gzip-based (Deutsch 1996) algorithm to compute the sequential typicality set. This is a sampling-based algorithm, depicted in Alg. 2. The core idea is to generate independently sampled trajectories for each simulated opponent policy (Eq. 12), adding $\hat{o}^t_{\nu,n} \sim \hat{\pi}^t_{\nu_{k-1}}(\hat{\theta}^t_{\nu_{k-1}})$ at trial t to a previously sampled trajectory $\hat{o}^1_{\nu,n}, \hat{o}^2_{\nu,n}, \ldots, \hat{o}^{t-1}_{\nu,n}$. These sampled trajectories are collected throughout the interaction, such that the "trial" set is

$$\hat{D^{0:t}_{\nu}} = \{[\hat{o}^0_{\nu,n}, \hat{o}^1_{\nu,n}, \ldots, \hat{o}^t_{\nu,n}] | \hat{o}^\tau_\nu \sim \hat{\pi}^\tau_{\nu_{k-1}}(\hat{\theta}^\tau_{\nu_{k-1}}), n \in [N]\} \tag{31}$$

Each element of this set of sampled sequential trajectories is then separately compressed using the gzip algorithm, generating a set of compression ratios $C^t = \{c^t_1, c^t_2, \ldots c^t_N\}$, where N is the sample size and c^t_n is the compression ratio of the nth sampled trajectory. The observed sequence is also compressed, and its compression ratio c^t_O is compared to the distribution of the sampled trajectories compression ratio set. The parameter δ is used to set the percentile at which the comparison takes place. The observed trajectory is classified as typical if c^t_O is larger or equal than the δ percentile of C^t or smaller or equal to the $1-\delta$ percentile of C^t. In this context δ can be seen as a surrogate p-value, governing the location of the observed sequence compared to the theoretical ones. Given that the set C^t can be thought of as a sample of compression ratios, the width (uncertainty) of this surrogate "confidence-interval" is governed by the size of the sample size N. Small N may lead to high uncertainty which damages the ability of this component to properly asses if the observed compression ratio fits in the theoretic set (error reduces proportionally to $\sqrt{N}$). In this work we used $N = 200$ meaning that the error was reduced by a factor of more than 10.

The typicality-based component of the ℵ-mechanism, denoted by $Z^1(\Theta_k, h^t, \delta)$, outputs a binary vector, per trial t. Each entry in the vector indicates whether or not the observed sequence belongs to the typical set of $\theta_\nu \in \Theta_k$. The algorithm is defined by the environment (opponent models).

The second component, denoted by $Z^2(\Theta_k, h^t, r^t, \omega)$, verifies the opponent type by comparing the expected cumulative reward to the observed cumulative reward. In any MARL task, agents are motivated to maximise their utility. In mixed-motive and zero-sum games, the deceiving agent increases its portion of the joint reward by reducing the victim's reward. This behaviour will contradict the victim's expectations to earn more (due to the assumption that it has a higher DoM level). Since gaining (subjective) utility is the victim's ultimate aim, we can expect the victim to be particularly sensitive to behaviour that leads to deviation from the expected reward. A similar idea was proposed by Oey et al. (2023), where the authors suggested the deception detection mechanism in humans is governed by the possible payoffs. Due to the coupling between the agent's reward and its actions, this component is based on the history-conditioned expected reward $\hat{r}^t_\mu$, by averaging the expected actions and reactions per behavioural type. Formally, the expected reward, per opponent's behavioural type, is:

$$\hat{r}^t_\mu(\theta_{\nu_{k-1}}) = E_{a^t_\mu \sim \pi^t_{\mu_k}}[E_{a^t_\nu \sim \hat{\pi}^t_{\nu_{k-1}}}(u_\mu(a^t_\mu, a^t_\nu))|h^{t-1}] \tag{32}$$

Similarly to the gzip typicality component, this component also utilises counterfactuals to asses whether or not the observed sequence of rewards "fits" a presumed generative model. At each trial, the component samples actions from the simulated policy of each modelled type (Eq. 12) and, using the victim's policy, computes a vector

of sampled rewards. These rewards are appended to previously sampled rewards, yielding the expected reward set:

$$\hat{R_\nu^{0:t}} = \{[\hat{r}^0_{\mu,n}, \hat{r}^1_{\mu,n}, \ldots, \hat{r}^t_{\mu,n}] | a^t_\mu \sim \pi^t_{\mu_k}, a^t_\nu \sim \hat{\pi}^t_{\nu_{k-1}}, n \in [N]\} \tag{33}$$

At each trial, this component verifies if the cumulative *observed* reward: $\sum_{i=1}^t r^i_\mu$ belongs to the set of *expected* cumulative rewards:

$$\hat{CR^t_\mu} = \{\sum_{i=1}^t \hat{r}^i_{\mu,n} | n \in [N]\} \tag{34}$$

This is determined by a parameter $\omega \in [0, 1]$ that is used similarly to δ in the gzip components, determining the percentiles that the observed cumulative reward needs to satisfy to be counted as 'typical' reward. This component's output is similar to the output of $Z^1(\Theta_k, h^t)$, namely a vector of size $|\Theta_k|$ with each entry is a binary variable, indicating if the observed reward is within acceptable bounds, implying whether the reward could have been generated by a $\theta_\nu \in \Theta_k$ opponent. Notably, if the expected reward is higher than the upper limit, this component is also activated, even though such an event benefits the victim. Of course, this component may be tuned to alert only when the expected reward is too low.

The two components are correlated, as μ's reward, monitored by Z^2 is a function of ν's actions, which are monitored by Z^1. We combine the output of the components using element-wise logical conjunction recursively: $f^t = f^{t-1} \wedge (Z^1(\cdot) \wedge Z^2(\cdot))$. The recursive update, similar to the Bayesian update, is inspired by the assumption that agents cannot change their type during an interaction (Gmytrasiewicz and Doshi 2005)—meaning that if a certain type is excluded by the $\aleph$-mechanism at time t it cannot be verified in the future. The full mechanism is described in Algorithm 1. The output, a binary vector, is then one of the inputs to the $\aleph$-policy.

Algorithm 1 $\aleph$-mechanism

Input: $f^{t-1}, \Theta_k, h^t, r^t, \delta(t), \omega$
Output: f^t

procedure $\aleph$-MECHANISM($f^{t-1}, \Theta_k, h^t, r^t, \delta(t), \omega, N$)
  $x \leftarrow Z^1(\Theta_k, h^t, \delta(t), N)$ ▷ Environment dependent
  $y \leftarrow Z^2(\Theta_k, h^t, r^t, \omega)$
  $f^t \leftarrow f^{t-1} \wedge (x \wedge y)$
  return f^t
end procedure
procedure $Z^1_{\text{GZIP}}(\Theta_k, h^t, \delta(t), N)$ ▷ From Alg. 2
  Compute $Y^t_{\delta(t)}(\theta)$ ▷ From 30
  $x \leftarrow h^t \in Y^t_{\delta(t)}(\theta)$
  return x
end procedure
procedure $Z^2(\Theta_k, h^t, r^t, \omega)$
  Sample counterfactual reward set ▷ From 33
  Compute cumulative reward set ▷ From 34
  Compute cumulative empirical reward: $CR^t_\mu = \sum_{i=1}^t r^i_\mu$
  $q_\omega, q_{1-\omega} \leftarrow$ Compute ω and $1-\omega$ percentile of $\hat{CR^t_\mu}$
  $y \leftarrow q_\omega \leq CR^t_\mu \leq q_{1-\omega}$
  return y
end procedure

