SiRToM: An implementation of computational Theory of Mind in R

K. Enevoldsen & P. Waade

Supervisor: R. Fusaroli

Bachelor project in Cognitive Science

School of Communication and Culture, University of Aarhus

20. December 2018

Regarding Initials used in the Paper

KCE: K. C. Enevoldsen
PTW: P. T. Waade

We regard this paper as a joint effort. However we have tried to indicate who contributed the majority for each section. When initials are comma separated, the first person mentioned is the primary contributor for the following section, albeit the whole paper has been heavily edited by both authors. Where initials are separated by an ampersand (&), both authors contributed evenly.

Summary

PTW & KCE

Computational implementations of Theory of Mind (ToM), the ability to attribute mental states to others, has been used to investigate a variety of issues. This includes the effect of framing effects on, or inter-species differences in, ability to do ToM (Devaine et al., 2014a, 2017), ToM in autists (d'Arc et al., 2018), or providing an explanation for the apparent limits on human ability to do ToM recursively (Devaine et al., 2014b). It has been implemented in the VBA package for Matlab (Daunizeau et al., 2014), but not in any free and open-source software. Therefore this thesis presents the Simulations in R: Theory of Mind (SiRToM) package, currently under development, for the free statistical software R (R Core Team, 2013).

The SiRToM package provides accessible tools for running agent-based models in a game theory context, and allows the implementation of a computational model of ToM, either in agent-based models or in interaction with a human player. The implementation of the ToM model was originally proposed by Yoshida et al. (2008), and was developed by drawing on the Free Energy Principle (Friston, 2010) to its current form as it is in Devaine et al. (2017), where it is generalized to any 2-player game which can be operationalized as a 2-by-2 payoff matrix. Importantly, the ToM implementation introduces a sophistication level k, which determines how many recursive simulations of its opponent it can perform, hereby assuming bounded rationality (Kahneman, 2003). An agent using the ToM model, denoted as k-ToM, uses a variational Bayes Laplace approximation (Daunizeau, 2017b) on a turn-by-turn basis to infer its opponent's model parameters and sophistication level, based on which it predicts the opponent's choice and acts accordingly.

An agent-based model simulation using the competitive matching pennies game was done to perform a preliminary investigation of the behaviour of the k-ToM model. Most importantly, it was found that k-ToM's prior beliefs about its opponent have a notable effect on its performance, even over many trials, warranting further research into how its priors should be formed. Various ways are suggested in which the SiRToM package and the k-ToM model could be applied and developed further, as well as a discussion on how to make it broadly available, so as to scaffold future research using computational ToM models.

Contents

1 Introduction

KCE & PTW

Theory of Mind (ToM), the ability to attribute mental states to others, has long been a subject of research, but only in recent years has there been made computational models of the processes behind it. This has provided a new framework for investigating Theory of Mind processes in humans and animals through agent-based simulation studies, modelling of human behaviour, and investigation of human interaction with simulated ToM agents (Devaine et al., 2014b,a, 2017). Artificial agents using ToM-like processes has even been used in recent A.I. applications, giving high performance on advanced cooperative games, and working towards applications in human-computer-interfaces and advances in interpretable A.I. (Rabinowitz et al., 2018; Foerster et al., 2018). Currently, the only way of conveniently modelling ToM simulations is the VBA toolbox extension for Matlab (Daunizeau et al., 2014), while no implementiation currently exists for the free to use statistical softwares R (R Core Team, 2013) nor other open source software. R does have extensions for agent-based modelling like the RNetlogo (Thiele, 2014) and SpaDes (Chubaty and McIntire, 2017) packages, but they do not allow for easy modelling of ToM agents, and do not allow for interaction between human players and simulated agents. Therefore, we present the Simulations in R: Theory of Mind (SiRToM) package for R, an implementation of a computational ToM model in game theory settings, either in an agent-based model or in interaction with human players. The following is a short introduction to Theory of Mind, game theory and agent-based modelling. It ends with an account of the computational implementation of Theory of Mind used in the SiRToM package, combining aspects of the previous sections.

2 Theory

2.1 Theory of Mind

PTW, KCE

Theory of Mind (ToM) is the ability to infer and attribute mental states to others, including beliefs, desires, intentions and emotions. It is thought to be a fundamental component of many social skills and interactions, like cooperation, deception and teaching. There are several conceptualizations of how ToM happens in humans, combinations of which has also been suggested (Goldman et al., 2012). The term ”Theory of Mind” comes from the original account, which describes it as a process similar to scientific discovery where relations are learned statistically between others' behaviour and mental states (Goldman et al., 2012). An alternative nativist account posits that ToM abilities are not learned, but become active at a biologically determined time, and has a modularity separate from more general processes (Goldman et al., 2012). A third rationality-centred approach posits that others' actions are predicted by assuming them to be rational agents who decide optimally in a goal-directed manner (Goldman et al., 2012). Lastly, the simulation-theory holds that ToM is performed by mentally simulating oneself in the position and with the mental states of others (Goldman et al., 2012).

ToM was originally investigated when trying to determine when children during development become able to attribute mental states to other people, take their perspective, and vocalize about the knowledge states of others

(Mossler et al., 1976; Marvin et al., 1976). Similar tasks were also given to chimpanzees, to attempt to infer whether apes are able to attribute mental states to humans and in general, or whether they understand others as non-sentient phenomena (Premack and Woodruff, 1978).

ToM has been seen as a core component of deceptive abilities, making it critical in many social-strategic situations. Early studies investigated whether primates are able to strategically withhold or give information, finding indication that they are able to perform some basic ToM (Woodruff and Premack, 1979). ToM has also been shown necessary for predicting others' actions, making it highly relevant in strategic social situations in general, and specifically relevant in many competitive games. Shultz and Cloghesy (1981) used card games to investigate the ability to do recursive ToM, i.e. taking the mental perspective of someone who is taking the mental perspective of another, possibly oneself, e.g. *"I think that you think, that I think"*. This allows for predicting the other's prediction of oneself and correspondingly foresee their next action. The term n^{th} order theory of mind then denotes how many levels of recursion is done.