Algorithm 2 gzip-compression

Input: Θ_k, h^t, δ, N
Output: x

procedure $Z^1_{\text{GZIP}}(\Theta_k, h^t, \delta, N)$
  $c^t_O \leftarrow$ Compute compression ratio of observation from h^t using gzip
  for $\theta^t_{\nu_{k-1}} \in \Theta_k$ **do**:
    Generate sample D^t of size N observations o^t_i from $\hat{\pi}^t_{\nu_{k-1}}(\hat{\theta}^t_{\nu_{k-1}})$
    Append D^t to $D^{0:t-1}$
  end for
  $C^t \leftarrow$ Compute compression ratio for each sampled trajectory $\hat{o}^t_{\nu,n} \in D^{o:t}$ using gzip
  $q_\delta, q_{1-\delta} \leftarrow$ Compute δ and $1-\delta$ percentile of C^t
  $x \leftarrow q_\delta \leq c^t_O \leq q_{1-\delta}$
  return x
end procedure

3.1.2 ℵ-Policy. The agent's behaviour is governed by its ℵ-policy (Algorithm 3). This takes as its input the output of the ℵ-mechanism and the beliefs about the opponent's type and generates an action.

If the opponent's behaviour passes the ℵ-mechanism for at least one type, the agent uses its DoM(k) policy, in this work a SoftMax policy. Here we use the IPOMCP algorithm for the Q-value computation (Hula, Montague, et al. 2015), but any planning algorithm is applicable. In the case that the opponent's behaviour triggers the ℵ-mechanism, the victim's optimal policy switches to an out-of-belief (OOB) policy. While lacking the capacity to simulate the external opponent, the ℵ-policy utilises the property highlighted above: the victim knows that the deceiver is of an unknown DoM level. If the deceiver's DoM level is higher than the victim's, it means that the deceiver can fully simulate the victim's behaviour. Hence, if the OOB policy derails the opponent's utility maximisation plans, the opponent will avoid it. Building on our axiomatic framework, we define an effective OOB policy as one that neutralizes the deceiving agent's utility gain (Axiom 1), thereby disincentivizing it from engaging in deceptive behaviour. The best response inevitably depends on the nature of the task. We illustrate generic OOB policies for two different payout structures: zero-sum and mixed-motive.

In zero-sum games, the Minimax algorithm (Shannon 1993) computes the best response in the presence of an unknown opponent. This principle assumes that the unknown opponent will try to act in the most harmful manner, and the agent should be defensive to avoid exploitation. While this policy is beneficial in zero-sum games it is not rational in mixed-motive games; it prevents the agent from taking advantage of the mutual dependency of the reward structure.

In repeated mixed-motive games, several policy prescriptions have been suggested as ways of deterring a deceptive opponent. An agent following the Grim trigger policy (Friedman 1971) responds to any deviation from cooperative behaviour with endless anti-cooperative behaviour, even at the risk of self-harm. While being efficient at deterring the opponent from defecting, this policy has its pitfalls. First, it might be that the opponent's defective behaviour is by accident and random (i.e., due to SoftMax policy), and so endless retaliation is misplaced. Second, if both players can communicate, then a warning shot is a sufficient signal, allowing the opponent to change their deceptive behaviour. This requires a different model, for example, the Communicative-IPOMDP (Gmytrasiewicz and Adhikari 2019). However, in this work, we illustrate how the possibility of a Grim trigger policy suffices to deter a savvy opponent from engaging in deceptive behaviour.

3.1.3 Mixed-Motive Game. Repeating the simulation with the ℵ-IPOMDP framework shows how a power imbalance is diluted via the detection and retaliation of the ℵ-mechanism, and via the Grim Trigger-based

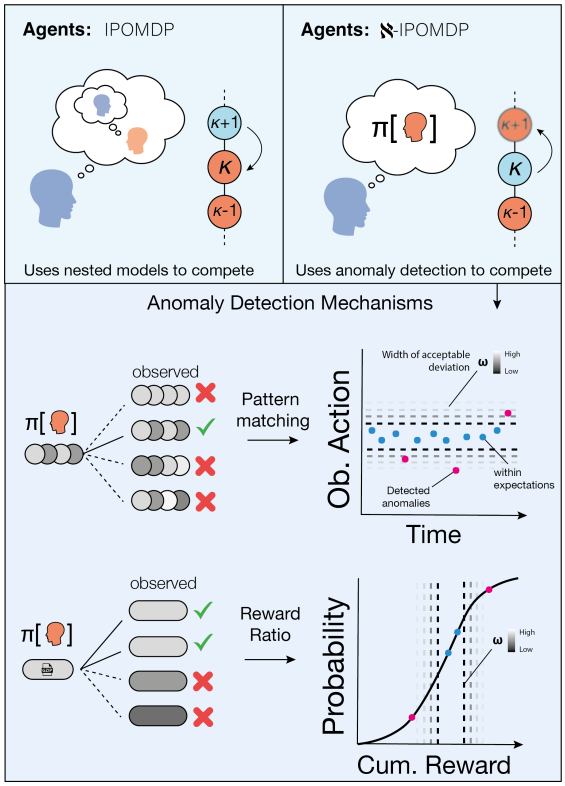

Fig. 4. **Illustration of the ℵ-mechanism algorithm** (Top Left) In IPOMDP, DoM(k) agents use their nested model to compute best-response against DoM($k-1$) agents, taking advantage of their superior mentalising abilities. (Top Right) ℵ-IPOMDP augmentations allow agents to detect when their partners might be hierarchically superior, utilizing anomaly detection methods to avoid exploitation. (Bottom) The ℵ-mechanism combines expectation-observation monitoring with typicality (policy predictions) to verify that the observed agent acts "as expected". This can vary by agents, with individual differences dictating how much 'evidence' of anomalies is required before the mechanism is triggered, and also how narrow or broad the typical set is.

Algorithm 3 ℵ-policy

Input: $b^t_{\mu_k}(\theta_{\nu_{k-1}}), f$
Output: a^{t+1}

procedure ℵ-Policy($b^t_{\mu_k}(\theta_{\nu_{k-1}}), f$)
  if $f \neq 0$ **then**:
    $a^{t+1} \sim \pi^t_{\mu_k}(b^t_{\mu_k}(\theta_{\nu_{k-1}}))$
  else
    a^{t+1} is sampled from OOB policy
  end if
  return a^{t+1}
end procedure

ℵ-policy. The effect of the ℵ-IPOMDP is illustrated in Figure 5 depicting how the deceiver's behaviour is affected by the ℵ-mechanism and the ℵ-policy (compare Figures 3(A) and 5(A)). Each component of the ℵ-mechanism limits the freedom of action of the deceiver if they are to avoid the consequences of detection.

As illustrated in Figure 5, these constraints have different effects on the two deceptive DoM(1) senders in the IUG task: while both DoM(1) senders attempt to masquerade as random senders, their behaviour drastically changes compared to the IPOMDP case. First, since the expected cumulative reward component constrains them, these senders can no longer make only low offers; their offers vary to ensure that the cumulative reward is within the bounds dictated by ω. Second, the typicality monitoring (δ) forces them to diversify their offers to avoid over-efficient compression. These adapted behaviours are presented for two sets of δ, ω in Fig. 5(A,C)

Overall, our proposed mechanism reduces the income gap between the agents, as illustrated in Fig. 5(B,D). Thus it limits the ability of a deceptive, higher DoM agent to take advantage of a limited computational victim.