Traditionally, ToM has been investigated using the False Belief task, where participants have to report predictions about the actions or knowledge of another person who has inaccurate information. For example, a desirable object is hidden while an individual is present, then unbeknownst to the individual moved to another position. The tendency of young children to predict that the individual will look for the object in its current position, and not its original, has been used to show that young children are unable to perform ToM. It has been noted that there are several issues with the False Belief task. Failing the test might simply be caused by the higher processing demands related to reasoning about counterfactuals and holding multiple knowledge states simultaneously in working memory, and not because of an inability to do ToM (Bloom and German, 2000). Bloom and German (2000) argues that being able to perform ToM does not necessarily require the ability to report it or reason about it. Indeed, it has been shown that there are implicit, unconscious components of ToM, which for example have been investigated using eye tracking in the False Belief task (Schneider et al., 2014). Another issue is that the False Belief task doesn't measure affective ToM, i.e. the attribution and appreciation of emotional states in others, which has been suggested as another sub-component of ToM. Methods have been developed for investigating affective ToM separately from the normal, or cognitive, ToM. This has been done by presenting socially embarrassing stories where understanding them requires attribution of emotional states, as opposed to stories more similar to the False Belief task, which only requires the attribution of knowledge and intentional states (Kalbe et al., 2007). The facial expression recognition tasks "Reading the Mind in the Eyes" and the "DANVA2-AF" have also been used to measure affective ToM capabilities, and the validated Yoni task has been developed to distinguish both between cognitive and affective ToM, and between first and second order ToM (Kidd and Castano, 2013). This has been used to investigate which things affect different kinds of ToM separately, like reading fiction (Kidd and Castano, 2013). However, these methods do not clearly measure, nor allow explicit modelling, the function of ToM in repetitive, strategically interactive situations, which would include most, if not all, social situations where the person herself takes part in the interaction. Game theory is another experimental and explanatory field designed to model exactly these kinds of situations, and which has also been used to understand ToM, often instantiated in agent-based modelling.

2.2 Game Theory

KCE, PTW

Game theory is an approach which seeks to formally model interactions between decision-making agents, be it social, economical or biological. This could include price negotiations in a market or choosing whether or not to go to a party. Game theory has been used to illuminate economic, behavioral and biological phenomena, such as causes for oligopoly between firms (Corts, 1998), the rise of cooperative norms (Gauthier, 1986; Epstein, 1998), and the evolutionary incentive for inefficient weapons for within-species competition (Smith and Price, 1973). A central assumption of game theory is rationality of the agents, where rationality is defined as the best action according to the agent's preferences (Osborne, 2004; Binmore, 2007). As such, if an agent's preference for the outcome of a given choice c^1 is higher than for another choice c^2, we expect the agent's utility of c^1 to be higher than c^2, eg:

$$U(c^1) > U(c^2)$$

If and only if the agent prefers c^1 to c^2. U is the payoff function (Osborne, 2004), which returns a numerical payoff given a choice. It is thus possible to attribute numerical value to specific outcomes. For example, re-examine the case of whether or not to go to a party. Suppose we have two agents each associating higher value with going to the party together as compared to staying home, but also prefer staying home over going alone. This situation can be formalized in the following payoff matrix, where each axis represent the choice of each individual:

Party Dilemma		Player 2	
		Party	Home
Player 1	Party	5,5	0,3
	Home	3,0	3,3

Note that the differences in values of each option are of an arbitrary size, but respects the relations given in the party dilemma. Expanding upon the payoff function, we can now denote that:

$$R = U(c^{self}, c^{op})$$

Where c^{self}, c^{op} denotes the choice of the agent and opposing agent respectively and the output R is the reward associated with the choices. Henceforth choices will be referred to in a binomial fashion using 1 or 0, where in the current case, 1 is going to the party and 0 is staying at home. As an example, we see that $U(c^{self} = 1, c^{op} = 1) = 5$. Similarly, for simplicity, opposing agents will henceforth be referred to as the opponent, even though the situation described might allow for cooperative behaviour. This above mentioned payoff matrix is traditionally also known as the stag hunt, in which the players choose between hunting a stag together or a rabbit alone, and is importantly a general model for situations in which players can obtain higher reward if both players cooperate with one another, but can obtain a smaller, albeit certain, reward by going alone. As such, by studying the stag hunt or party dilemma, we study a broader array of real interactions between two individuals in a simplified environment.

Many types of interactions between multiple agents can be described in game theoretical terms. However, in this thesis, we only concern ourselves with 2-player games with simultaneous turns and a 2-by-2 payoff matrix, and where agents are only concerned with maximizing a score, albeit as previously mentioned a score is simply an abstraction of the agent's preference, be it biological fitness or social status. This framework has its restrictions, but using consecutive turns and multiple interactions have proved to be a powerful way of describing a wide variety of situations, such as the emergence of cooperation (Epstein, 1998) and the emergence of other evolutionary stable strategies, such as a balanced ratio of sexes (Hamilton, 1967). This is examined further in *2.3 Agent-Based Models*.

The SiRToM package supplies a wide range of possible payoff matrices, we will primarily be using the anti-symmetric zero-sum matching pennies game. The matching pennies game is a purely competitive game where gains are directly associated with opponent's losses and has previously been used to examine ToM (Devaine et al., 2014a,b, 2017). It is operationalized as the following payoff matrix:

Matching Pennies		Player 2	
		Right	Left
Player 1	Right	1,-1	-1,1
	Left	-1,1	1,-1

The payoff matrix symbolizes a game in which one player hides a coin in either its left or right hand, and is rewarded if the opponent cannot guess the coin's location. The opponent gets the point if she guesses correctly. One of the features of the matching pennies games which makes ot especially interesting is that there exists no pure strategy Nash equilibrium, i.e. there exist no two strategies such that neither player would want to switch (Nash, 1951; Devaine et al., 2014b). This necessitates predicting the opponent's next move, consequently making the game is ideal for investigating ToM.

2.3 Agent-Based Models

KCE, PTW

Agent-based modelling (ABM) or agent-based simulation is a computational approach to analyzing complex behaviour by simulating interacting autonomous agents with individual goals and strategies. Examples of goals in contexts could be goals such as maximizing a score, reproducing, or finding a comfortable neighborhood (Bowles and Gintis, 2013). ABM was initially introduced to behavioural sciences by Thomas Schelling's (1978) study on neighborhood segregation. Using a chessboard, pennies and nickels, Schelling simulated how even a small preference (one third) for living with groups of similar ethnicity, social or economic status lead to neighborhood segregation.

ABM has also been used to simulate the workings of the stock market (Ehrentreich, 2007), the formation and behaviour of fish schools (Huth and Wissel, 1992), the emergence of price systems and price convergence (Gintis, 2006, 2007), simulating the demography of the ancient Anasazi civilizations (Dean et al., 2000; Axtell et al., 2002), modelling civil violence and decentralized rebellions (Epstein, 2002) and simulating the spread of diseases (Epstein et al., 2002). As previously mentioned, ABM often utilizes elements from game theory (see *2.2 Game Theory*) by simulating an environment in which multiple games take place. This allows for experimental manipulations, such as examining how states of informational openness can increase the likelihood of cooperation (Skewes et al., 2017).

Implementing ABM includes a variety of stochastic processes, e.g. agents are competing against each other structured in a randomized fashion, and select strategies or make choices probabilistically. Consequently, ABM often yield varied results, as opposed to analytic calculations of similar situations, which could suggest that analytic approaches are more suitable. However, Bowles and Gintis (2013) notes that many scenarios are too complicated to allow for an analytical approach. Even when analytical approaches are viable, these approaches often require assumptions such as an infinite number of agents or repetitions, which have proved problematic compared to ABM, as demonstrated by Durrett and Levin (1994). Durrett and Levin (1994) demonstrates that ABM have more reasonable assumptions, which closer resembles the actual situation, and yield models and outcomes more akin to the modelled scenario.

While the specific game played in an ABM is important, so is the environment in which the game is played. The environment can symbolize a wide variety of social, economical or biological contexts, operationalized, for example, as a network of interacting agents, or by using a grid-like structure in which agents only interact with other adjacent agents, or agents within their defined vicinity. An example is Schelling's (1978) simulation of neighborhood segregation, in which the agents move on a 2 dimensional grid, in which a cell symbolizes a housing area and the environment as a whole symbolizes a city. While the environment includes the structure of a competition, the environment can also include elements such as types of other competing agents or the number of iterations. The importance of the environment is seen in Axelrod and Hamilton's (1981) seminal study on multiple iterations of the prisoner's dilemma (see *Appendix*), in which the environment affects the effectiveness of different strategies. Axelrod's (1981) ABM used a round robin tournament structure in which each agent competed against all other agents. Axelrod's (1981) tournament famously showed that the simple and cooperative strategy Tit-for-Tat (see

3.1 Simple Agents) outperformed all other strategies submitted to the tournament. However, while Tit-for-Tat is indeed ideal in a variegated environment, it looses in an environment dominated by non-cooperative agents, or in an environments in which each agent is highly unlikely to compete more than once (Axelrod, 2006).

2.4 Computational Modelling of Theory of Mind

PTW, KCE

The computational implementation of ToM used in SiRToM package is methodologically and conceptually based on the Free Energy Principle, as presented by Karl Friston (Friston, 2010). The Free Energy Principle is a theoretical framework aiming to unify many theories of the brain. It states that any self-organizing system is a generative predictive model of its surrounding world, which must maximize the model evidence of its own existence, identical to minimizing its information theoretical surprise. The minimization of variational free energy (an information theoretic quantity) is then used to implicitly minimize surprise (Friston, 2010).

Variational free energy is a function of two things: the sensory states of the organism, and its internal states (denoted μ) which encodes the recognition density, where the recognition density is a probability distribution of causes of sensory states, an inverted generative model, including the model parameters (denoted θ) (Friston, 2010). Variational free energy has multiple compatible definitions. One is the surprise added to the (Kullback–Leibler) divergence between the recognition density and the probability distribution of the cause of sensory data conditional on sensory data, i.e. the best guess of what caused the sensory data after receiving it. Changing the recognition density to be equal to the conditional probability then minimizes free energy, and thus, implicitly, surprise. Another is the accuracy subtracted from the Bayesian surprise, i.e. the difference between beliefs prior to receiving sensory data and the recognition density or posterior belief. Reducing the Bayesian surprise, by sampling only sensory input corresponding to the prior, then also must maximize accuracy. The first of the two definitions makes it clear how changing internal states in order to change the recognition density is a way of minimizing free energy, while the second definition makes it clear how acting to change the world, and thus changing sensory inputs, is another way of minimizing free energy, called active inference.[1]

The Bayesian Brain Hypothesis conform with the free energy principle. It postulates that the brain has a model of the world, which it optimizes in an approximately Bayesian manner by generating predicted sensory inputs and comparing them to the actual sensory inputs. This process of inverting the mapping from cause to data, to be able to infer causes given data, is the same as minimizing the difference between the recognition density and the posterior belief conditional on the data, in order to minimize free energy. Here, the internal states that encode the recognition density are then neuronal activity (Friston et al., 2006a). The variational Bayes technique is, compared to traditional Bayesian sampling methods, a more computationally feasible approximation technique for modeling these Bayesian-like updates in the brain (Friston et al., 2006a; Daunizeau, 2017b). This possibility for an explicit mathematical modelling of the model-updating in the brain which is also neurologically feasible, allows for modelling and simulating processes like ToM in humans. While the application of the Free Energy Principle has been subject to discussion (Friston et al., 2012), it has among other things resulted in computational models of

[1] See Friston (2010) for a more detailed description of the Free Energy Principle and its relations to other theories.

ToM, which, independently of an adherence to the Free Energy Principle, can be used to model Theory of Mind computationally.