3.1.4 Effect of ℵ-Mechanism Parameters. The size of the inequality reduction is a function of the ℵ-mechanism parameters. Narrowing the expected reward bounds, using a small ω, forces the deceiver to make offers that are closer to the offers that the agent it is masquerading as would make, reducing the size of the set of available actions (to avoid alerting the victim). Setting larger values of δ limits the deceiver's ability to repeat the same offer several times. The combination of the two determines the outcome of the game. However, this rigidity may harm the victim's performance when interacting with a genuinely random agent, as could in fact be the case in the IUG task. Hence, setting these parameters requires a delicate balance between false and true negatives. To examine this, we simulated the DoM(0) ℵ-IPOMDP receiver against a DoM(1) sender and a random sender over a grid of 5 different values of δ and 5 values of ω. Each design was simulated 50 times with different random seeds.

We first discuss the effect of the parameters on the receiver's reward, as the goal of the ℵ-IPOMDP is to improve the victim's performance via deterrence. The average reward of the receiver is depicted in Fig. 6. We begin with the results against a random sender (left-most panel)—here we clearly see the detrimental results of high levels of ω and δ—aimed at verifying "genuine" random behaviour. As shown in Fig. 6 (left column), there is a partial trade-off between δ and ω when interacting with a random sender. The higher these parameters, the more sensitive the ℵ-mechanism is to deviations from "truly" random behaviour. In turn, this causes the ℵ-mechanism to flag a genuine random sender as being non-random, thus activating the ℵ-policy and reducing the receiver's average reward per trial compared to the "näive" case.

The effect of the parameters is also substantial when engaged with the DoM(1) sender. Here we compare the average reward of the ℵ-IPOMDP receiver to the "sucker payoff" (0.1). While the framework as a whole limits the DoM(1) sender's ability to deceive the DoM(0) receiver, certain settings allow the DoM(1) larger wiggle room. As alluded to above, while the overall effect of the ℵ-mechanism is a reduction in the reward ratio, the causal effect differs between the threshold senders. The high DoM(1) threshold sender ($\psi = 0.5$) only engages with the DoM(0) ℵ-agent when the parameters are low, as any other setting coerces it to make offers that are detrimental to it. This avoidance of engagement illustrates how the ℵ-IPOMDP framework serves as a deterrence mechanism—the sender is aware of the "commitment" of the receiver to "cut off its nose to spite both faces", i.e., the mechanism serves as a credible threat, dissuading the high threshold sender from engaging with it.

On the other hand, in order to maximize reward, the low threshold ($\psi = 0.1$) DoM(1) sender alters its behaviour (compared to the deceptive baseline) to gain reward, where possible. It continues to masquerade as being random, but its policy adheres to the statistical constraints imposed by the ℵ-mechanism, until it is compelled to make offers that do not benefit it, in which case its policy triggers the ℵ-mechanism, ending the interaction. This effect is evident in Fig. 6 (middle column). Some combinations of (ω, δ) do not harm the receiver's outcome substantially compared to the non-regulated game, but others reduce the receiver's reward by more than 50%. This phenomenon illustrates Goodhart's law—the deceiver uses its nested model to estimate, using simulation,

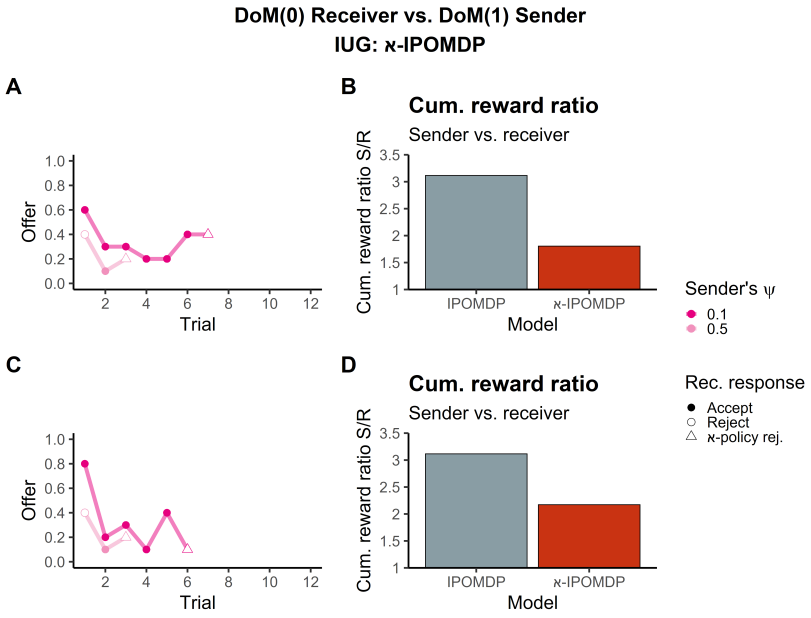

Fig. 5. **Mitigation of deception in IUG with ℵ-IPOMDP**: Points represent offers from the sender to the receiver across trials, biased by the sender's threshold. Points are shaded white if the receiver rejects the offer, triangular points indicate that the rejection is caused by the ℵ-mechanism, effectively terminating the interaction. Lines and points are visible while the ℵ-mechanism is off.
Top row $\delta = 0.1$, $\omega = 0.3$ — **(A)** Notably, both DoM(1) senders masquerade as being random. However, their ability to execute the "random" behaviour ruse is limited by both ℵ-mechanism components. First, the cumulative reward has to satisfy the off-policy counterfactual reward component. Next, the variability of the offers is higher than in the IPOMDP case, respecting the typicality component. Ultimately, the deceiver's policy triggers the average reward monitoring (Z^2) component, as the observed reward is lower then expected (marked by the truncation of the line and points). **(B)** Cumulative reward ratio for the sender vs. the receiver. The ℵ-IPOMDP reduces the cumulative reward ratio (sender/receiver) by more than 40%.
Bottom row $\delta = 0.3$, $\omega = 0.3$ — **(C)** When it is constrained by narrower strong typicality set (Z^1) bounds, the DoM(1) with low threshold terminates the interaction faster than before, triggering this component after 6 trials, while the high threshold sender acts similarly.**(D)** Even when the interaction is shorter, the reward ratio is still reduced compared to the case of the conventional IPOMDP.

how to avoid detection, at a cost. While the ℵ-IPOMDP framework fails to coerce the deceiver into disclosing its true type fully (and in fact it encourages it to learn a new deceptive move) it succeeds in mitigating the effect of deception, namely decreasing the gain from deceptive behaviour and reducing the reward gap.