The implementation of ToM applied in this thesis was derived from Devaine et al. (2017). However, it was originally developed by Yoshida et al. (2008), drawing on optimal control theory and game theory. Computational implementation of ToM models encounter a problem when an agent simulates its opponent simulating the agent, which in turn is simulated to be simulating the opponent, ad infinitum. This infinite recursive problem is solved by introducing an assumption of bounded rationality (Kahneman, 2003; Yoshida et al., 2008), where the amount of recursions possible of a person or a simulated agent is bounded (see *3.2 Computational model* for the mathematical implementation). The amount of recursions possible, traditionally called the n^{th} order ToM, is in the computational implementation called the sophistication level, and is denoted with a k.

Expanding on Yoshida et al.'s (2008) model, Devaine et al. (2014b) applies earlier work on meta-Bayesian updating within the Free Energy Principle framework (Daunizeau et al., 2010a,b) on the modelling of Theory of Mind to develop a meta-Bayesian variational Bayes implementation. The meta-Bayesian ToM agent now learns about its opponent's beliefs in a Bayes-optimal manner, which includes learning about the opponent learning about ToM itself, assuming that its opponent also learn in a Bayes-optimal manner. Devaine et al. (2014b) also introduced the term k-ToM, denoting a ToM agents of sophistication level k. By convention, 0-ToM then denotes an agent which does not attribute mental states to its opponent, but only estimates the opponent's choice probability from the frequency of its choices, as if it was a random phenomenon. As a consequence of the bounded rationality assumption (Yoshida et al., 2008), a k-ToM agent can only simulate agents of lower sophistication levels. A k-ToM's opponent's sophistication level is denoted κ, defined as $\kappa \in [0, k-1]$, which means that a 1-ToM by definition assumes its opponent to be a 0-ToM.

This ToM model, implemented in the VBA package for Matlab (Daunizeau et al., 2014), has been used by Devaine et al. (2014b) to investigate why humans seem to be limited to ToM sophistication levels around 2, even though higher levels has long been thought to be more advantageous. Devaine et al. (2014b) show that higher Theory of Mind sophistication levels have a non-trivial information cost in competitive settings. The more complex models have more parameters to estimate, making them slower to adapt and more uncertain, hereby preventing them from exploiting lower levels efficiently. Furthermore it was found that in a cooperative settings, heterogeneity in sophistication levels is an advantage, because the more sophisticated agents can efficiently predict the simpler agents, and therefore teach them to cooperate. If two sophisticated agents of the same level play a cooperative game, they will estimate each other as, and act like, a heterogeneous set of lower level agents, to the same effect. Using evolutionary game theory simulations (see Maynard Smith (1982) for an introduction to the method), Devaine et al. (2014b) finds the evolutionary stable mixture of ToM levels to be a mixture of levels 1 and 2, where 0-ToM agents are quickly outcompeted by the more sophisticated agents. k-ToM agents with $k > 2$ slowly go extinct because of too high informational costs that prevent them from efficiently exploiting lower levels (Devaine et al., 2014b).

Additionally, Devaine et al. (2014a) shows that people perform better against artificial ToM agents when they think they play against humans, as compared to when they interact with a slot machine, and that they seem to play out evenly against 2-ToM. By comparing data generated by generative models to the behavioural data of the human participants, (Devaine et al., 2014a) also shows that k-ToM models with $k > 0$ only explains the human behaviour

when participants think they are playing against humans. Consequently, it seems that *a priori* attribution of mental states decides which strategy humans use, when trying to predict the behaviour of others and the physical world, and that without attribution of mental states, people cannot decipher intentional states. In a follow-up study, it has also been shown that this framing effect doesn't influence participants with autism spectrum disorder (ASD), which d'Arc et al. (2018) argues to implicate that ASD affects ToM abilities.

The ToM implementation was also applied to examine ToM capabilities among apes, as to investigate ToM relation to group sizes and brain size (Devaine et al., 2017). Results seem to indicate that ToM capabilities are not evolutionary caused by large group sizes placing a demand on better social abilities, but rather by brain size, upon which it is scaffolded (Devaine et al., 2017). Furthermore an evolutionary gap between humans and other great apes were found, where other great apes are predicted well by an influence learning model designed to lie be between a 0-ToM and 1-ToM, while humans are dominated by 2-ToM strategies (Devaine et al., 2014a, 2017).

Several other computational implementations of Theory of Mind have been proposed which do not rely on the Free Energy Principle, and do not use variational Bayes to estimate the opponent's parameter values. One such model was designed for multiplayer games, where the agent utilize a ToM-like estimation of the group as a whole and the influence of its own choices on the group (Khalvati et al., 2018). This model outperformed all previous models in predicting participants' behaviour in a Public Goods game (Khalvati et al., 2018). More technologically oriented approaches also include neural network based ToM models (Rabinowitz et al., 2018; Foerster et al., 2018). One such multi-agent model is meta-ToM deep neural network, which estimates the strategies of encountered agents by forming strong general priors, helping it predict new agents (Rabinowitz et al., 2018). Another multi-agent model has been shown to perform well on more advanced cooperative games, where the ability to predict each other is crucial for success (Foerster et al., 2018).

While the neural network models have high performance on advanced game, they have not been directly related to internal processes behind human ToM processes. Comparably, the implementation by Devaine et al. (2017) has been used to successfully model human behaviour (Devaine et al., 2014a,b).

3 Methods

PTW & KCE

This thesis presents the Simulations in R: Theory of Mind (SiRToM) package, which is an extension software for the free-to-use statistical programming language R (R Core Team, 2013). The SiRToM package is an easy tool to create agent-based model simulations in a game theory framework, with an emphasis on making computational Theory of Mind agents accessible. The code behind the package is publicly available. What follows is a description of the main functionalities currently implemented in the package, and how they have been used to do some preliminary simulation-based investigation into the behaviour of the computational Theory of Mind (ToM) model.