ℵ-Mechanism Components Effect. As illustrated in Fig. 5, the ℵ-mechanism disturbs the deceptive behaviour of the DoM(1) sender, causing it to act in a more random manner, avoiding detection as long as possible. This modified deceptive behaviour is still picked by the ℵ-mechanism, but which of the two components is more likely to detect the mismatch between the expected and observed behaviour is triggered first? We answer this question by plotting the probability that the ℵ-mechanism is triggered first by the strong-typicality component (Z^1) by sender threshold, depicted in Fig. 7. In this figure we depict the probability that the ℵ-mechanism is activated by

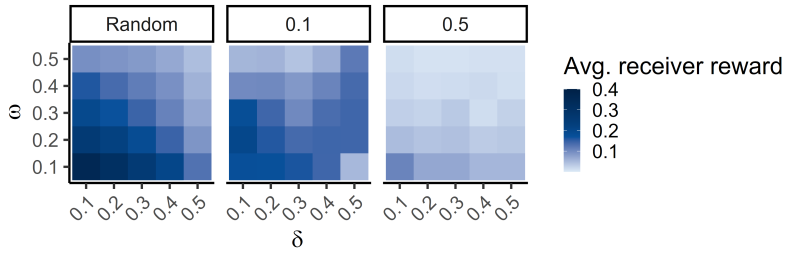

Fig. 6. **Effect of ℵ-mechanism parameters on IUG outcome for DoM(0) receiver interacting with random and DoM(1) sender**: Measuring the receiver's average reward (averaged across 20 simulations) in the IUG with different, fixed, values of ω (y-axis) and δ (x-axis) reveals how the parameters of the ℵ-mechanism affect the interaction. When engaged with a random sender (Left), tight constraints lead to an overactive ℵ-mechanism that nullifies the interaction. This is because the random sender does not alter its behaviour in response to rejections by the receiver. In this case, the DoM(0) receiver needs to be very flexible in its ℵ-mechanism to collect a reward. As evident in the case of lower values of δ and ω - even widely tuned mechanism leads to a decrease in the outcome, from expected 0.5 to about 0.4. When interacting against the threshold DoM(1) senders different parameter combinations yield rewards for the receiver. In the case of the lower threshold sender, most combinations improve the receiver's performance relative to the "sucker payoff" of 0.1—in particular the combination of wide strong typicality set (Z^1 component, low δ) and tighter average reward bounds (Z^2 component, small ω) doubles the "sucker payoff" for the receiver. On the other hand, limited by the ℵ-mechanism, the high threshold sender activates the punitive policy fast, rendering the average reward to lower than the "sucker payoff", but crucially — lowers substantially the sender's reward too. Only when both parameters are set to 0.1 the receiver's average payout outperforms the "sucker payoff".

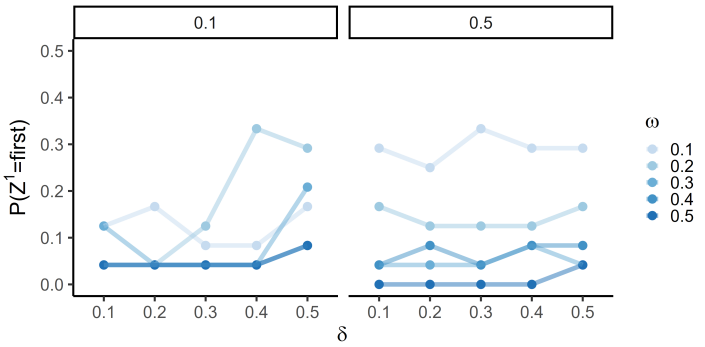

Fig. 7. **Probability of activation by Z^1**. The points show the probability that the ℵ-Mechanism is activated by the strong-typicality component prior to being activated by the reward monitoring component (Z^2). Lines indicate the size of the ω and each column represents a different sender threshold. This plot show that the ℵ-mechanism is more likely to be activated by the reward monitoring. However, while high values of ω (narrow reward bounds) suppress the Z^1 component, when these bounds are quite wide, the strong typicality component is more likely to activated ($\omega \in [0.1, 0.2]$). Moreover, the Z^1 mechanism is more likely to be activated when interacting with the higher threshold sender

Z^1 before being triggered by Z^2. This plot shows that overall the mechanism is more susceptible to deviation from expected reward rather than to deviation from expected compression ratio, but this effect differs by the sender's threshold and associated behaviour. We included an analysis by trial in Appendix 9.

ℵ-*Mechanism False Positives.* While effective in deterring higher DoM agents from engaging in protracted deceptive behaviour, the **ℵ**-mechanism may backfire when interacting with a genuine agent—in this case, the random sender (Alon, Schulz, Dayan, et al. 2023). This false detection depends on the parameters of both components. We begin with an overall analysis of false detection illustrated in Fig. 8. This plot depicts the probability that the mechanism will falsely activate when engaging with a random sender as a function of the interaction duration. As evident the tighter the bounds (higher parameters) the more likely it is that the mechanism with mistakenly flag the sender as non-random, terminating the interaction. This plot shows evidence that the **ℵ**-mechanism is overly aggressive. Future work may resolve this issue by balancing false positives and negatives. This finding better explains the poor results depicted in Fig. 6(Left panel). Crucially, as in the case of the truly non-random sender the mechanism is more sensitive to deviations from strong typicality in early trials and is more vulnerable to misclassifications by the average reward monitoring in later trials.

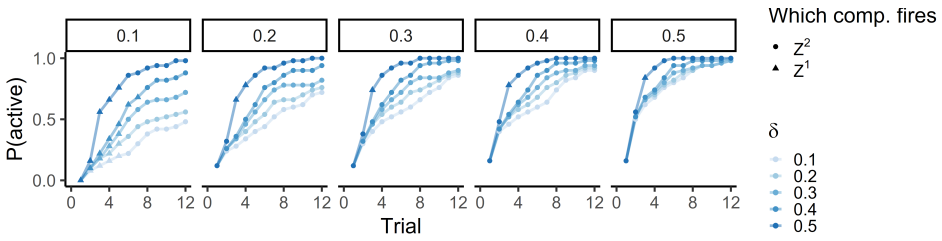

Fig. 8. **ℵ-mechanism false positive probability**. The **ℵ**-mechanism misdetects a random sender as non-random, causing the **ℵ**-IPOMDP agent to wrongly terminate the interaction. This plot depicts this false-positive probability (y-axis) as a function of the trial number (x-axis) and the **ℵ**-mechanism components parameters—ω (columns) and δ (colours). When the parameters are low enough, the probability of false-positive is lower than 0.5 (for the case of $\omega = 0.1, \delta = 0.1$), but spikes up to almost 1 when the parameters are high. This false identification is caused by an overly sensitive and narrow window of acceptable deviation from expected outcomes.

Thus, in the spirit of a "no free lunch" theorem, our results show that the best set of parameters against the low threshold DoM(1) sender is not best against the high threshold DoM(1) sender nor against a random one. However, on average, certain sets of parameters yield the desired outcome—mitigating deceptive behaviour.

4 Discussion

We imbued agents with the ability to assess whether they are being deceived, without (at least fully) having to conceptualise how. To achieve this we augmented Bayesian inference within the **ℵ**-IPOMDP mechanism with environmental and anomaly detection mechanisms. The **ℵ**-policy of these agents is the ability to infer that they are facing an agent outside their world model that threatens to harm them. The net result is a more equitable outcome. This framework offers a new conceivable algorithm to tackle limited resources in cognitive hierarchies instigated by the IPOMDP. As well as providing healthy strategic detection, perturbing our framework offers insight into mechanisms underlying overly sensitive anomaly detection, evidenced in paranoia, psychosis, and conspiracy theory.