As previously mentioned (see *2.3 Agent-Based Models*) the environment in which the agents compete is highly relevant for the outcome of the competition. The SiRToM package currently implements a round robin tournament structure in which specified agents compete against all other specified agents. Additionally, the package has an option in which each agent is paired up at random as well as a function for a single match-up, allowing for other

environments to be built e.g. a grid structure. SiRToM can currently simulate any simultaneous 2-player game using a 2-by-2 payoff matrix. A custom payoff matrix can be specified, but SiRToM has a wide array of pre-made matrices, including, but not limited to, the prisoner's dilemma, the stag hunt and the matching pennies games [2]. It is possible to specify number of trials agents compete as well as number of times it should be simulated. Apart from the Theory of Mind implementation, SiRToM has several simple strategies that agents can employ, which will be presented in the following sections.

3.1 Simple Agents

PTW, KCE

Many different simple heuristic-based strategies have historically been used in agent-based modelling (ABM), showing that even simple heuristics can be used to produce efficient and sophisticated behaviour (see *2.3 Agent-Based Models*). Some of those strategies have been included in the package. While these strategies do not use ToM, they can generate ToM-like behaviour without needing recursive simulation of the opponent. The heuristic strategies are the following: Random Bias (RB), Tit-for-Tat (TFT) and Win-stay Loose-shift (WSLS).

The RB strategy is the simplest of the strategies. The agent simply makes a random choice with a potential bias, according to its probability parameter. This is shown in the following equation:

$$P(c_t^{self} = 1) = p \tag{1}$$

Where $P(c_t^{self} = 1)$ is RB's probability of choosing 1, and p is RB's probability parameter. Note that, remarkably, in competitive games like the matching pennies game, a RB with p=0.5 is the Nash equilibrium (Nash, 1951), because any other strategy, regardless of complexity, will not be able to predict it above chance level. Consequently RB agent would tie against all other strategies. In the SiRToM package, RB agents uses probability parameters sampled from a normal distribution with mean 0.5 and a wide ensuring parameter values from most of the parameter space.

Slightly more advanced is the TFT model known from Axelrod and Hamilton's (1981) tournament, which copies its opponent's last move. For flexibility, we have added a parameter-given probability to TFT's choice. This can be seen in the following equation:

$$P(c_t^{self} = c_{t-1}^{op}) = p \tag{2}$$

Where $P(c_t^{self} = c_{t-1}^{op})$ is TFT's probability of selecting the same option as the opponent did last trial, and p is TFT's probability parameter. Note that, traditionally, TFT has had a probability parameter of 1. As previously mentioned TFT performed remarkably well in the prisoner's dilemma (see *Appendix*), where its simple heuristic enables cooperation without being easily exploitable (Axelrod and Hamilton, 1981). As the focus of this thesis is on

[2] for a complete list see https://github.com/KennethEnevoldsen/SiRToM

the matching pennies game in which there is no cooparative dimensions, the TFT is excluded from further analysis as its simple behaviour is easy to predict, and to exploit.

The heuristic strategy Win-Stay Lose-Shift (WSLS) which outperforms the TFT in the prisoner's dilemma was suggested by Nowak and Sigmund (1993). WSLS outperformed TFT as it was more robust and able to exploit unconditional operators. The WSLS makes the same choice as last trial if it wins, but switches strategy if it loses. This strategy approximates a maximum-learning reinforcement learning rule (Devaine et al., 2017). Winning has earlier been defined as getting a positive score, yielding the following decision rule:

$$c_t^{self} = \begin{cases} c_{t-1}^{self} & if \quad R_{t-1} \quad is \quad positive \\ (1 - c_{t-1}^{self}) & if \quad R_{t-1} \quad is \quad negative \end{cases} \tag{3a}$$

Where c_t^{self} is WSLS's own choice on trial t, and R_{t-1} is the reward gained on last trial. The WSLS heuristic is also useful for competitive games such as the matching pennies game, because it is able to capitalize on the bias of an RB agent.

A problem with this implementation is the WSLS strategy's definition of winning which generalizes poorly across games, especially those with payoff matrices holding only positive, or only negative, values, like the stag hunt game. To make the WSLS strategy generalize, we have changed the definition of winning to getting a score above the mean score across the payoff matrix. In the penny game and the prisoners dilemma this makes no difference, but in the stag hunt game it allows WSLS to switch away from situations where the opponent does not cooperate. Inserting this new definition, and also introducing a probability parameters, results in the following WSLS decision rule:

$$P(c_t^{self} = c_{t-1}^{self}) = \begin{cases} Wp & if \quad R_{t-1} > \bar{R} \\ 0.5 & if \quad R_{t-1} = \bar{R} \\ (1 - Lp) & if \quad R_{t-1} < \bar{R} \end{cases} \tag{3b}$$

Where $P(c_t^{self} = c_{t-1}^{self})$ is WSLS agent's probability of repeating its own choice, $\bar{R}$ is the mean payoff across the payoff matrix, and Wp and Lp are its probability parameters deciding its probability of staying when winning and shifting when losing. While this version of the WSLS strategy does not generalize well to all situations, for example does it not make distinction between smaller and larger victories, it is now able to play most games implemented as default options in the SirToM package. As default in the SiRToM package, TFT and WSLS agents uses probability parameter sampled from a normal distribution with mean 0.9 and a small standard deviation.

3.2 Computational model

PTW & KCE

What follows is the implementation of Theory of Mind (ToM) used in this thesis, also called the k-ToM model. It is based on the trial implementation used in (Devaine et al., 2017), closely following the implementation in the VBA toolbox package for Matlab (Daunizeau et al., 2014), to allow for comparison. It has been expanded to its current form by using the variational Bayes Laplace approximation to estimate opponent parameter values, since there is a nonlinear relation between parameters and observed behaviour. The variational Bayes Laplace approximation [3] is essentially a way of approximating Bayes-optimal updates of unknown values with nonlinear relations to observed states. It is computationally feasible enough that it is a realistic model of the Bayesian brain, as opposed to traditional sampling methods (Friston et al., 2006a; Daunizeau, 2017b).

The following section contains a brief overview of the k-ToM model, after which each step and its mathematical formulation will be explained in depth.