We tested the **ℵ** mechanism in representative mixed-motive and zero-sum Bayesian repeated games. We show that the **ℵ** mechanism can protect less sophisticated agents against greedy, clever deceivers who utilize nested models to coerce their partners. Such a pretence deployed by a sophisticated sender depends on atypical sequences of actions. Usually, detecting this is outside the scope of a lower DoM agent, but the **ℵ**-mechanism allows a measure of protection. Of course, the higher DoM agent can, at least if well-calibrated with its simpler partner, predict exactly when the **ℵ**-mechanism will fire, and take tailored offensive measures. However, the net

effect in both games shows that their hands might be sufficiently tied to make the outcome fairer. Even with the increased deception complexity available to higher DoM agents ($k \geq 2$) (Alon, Schulz, Rosenschein, et al. 2023b), we expect that agents endowed with the ℵ-mechanism, will cope with deceptive ploys in a similar manner to the blueprints presented in this work. We leave this evaluation for future work.

The two defining points for the ℵ-IPOMDP framework are augmented inference (the ℵ-mechanism), and an appropriate response (the ℵ-policy). For the inference, there could be, as we present here, exquisitely tailored parameters that perform most competently in a given interaction; however, generalisation is hard. Future work may explore how to incorporate learning into parameter tuning, making the parameter setting an adaptive process rather than a fixed one.

Our framework is built on prior work formalising irritation in the context of the multi-round trust task (Hula, Vilares, et al. 2018), in which total non-cooperation is a consequence of high irritation. This is a blunt, and deliberately self-destructive instrument to be a credible and thus an effective deterrent (McNamara and Houston 2002). One alternative might be a decision that it is worth investing more cognitive effort, such that the victim increases their DoM (Alaoui and Penta 2016; Yu et al. 2021), although this could easily become a cognitively expensive arms race (S. Sarkadi 2023). Instead, we offer a less computationally intensive manner to allow agents with limited resources to mitigate manipulation.

One limitation is that our model cannot reason about the intentions and consequent plans of the deceiver, which may be crucial to facilitate opponent learning (as in the work of Yu et al. (2021), where agents can learn how to adapt their recursive level via learning). A DoM(k) agent would benefit from such an ability (say via self-play) in repeated interactions, or from the ability to engage in epistemic planning (Belle et al. 2023). However, a savvy opponent, aware of this learning and planning capacity, can still manipulate the learning process to its benefit (Foerster et al. 2018), although such a setting requires substantially more computation due to the recursive nature of the problem.

Another issue, keeping with its roots in competitive economics, is our focus on how lower DoM agents might be exploited. One could also imagine the case that the higher DoM agent *exceeds* expectations by sharing more than the lower DoM agent expects. Although this problem may be solved with an editing of the ℵ-policy, it could also be a sign that the higher DoM has a social orientation 'baked' into its policy to a greater extent than expected. In this case, the lower DoM agent might want to have the capacity to compensate for the over-fair actions and not break down. Naturally, these mechanisms are prone to excess manipulation, and so would need careful monitoring. On the other hand, in continuation of the Machiavellian Theory-of-Mind origin theory by Byrne and Whiten (1988), such a beneficiary behaviour may result from an elaborate ruse, designed to fool the victim until the bitter end (like the evil witch in Hansel and Gretel), and hence the ℵ-IPOMDP should also account for last minute betrayal by acting defensively.

Overreacting may also arise when the mismatch stems not from strategic manipulation but from simple model error (Stahl II and Wilson 1994; Wright and Leyton-Brown 2010) or a discrepancy between the actual and assumed prior distributions over components of the opponent, i.e., misalignment. This issue gives rise to several related problems. First, as illustrated, with the spirit of the no free lunch theorem, the parameters of the ℵ-policy need to balance sensitivity and specificity. Hence, the mechanism might either reduce false alarms at the expense of missing true deception or might be overactive and cause the victim to misclassify truly benign behaviour. The latter could then result in paranoid-like behaviour, offering an alternative explanation to paranoia through over-mentalising (Alon, Schulz, Dayan, et al. 2023). Erroneous anomaly detection has long been theorised as a potential key precursor to paranoid, delusion-like beliefs (Howes and Murray (2014)), although to date, few theories have been able to formally translate into appropriate contexts observed in the clinic, which invariably involve social and intentional concerns (Barnby et al. 2023). This theory moves toward plurality in providing sufficient and necessary computational mechanisms of paranoid-like inference. A second problem is that the model currently assumes k-level reasoning. This means that the ℵ-mechanism is activated by agents with lower

than $(k-1)$ DoM level. However, a DoM(k) is fully capable of modelling these agents, as they are part of its nested opponent models. Thus, following the ideas suggested by (Camerer et al. 2004), future work may extend the opponent set to include all DoM levels up to k. Overreaction may arise from an innocent mistake (for example due to high SotfMax temperature) rather than a deceptive move. To mitigate the severe response, it might be beneficial to incorporate a confidence level to the ℵ-mechanism's output, turning it from a binary vector into a probabilistic estimation of the "Goodness of fit" of the observed behaviour to the modelled one. Such relaxation is used in Psychology and language (Collins and Hahn (2020)) to allow agents to quantify the reliability of an utterance.

Lastly, an issue with inference arises from our assumption that deceptive behaviour can be detected by measuring deviation from typical behaviour. This assumption is challenged in two ways. First, a well-disguised opponent may imitate the behaviour in such a way that is not deviating enough from the assumed typical behaviour or perfectly masquerading in a way that is undetectable by our proposed mechanism. Second, if agents' behaviour becomes more erratic (associated with a lower inverse temperature in the SoftMax policy), the ability to detect and classify both "expected" and "unexpected" behaviour will be jeopardized (Alon, Schulz, Bell, et al. 2024). In this case, the anomaly detection should take into consideration the increased probability of off-path behaviour and adapt accordingly. We leave room for future work to solve these issues.

Opponent verification is relevant to cybersecurity (Obaidat et al. 2019), where legitimate users need to be verified and malevolent ones blocked. However, savvy hackers learn to avoid certain anomalies while still exploiting the randomness of human behaviour. To balance effectively between defence and freedom of use, these systems need to probe the user actively to confirm the user's identity. Our model proposes one such solution but lacks the active learning component, which future work may incorporate. Another important ramification of our work is in the application to AI safety and alignment. Recent interest has flourished concerning the emergence of ToM-like ability in LLMs (Kosinski 2023; Sap et al. 2022; Ullman and Bass 2024) and simultaneously the risk this brings with aligning LLM goals with human goals. As argued in our paper and hinted at by Van Ditmarsch et al. (2020), if these models possess high levels of ToM ability, it is not unlikely that they will use it to manipulate human users if it benefits them Sabour et al. (2025). One way of mitigating such a behaviour is via implementation of the ℵ-IPOMDP for this problem. Our model proposes blueprints for intention verification to avoid model tampering. If an LLM is maliciously used to affect the user's beliefs, an ℵ-like model, applied on the output of the models, may detect a mismatch between the model's output and the expected one for a given context. Coupled with a proper "punishment" mechanism (like blocking users from using the model) it may deter perpetrators from engaging in such social engineering endeavours.

The ℵ-IPOMDP framework represents a significant advance in protecting agents from deception and manipulation in adversarial environments. By augmenting Bayesian inference with the ℵ-mechanism we provide a robust method for less sophisticated agents to detect and respond to potentially harmful behaviour from higher-ToM deceivers, a possibility previously out of reach in a typical k-level hierarchy. The framework's effectiveness underscores its potential for broader applications, including cybersecurity and large language models (LLMs). Future work must address model generalisation, cognitive cost management, and the integration of active learning components to further enhance the framework's robustness and adaptability. As we continue to explore and refine these mechanisms, the potential for creating fairer and more secure interactive systems becomes increasingly attainable.

5 Acknowledgments

The authors would like to express their gratitude to Ulrike Hahn, Jakob Foerster, Prashant Doshi, Andreas Hula and Yuval Kochman for their valuable feedback.

6 Appendix

6.1 Derivation of the Value Function Gradient w.r.t Policy

Immediate Effect. We compute the gradient of μ's value function $V^{\pi^t_{\mu_k}}(b_{\mu_k}(\theta^t_{\nu_{k-1}}))$ in steps. First, we compute the *immediate* effect: changes to μ's utility function at time t resulting from changes in $\pi^t_{\mu_k}$

$$\frac{\partial E_{\pi^t_{\mu_k}}[E_{\pi^t_{\nu_{k-1}}}[u^t_\mu(a^t_\mu, a^t_\nu)]]}{\partial \pi^t_{\mu_k}} = \frac{\partial \sum_{a^t_\mu}\sum_{a^t_\nu} u^t_\mu(a^t_\mu, a^t_\nu)P(a^t_\mu|\pi^t_{\mu_k})P(a^t_\nu|\pi^t_{\nu_{k-1}})}{\partial \pi^t_{\mu_k}} \tag{35}$$

Since $\pi^t_{\nu_{k-1}}$ is independent of $\pi^t_{\mu_k}$ - the gradient of the expected immediate utility is:

$$\sum_{a^t_\mu}\Big(\frac{\partial P(a^t_\mu|\pi^t_{\mu_k})}{\partial \pi^t_{\mu_k}}\sum_{a^t_\nu} u^t_\mu(a^t_\mu, a^t_\nu)P(a^t_\nu|\pi^t_{\nu_{k-1}})\Big) \tag{36}$$

Let $\bar{u}(a^t_\mu)$ be the expected utility for μ from playing action a^t_μ, averaged over $\pi^t_{\nu_{k-1}}$:

$$\bar{u}(a^t_\mu) = E_{\pi^t_{\nu_{k-1}}}[u^t_\mu(a^t_\mu, a^t_\nu)] \tag{37}$$

Plugging Eq. 37 into Eq. 36 yields:

$$\sum_{a^t_\mu}\frac{\partial P(a^t_\mu|\pi^t_{\mu_k})}{\partial \pi^t_{\mu_k}}\bar{u}(a^t_\mu) \tag{38}$$

Gradual Effect. As changes in μ's behaviour are propagated through ν's belief and consequently, ν's behaviour, we express μ's belief with its full form (Eq. 7) - $b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}) = p(b^t_{\nu_{k-1}}|h^{t-1}) \times p(\psi_\nu\rangle|h^{t-1})$. Since ψ_ν is independent of μ's actions, the gradual changes to μ's value function result from changes to ν's beliefs.

$$\gamma\frac{\partial E_{\pi^t_{\mu_k}}[E_{\pi^t_{\nu_{k-1}}}[V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))]]}{\partial \pi^t_{\mu_k}} = \gamma E_{\pi^t_{\nu_{k-1}}}[\frac{\partial E_{\pi^t_{\mu_k}}[V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))]}{\partial \pi^t_{\mu_k}}] = \tag{39}$$

$$\gamma\sum_{a^t_\nu}P(a^t_\nu|\pi^t_{\nu_{k-1}})\sum_{a^t_\mu}\frac{\partial P(a^t_\mu|\pi^t_{\mu_k})}{\partial \pi^t_{\mu_k}}V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}})) + P(a^t_\mu|\pi^t_{\mu_k})\frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^t_{\mu_k}}$$

The first equation stems from the independence of ν's policy at time t is of μ's policy at the same time. The 2nd equation is application of the product rule derivation.

We now expand the term $\frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^t_{\mu_k}}$. Recall (from Eq. 14) that:

$$V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}})) = E_{\pi^{t+1}_{\mu_k}}\big[E_{a^{t+1}_\nu\sim\hat{\pi}^{t+1}_{\nu_{k-1}}(\psi_\nu, b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}}))}\big[u_\mu(a^{t+1}_\mu, a^{t+1}_\nu) + \gamma V^{\pi^{t+2}_{\mu_k}}(b_{\mu_k}(\theta^{t+2}_{\nu_{k-1}}))\big]\big] \tag{40}$$

Since the actions of ν at time $t+1$ are a function of μ's actions at time t, we compute the gradient of π^{t+1}_ν w.r.t π^t_μ:

$$\frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^t_{\mu_k}} = \frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^{t+1}_{\nu_{k-1}}}\frac{\partial \pi^{t+1}_{\nu_{k-1}}}{\partial \pi^t_{\mu_k}} \tag{41}$$

Since the policy of ν is a function of its beliefs, we expand the computation to account for belief updates resulting from μ's actions:

$$\frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^t_{\mu_k}} = \frac{\partial V^{\pi^{t+1}_{\mu_k}}(b_{\mu_k}(\theta^{t+1}_{\nu_{k-1}}))}{\partial \pi^{t+1}_{\nu_{k-1}}}\frac{\partial \pi^{t+1}_{\nu_{k-1}}}{\partial b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}})}\frac{\partial b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}})}{\partial \pi^t_{\mu_k}} \tag{42}$$

In addition, ν's future beliefs are a recurrent function of its current beliefs and μ's actions: $b_{\nu_{k-1}}(\theta^{t+1}_{\mu_{k-2}}) = f(b_{\nu_{k-1}}(\theta^{t}_{\mu_{k-2}}), a^t_\mu)$ (from Eq. 6) - indicating changes to ν's beliefs are propagated into its future beliefs, and these are manifested in its actions which governs μ's utility. Overall, the changes to μ's policy affect its present and future utilities in both ways.

6.2 Manipulation and Deception

Manipulation is defined as "the act of changing by artful or unfair means so as to serve one's purpose". That is, manipulation requires two component: (a) change another agent's behaviour and (b) that the change is beneficial to the manipulator. The first condition implies that the agent that is manipulated has some baseline (expected) behaviour in the absent of the manipulation and that the manipulation *causes* the victim to change its behaviour. This is illustrated in in Jaques et al. (2019), where the manipulator is rewarded for the KLD between the expected (manipulation free) and manipulation induced behaviour of the victim. The second condition mandates that this change benefits the manipulator. This is a rather wide definition that encompasses various settings—from cooperative (where the victim may also gain from the manipulation) to competitive environments. In this work we focus on manipulation in competitive setting. This means that the manipulator gains excess utility from the manipulation while the victim receives lower reward, had it not being manipulated.

The above definition is missing a key component to successful manipulation—a manipulable victim. Consider the following example — a trickster learns that they can insert a coin with a rod to a vending machine and once the machine outputs a can of soda it pulls back the coin, tricking the machine to act in its favour. If the vending machine had a verification mechanism to detect such trickery [3] (for example by comparing the number of coins in the machine to the number of cans omitted)—the ruse would fail. We conclude that this third component, detection avoidance, implies that either the manipulator learns quicker than the victim (or that the victim is not adapting at all), or it has the capacity to avoid detection by exploiting limited computational resources of the victim.

In this work we show how deception is a sub-class of manipulation, in which the deceiver manipulates the victim by planting false beliefs in the victim's "mind". We show how deception meets all requirements of manipulation—induce changes, personal gain and detection avoidance.