All trial agents, regardless of sophistication level, seek to estimate the probability of the opponent choosing 1 P_t^{op} on the current trial, and based on that make their own decision. Agents of different sophistication levels makes their own decision in a similar way once they have estimated the probability of the opponent's choice, but they vary in how they make that estimation, that is, they have a similar decision process, but different learning processes, both of which are performed on a turn-by-turn basis. All ToM-agents have two parameters θ, a behavioural temperature β which randomizes behaviour, and a volatility σ which is ToM's assumption of noise in the opponent's behaviour.

As previously mentioned (*2.4 Computational Modelling of Theory of Mind*), k-ToM denotes a trial agent of sophistication level k. The 0-ToM agent then by convention denotes an agent which does not attribute mental states or intentions to its opponent, but assumes its opponent to be a RB agent, and tries to estimate its choice probability parameter p. It does this by using a variational Bayes Laplace approximation, but without having to assume nonlinearity, because the choice probability parameter p of a RB is equal to its choice probability. Once 0-ToM has estimated its opponent's choice probability parameter, it has also estimated its choice probability, and 0-ToM's learning process is concluded.

Once the opponent's choice probability P_t^{op} has been calculated, 0-ToM's decision process begins. First, the expected payoff of the agent itself choosing 1 ΔV is calculated, given the opponent's choice probability. This is inserted into a softmax decision rule, to calculate a probability of the agent itself choosing 1 $P(c^{self} = 1)$, which is finally evaluated to decide the agent's choice.

The k-ToM agent with $k > 0$ has a more complex learning process. It assumes its opponent to be a ToM agent with a lower sophistication level $\kappa \in [0, k-1]$, and must estimate the probability λ for the opponent having each of the possible levels κ. It must then, for each κ, estimate its opponent's parameter values θ, containing the β and σ values. It again does this by using a variational Bayes Laplace approximation, but this time more complex, because the relation between parameter values θ and the opponent's choice probability is non-linear. Once k-ToM has estimated its opponent's parameter values θ, it uses these to simulate its opponent's learning and decision

[3] See Beal (1998); Friston et al. (2006b); Daunizeau et al. (2009); Daunizeau (2017b) for a full technical derivation of the variational Bayes Laplace approximation.

processes, in order to learn its opponents choice probability P_t^{op}, again for each level κ. For a k-ToM with $k > 1$, this involves a simulation of the opponent simulating oneself, and thus includes recursive simulations of gradually lower levels, until the simulation of 0-ToM, where the recursion ends. The last part of k-ToM's learning process consists of calculating a gradient W of the relation between each of the estimated parameter values θ and the estimated opponent's choice probability P_t^{op}, again for each level κ. This is a necessary part of the nonlinear variational Bayes Laplace approximation for next trial.

k-ToM's decision process is similar to that of 0-ToM. But k-ToM has estimated an choice probability P_t^{op} for each of its opponent's levels κ, and must unite these estimates into a single value in order to complete the decision process. It does this by taking a mean of the estimated choice probabilities P_t^{op} for each level κ, weighted by the probability λ for the opponent having each level κ. Once the single choice probability P_t^{op} has been estimated, it is used to calculate the expected payoff ΔV and inserted into the softmax rule in the same way as in 0-ToM's decision rule, finally yielding k-ToM's own choice probability $P(c^{self} = 1)$.

This computational implementation of trial uses an assumption of bounded rationality (Kahneman, 2003), putting a bound on how many recursion an agent can do, which is determined by an agent's sophistication level k, which solves the issue of infinite recursion being possible. Because of using the variational Bayes Laplace approximation, the model also makes a Laplace-assumption of a Gaussian uncertainty distribution on all its estimates (Daunizeau, 2017b).

3.2.1 0-ToM

All ToM agents estimate their opponents' parameter values θ in order to learn the choice probability of their opponents P_t^{op}, but since 0-ToM assumes its opponent to use a RB strategy, the estimation of the probability parameter p and the choice probability P_t^{op} becomes identical. The choice probability parameter p is estimated as a normal distribution with mean μ and variance Σ, each of which are updated on a turn-by turn-basis, based on the opponent's last choice. This is done using a simple variational Bayes Laplace approximation for parameters with a linear relation to observed behaviour. In the graphical model[4] (figure 1), 0-ToM's learning and decision process can be seen. First the variance Σ is updated, then the mean estimate μ. This allows the estimation of the opponent's choice probability P^{op}, after which the expected payoff difference ΔV can be calculated and inserted in the softmax function to decide 0-ToM's own choice probability $P(c^{self} = 1)$.

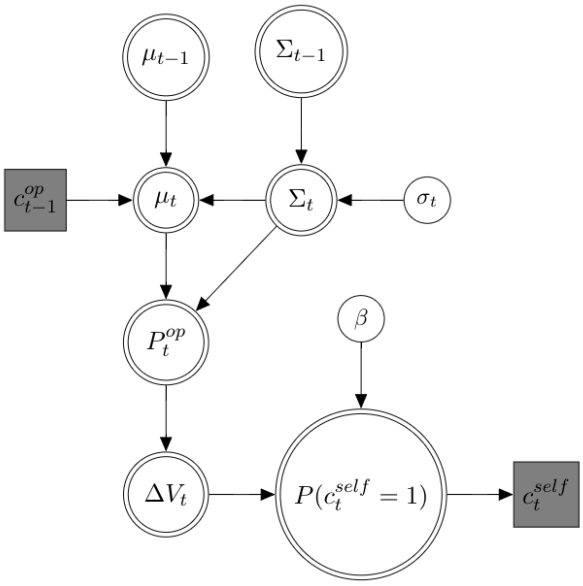

Figure 1: A graphical model of a single trial for the 0-ToM model. Shaded variables are observed, squares are discrete while circles are continuous, and double bordered variables are deterministic and unobserved.