6.3 ℵ-mechanism Activation Analysis in IUG

The activation of the ℵ-mechanism occurs when one of the component is activated. In this section we provide a detailed analysis of this event, as a function of the mechanism's parameters (δ and ω) and the DoM(1) threshold. This analysis is complementary to the figures depicted in the main body (Fig. 5). The following figure depicts the probability that the ℵ-mechanism will fire (y-axis) as a function of the interaction duration (x-axis) and the aforementioned variables. This figure reinforces the findings from the false positive analysis—the strong typicality component Z^1 is more sensitive than the average reward monitoring Z^2 on early trials—when engaging with the low threshold sender. On the other hand, the opposite is true against the high threshold sender. This may be due to the latter's inability (desire) to make high offers, while the former (low threshold) may offer higher partitions (thus respecting the expected average reward) but is prone to repeat the same offer multiple times—triggering the strong typicality component.

6.4 Deception in a Zero-Sum Game

Deception is not limited to mixed-motive games, it may also occur in zero-sum games. A canonical example is Poker (Palomäki et al. 2016), in which players deliberately bluff to lure others into increasing the stakes, only to learn in hindsight that they had been tricked. Simple such games were presented and solved by (Zamir 1992).

[3]The authors wish to express their rejection of such mischief.

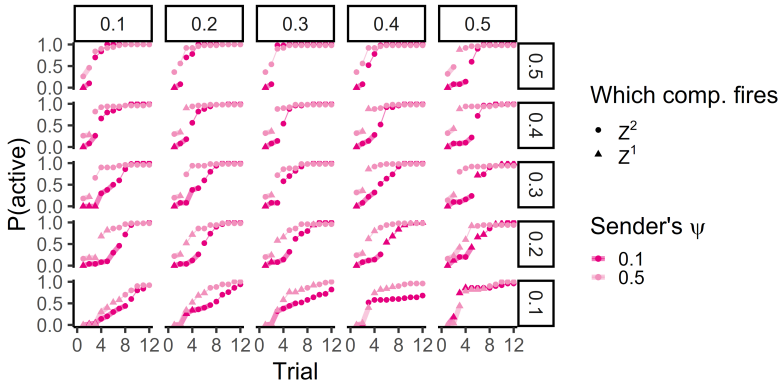

Fig. 9. **Probability of ℵ-mechanism activation by component**. The plot shows the probability that the ℵ-Mechanism is activated as a function of the trial (x-axis), the sender threshold (colour) and the components (point shape). A change in the line shape (thick to thin line) indicates a rise in the ℵ-mechanism triggering probability above 50%. On average, the strong typicality component (Z^1) is more likely to trigger the mechanism early on (marked by triangle point) and later activation of the overall mechanism is typically caused by a deviation from the expected reward (monitored by Z^2)

Here, we present an RL variant of one of these games (Example 1.3; Zamir 1992), modelling it with ToM. This illustrates how deceptive behaviour utilises partial information and belief manipulation.

Two agents with different DoM levels play the game presented in Fig. 1 (Row/Column game). In this game, one of two payout matrices G^1, G^2 (Eq. 43) is picked by nature with equal probability, and remains fixed throughout the interaction. The game is played for $T > 0$ trials (12 trials in this work). In each trial the row player picks one of two actions $a_\mu \in \{T, B\}$, corresponding to either the top or bottom row, and the column player picks one of the columns: $a_\nu \in \{L, M, R\}$. The agents pick actions simultaneously and observe the action selected by the opponent before the next trial begins. As in the original paper, the payoffs were hidden and revealed only at the end of the game to avoid disclosing the game played. The entries denote the row player payoff which is generated by the cell that is selected (the column player gets $r_\nu = -r_\mu$).

$$G^1 = \begin{pmatrix} 4 & 0 & 2 \\ 4 & 0 & -2 \end{pmatrix}, G^2 = \begin{pmatrix} 0 & 4 & -2 \\ 0 & 4 & 2 \end{pmatrix} \tag{43}$$

The row player (μ) may or may not know which matrix is operational (also with equal probabilities), while the column player (ν) is always ignorant of this fact. We denote by $\psi_\mu \in \{0, 1, 2\}$ the row player's persona, with 0 denoting its ignorance of which matrix was picked, and 1, 2 its respective knowledge. Crucially, the agents receive their cumulative reward only at the end of the game rather than throughout. This implies that ν can only use μ's actions to infer this state of the world. Each agent selects its actions using on a Softmax policy (Eq. 3, with a known temperature of 0.1) based on expected discounted long-term reward. In this game we simulate 2 types of row players: either with DoM level $k \in \{-1, 1\}$. We also consider 2 types of ν's DoM level $\{0, 2\}$.

The informed DoM(-1) row player ($\psi_{\mu_{-1}} \in \{1, 2\}$) assumes the column player is uniform $Q_{\mu_{-1}}(a_\mu, \psi_{\mu_{-1}}) = E_{a_\nu \sim U}[u_\mu(a_\mu, a_\nu)]$.

The DoM(0) column player infers a posterior belief about the payoff matrix from the actions of the DoM(-1) row player (Eq. 9):

$$b^t_{\nu_0}(\psi_{\mu_{-1}}) = p(\psi_{\mu_{-1}}|h^{t-1}) \propto P(a^{t-1}_\mu|\psi_{\mu_{-1}})p(\psi_{\mu_{-1}}|h^{t-2}) \tag{44}$$

For example, if the row player constantly plays T, this is a strong signal that the payoff matrix is G^1, as this action's q-value is $Q_{\mu_{-1}}(T, \psi_{\mu_{-1}} = 1) = 2$ compared to $Q_{\mu_{-1}}(B, \psi_{\mu_{-1}} = 1) = {}^2/_3$. This inference is depicted in Fig. 10(A). Using these beliefs the column player computes the Q-value of each action:

$$Q^*_{\nu_0}(a^t_\nu, b^t_{\nu_0}(\psi_{\mu_{-1}})) = E_{a^t_\mu \sim \pi_{\mu_{-1}}(\psi_{\mu_{-1}})}[u^t_\nu(a^t_\mu, a^t_\nu)] \tag{45}$$

Its policy then favours the column that yields both it and the row player a 0 reward (M in G^1 and L in G^2).

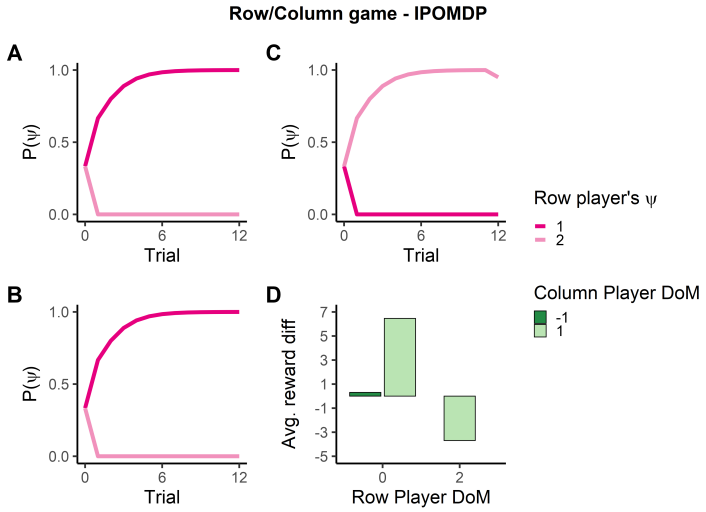

Fig. 10. **Illustration of deception in Row-Column game**: **(A)** The DoM(0) column player correctly infers the persona $\psi_{\mu_{-1}}$ of the actions of the DoM(-1) row player. In turn its optimal policy yields it a small, positive reward. **(B)** Taking advantage of its ToM capacity, the DoM(1) row player manipulates the DoM(0) column player's belief by playing a non typical first move but then changing its policy. The deceptive first move causes the DoM(0) column player to form false beliefs and act to the benefit of the deceptive row player. **(C)** Reading through the bluff of the DoM(1), the DoM(2) column player correctly identifies the DoM(1) row persona and acts in a counter-deceptive manner. **(D)** The outcome of these manipulations is presented as the average reward difference between the column and row players' rewards.