The variance Σ is updated using the following equation:

$$\Sigma_t^0 \approx \frac{1}{\frac{1}{\Sigma_{t-1}^0 + \sigma^0} + s(\mu_{t-1}^0)(1 - s(\mu_{t-1}^0))} \tag{4}$$

Where μ_t^0 denotes 0-ToM's mean estimate in logodds at trial t of the opponent's probability parameter p, and Σ_t^0 denotes the subjective uncertainty of the parameter estimate at trial t, and where $t-1$ indicates the previous turn. s is the sigmoid function, and the expression $s(\mu)$ is 0-ToM's estimate in probability of opponent's choice probability P_t^{op}, without taking uncertainty Σ into account. Note that a μ close to chance level results in a lower updated variance Σ. σ^0 denotes 0-ToM's volatility parameter, which captures her prior assumptions on how much

[4] See Bartlema et al. (2014) for an introduction to graphical models.

the opponent's parameters varies with time. (See Mathys et al. (2011)) for a model where σ is learned instead of assumed). The volatility parameter σ controls the updating of variance Σ, where a higher volatility results in a higher updated variance on every trial. Together, the size of σ and μ creates a dynamic lower bound for variance Σ.

After Σ has been updated, it is used for updating the mean estimate μ of the opponent's choice probability parameter p by insertion into the following equation:

$$\mu_t^0 \approx \mu_{t-1}^0 + \Sigma_t^0(c_{t-1}^{op} - s(\mu_{t-1}^0)) \tag{5}$$

Where c_t^{op} denotes the opponent's choice at trial t. Consequently, the term $c_{t-1}^{op} - s(\mu_{t-1}^0)$ becomes the prediction error on last trial, which is added to the previous mean to update it. It is weighted by the variance Σ, meaning that very uncertain beliefs are affected more by new data than very certain ones.

After having updated the mean and variance of the estimate of the opponent's choice probability, 0-ToM uses the following equation to make its estimate the opponent of choosing 1 $P(c^{op} = 1)$:

$$p_t^{op} \approx s\left(\frac{\mu_t^0}{\sqrt{1 + (\Sigma_t^0 + \sigma^0)3/\pi^2}}\right) \tag{6a}$$

Where P_t^{op} is the estimated probability of the opponent choosing 1. Note that higher variance Σ and volatility σ values result in the choice probability estimates P_t^{op} closer to chance level. This means that high subjective uncertainty and assuming more noise in the opponent's behaviour makes 0-ToM's estimates of its opponent's choice probability less extreme, and thus 0-ToM's own choices more random, hereby preventing overfitting.

Importantly, while 6a is the theoretically derived equation (Devaine et al., 2017), the implementation used in the VBA package (Daunizeau et al., 2014) uses an approximation to avoid identifiability issues:

$$p_t^{op} \approx s\left(\frac{\mu_t^0}{\sqrt{1 + a \cdot \Sigma_t^0}}\right) \tag{6b}$$

Where, a=0.36 is an approximation which contains within it the volatility parameter σ. This means that the volatility parameter σ is held constant. Note that small uncertainties result in estimates of opponent choice probabilities P_t^{op} closer to the mean μ, i.e. $\Sigma \to 0 \Rightarrow P_t^{op} \to \mu$.

The two equations 6a and 6b give similar results when σ values are below 1. This means that large volatility values do not affect the choice probability estimate directly, but only through 4. In initial simulations, using 6a has a tendency to yield extreme parameter estimates because of identifiability issues, especially in the more complex k-ToM model.

The SiRToM package is able to use both variants, but similarly to the VBA implementation (Daunizeau et al., 2014), it defaults to using eq. 6b, since it gives better performance and more stable results.

The probability P_t^{op} of the opponent choosing 1 is estimated, and 0-ToM's learning process is concluded. Now, as the first step in 0-ToM's decision process, P_t^{op} is inserted into the expected payoff function, shown below:

$$\Delta V_t = p_t^{op}(U(1,1) - U(0,1)) + (1 - p_t^{op})(U(1,0) - U(0,0)) \tag{7}$$

Where ΔV_t is o-ToM's expected payoff of choosing 1 relative to 0 on the current trial t. The notation $U(c^{self}, c^{op})$ denotes the payoff function, which returns the reward R given a payoff matrix and the (hypothetical) choices of 0-ToM herself c^{self} and the opponent c^{op} (see *2.2 ReferencesGT* for an explanation). This equation essentially finds the payoff difference of choosing 1 relative to 0 given both possible opponent choices, and sums them weighted by the probability of the opponent making that choice.

To calculate 0-ToM's own probability of choosing 1, 0-ToM's expected payoff of choosing 1, ΔV_t ,is now inserted in the softmax decision rule, as shown below:

$$P(c_t^{self} = 1) = \frac{1}{1 + exp(-\frac{\Delta V_t}{\beta^0})} \tag{8}$$

Where $P(c_t^{self} = 1)$ is 0-ToM's probability of choosing 1 on the current trial t. β^0 then denotes 0-ToM's behavioural temperature parameter, which randomizes behaviour. An expected payoff of 0, i.e. equal values of choosing 1 or 0, results in a random choice, $P(c_t^{self} = 1) = 0.5$. Higher expected values then result in higher probabilities of choosing 1, in a sigmoidal manner asymptotic to 1 and 0. Higher β values makes choice probabilities closer to 0.5, i.e. increases exploration. The softmax choice rule has previously proven efficient in game theory and in modelling choices on tasks such as the Iowa gambling task, as it provides a good balance between exploitation and exploration (Camerer, 2003; Steingroever et al., 2013).

Now that $P(c_t^{self} = 1)$ has been calculated, 0-ToM's decision process is concluded. All that follows is for the probability to be evaluated so 0-ToM can make its choice, after which the next trial commences.

3.2.2 *k*-ToM

The learning process of k-ToM with $k > 0$ differs from that of 0-ToM mainly in that k-ToM must simulate its opponent's learning and decision processes, in order to learn the opponent's choice probability P_t^{op}. To do this, k-ToM assumes its opponent to also be a ToM agent of a lower sophistication level $\kappa < k$. It must then estimate the opponent's parameter values θ, which includes a behavioural temperature β and a volatility σ. This is also done using a variational Bayes Laplace approximation, similar to 0-ToM's learning of RB's probability parameter, but since the parameters θ of a ToM agent has a nonlinear relation to the observed behaviour, linearity is no longer assumed, and the Laplace approximation gets more complex. The nonlinear variational Bayes Laplace approximation estimates parameter values as a normal distribution with a mean μ^θ and variance Σ^θ, for each of the opponent's parameters θ. k-ToM then simulates its opponent's learning and decision process to calculate a mean estimate of the opponent's choice probability. This mean is also denoted μ, but is distinct from μ^θ by not being an estimation of a parameter, but of the opponent's behaviour. Because different behaviour is expected given different opponent levels κ, k-ToM makes parameter estimates μ^θ and choice probability estimates μ for each of the opponent's possible levels κ. But since the opponent's sophistication level κ is not known, k-ToM must first estimate the probability λ for the opponent having each of the possible levels $\kappa < k$.