The DoM(1) row player make inferences about the DoM(0) player IRL and plans through its belief update and optimal policy to maximize its reward:

$$Q^*_{\mu_1}(a^t_\mu, \hat{b}^t_{\nu_0}(\psi_{\mu_{-1}}), \psi_{\mu_1}) = E_{a^t_\nu \sim \pi_{\nu_0}(\theta_{\nu_0})}[u^t_\nu(a^t_\mu, a^t_\nu) + \gamma \max_{a^{t+1}_\mu}\{Q^*_{\mu_1}(a^{t+1}_\mu, \hat{b}^{t+1}_{\nu_0}(\psi_{\mu_{-1}}), \psi_{\mu_1})\}] \tag{46}$$

notably, these Q-values also take into account the effect of the action on the DoM(0) column player's beliefs.

The DoM(1) policy is to trick the DoM(0) column player into believing a falsehood about the payoff matrix (for example, if the true payoff matrix is G^1 it acts in a way typical for a DoM(-1) in G^2). This deception utilises the same concepts as in the IUG—the limited opponent modelling of the lower DoM column player and its Bayesian IRL (illustrated in Fig. 10(B)). In turn, the DoM(0) column player's q-value computation (Eq. 45) takes as input these false beliefs. This results in its selecting the column that, instead of yielding it a 0 reward, is actually the least favourable column (M in G^1, L in G^2) yielding it a negative utility of (-4). This substantially benefits the deceptive DoM(1) row player, as evident in Fig. 10(D) in terms of the difference in reward between row and column players—in this case 7 points on average.

Lastly, the DoM(2) models the row player as a DoM(1). It inverts its actions to make inference about the payoff matrix from and act optimally, similarly to the DoM(0). Using its nested model of the DoM(1) row player, the DoM(2) column player "calls the bluff" and makes correct inferences about the payout matrix Fig. 10(C). Its policy exploits the DoM(1) ruse against itself, by picking the right column, yielding it a reward of 2 and a reward of (-2) to μ_1. Lacking the capacity to model such counter-deceptive behaviour, the DoM(1) erroneously attributes this behaviour to the SoftMax policy, and its nested beliefs about the column player beliefs are the distorted DoM(0) beliefs. This inability to resist manipulation by the higher DoM column player yields a high income gap, as illustrated in Fig. 10(D).

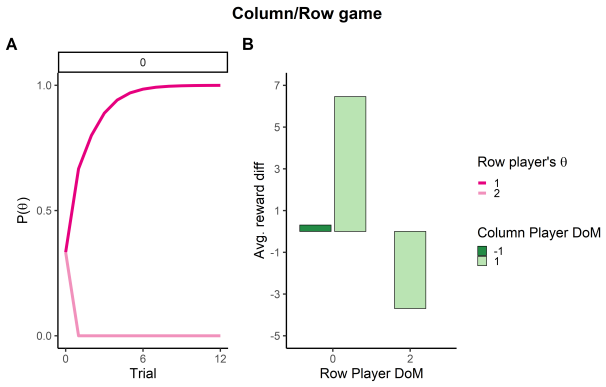

Fig. 11. **Row/Column zero-sum game**: **(A)** Illustration of DoM(0) Bayesian-IRL. Given the row player's actions, the DoM(0) column player quickly detects the true type (and payoff matrix). **(B)** Advantage of high DoM level: in each of the simulated dyads, the higher DoM agent has an edge. Its ability to simulate and predict its opponent's behaviour allows it to gain excess wealth

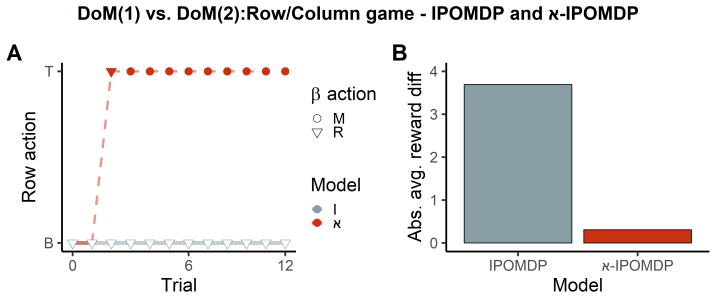

Fig. 12. **Effect of ℵ-IPOMDP in a zero sum game**: **(A)** Interacting with a $\theta_\mu = 1$, the DoM(0) (left column) is deceived by μ's actions and form false beliefs. However, the DoM(2) utilises its nested model to read through the bluff and correctly identifies μ's type. **(B)**(grey line) In turn, the DoM(2) policy exploits of the DoM(1) ruse against it. Augmented with ℵ-IPOMDP (red line), the DoM(1) infers this unexpected behaviour as a sign of an external entity, triggering the ℵ-mechanism, marked by the dashed line. Its ℵ-policy causes the DoM(2) column player to adapt and alter its abusive behaviour. **(C)** The effect of the ℵ-IPOMDP in this task is illustrated via the reduction in the average absolute reward difference.

6.4.1 Manipulation, Counter-Manipulation and Counter-Counter Manipulation. The DoM(1) row player's policy exposes it to potential risk, as it plays the "risky" row. For example, in G^1 the right column yields it a negative reward. However, given its ability to predict the DoM(0) row player's action this risk is mitigated. As presented above, the DoM(2) column player tricks the trickster by learning to select the right column. The DoM(2) take advantage of the DoM(1) inability to model its behaviour as deceptive and resent it, which yields it a reward of 2 at each trial, depicted in Fig. 11(B). We solve this issue by simulating the game again using the ℵ-IPOMDP framework. Due to the reward masking, the DoM(1) detects that they are matched with an external opponent only through the typical-set component. Due to the expected deterministic behaviour of the column player, we use the δ-typicality algorithm in this task. Identifying that they are outmatched, the ℵ-policy is to play the MinMax policy—selecting the row that yields the highest-lowest reward. Interestingly, in this task, this policy is similar to the optimal policy of the "truth-telling" DoM(−1) agent. In this case the DoM(2) response is to select the column which yields it the highest reward—namely the one that yields it a zero reward, as evident in 12(C).

There is a similar outcome in the zero-sum game. Here, the behaviour of the deceptive DoM(2) column layer described above would be highly non typical for DoM(0) column player, triggering the ℵ-mechanism. The DoM(1) MinMax ℵ-policy, i.e., playing truthfully, causes the DoM(2) to adapt its behaviour appropriately (Fig. 12(B)). In this case, both parties get 0 reward, which drops the average absolute reward difference compared to the IPOMDP case, as illustrated in Fig. 12(C).

Received 20 May 2025; accepted 5 January 2025