In the graphical model (figure 2), k-ToM's learning and decision process can be seen. First the probability λ of the opponent having each possible level κ is updated. Then the variances Σ^θ and means μ^θ of the parameter estimates are updated. Using the estimated parameter values, k-ToM performs a recursive simulation of its opponent to calculate a mean choice probability estimate μ, which it compares to choice probability estimates using

incremented parameter estimates in order to calculate the new gradient W. Now the choice probability for each possible opponent level κ, $P^{op,\kappa}$, can be calculated and averaged to a single choice probability estimate P^{op}. This is used to calculate the expected payoff ΔV, which is inserted into the softmax function to calculate k-ToM's own choice probability $P(c^{self} = 1)$.

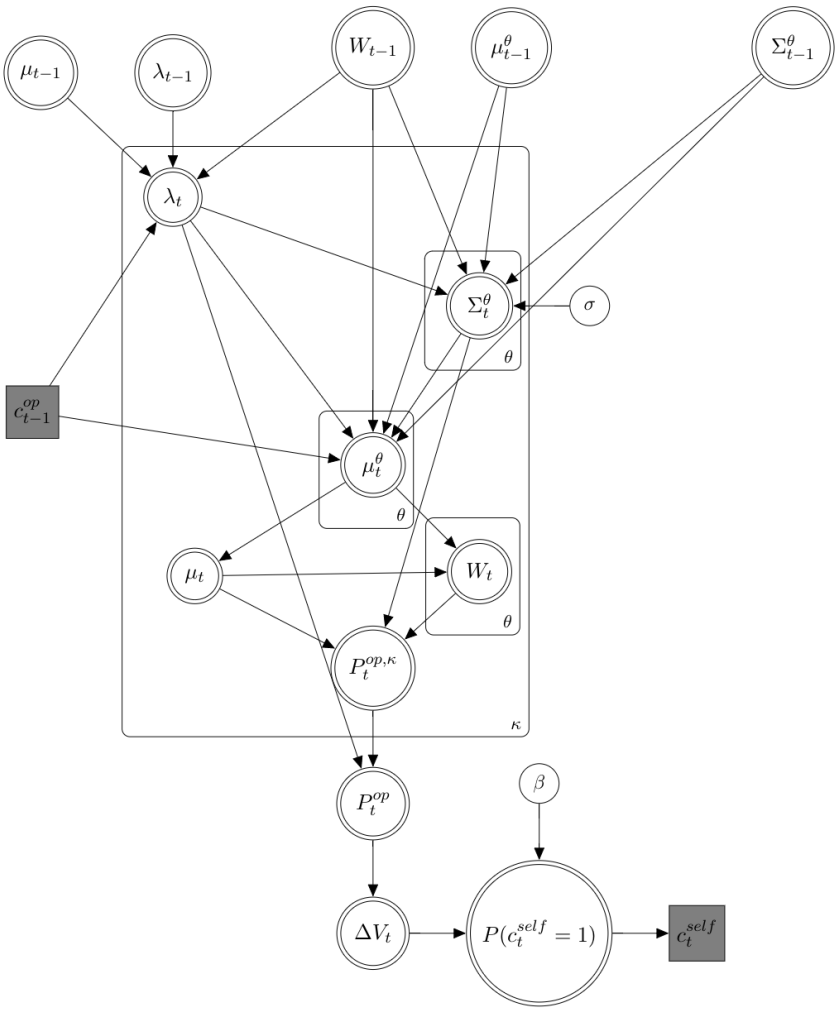

Figure 2: A graphical model of a single trial for the k-ToM model. Shaded variables are observed, squares are discrete while circles are continuous, and double bordered variables are deterministic and unobserved.

For 1-ToM, calculating λ is trivial, as there is only one option, as $P(\kappa = 0) \equiv 1$. But k-ToM agents with $k > 1$ must estimate λ by comparing the expected behaviour of the opponent, given each level, to the actual behaviour. This is done by using the following equation:

$$\lambda_t^{k,\kappa} \approx \left(\frac{\lambda_{t-1}^{k,\kappa} \quad P_{t-1}^{op,\kappa}}{\sum_{\kappa'<k} \lambda_{t-1}^{k,\kappa'} \quad P_{t-1}^{op,\kappa'}} \right)^{c_{t-1}^{op}} \left(\frac{\lambda_{t-1}^{k,\kappa} \quad (1 - P_{t-1}^{op,\kappa})}{\sum_{\kappa'<k} \lambda_{t-1}^{k,\kappa'} \quad (1 - P_{t-1}^{op,\kappa'})} \right)^{1-c_{t-1}^{op}} \tag{9a}$$

Where $\lambda_t^{k,\kappa}$ denotes k-ToM's estimated probability λ at trial t of its opponent having the sophistication level κ, and $p_{t-1}^{op,\kappa}$ denotes the probability of opponent choosing 1, for each simulated level κ, on the previous trial. This equation updates the probability λ for each κ relative to the probability of the actual outcome that was estimated for that level. Thus, levels with incorrect predictions about the opponent's choices become less probable over time. Dividing each calculated probability with the sum over levels κ ensures that the probabilities λ always sums to 1. Note that the exponentiation of the opponent's choice c_{t-1}^{op} decides by which choice (1 or 0) λ should be updated, e.g. if the opponent chose 1, λ is updated according to the first term. Furthermore, note that λ is updated relative to itself, which means that $\lambda_{t-1}^{k,\kappa}$ close to 0 for a given κ will only change slowly, even if they predict correctly.

In the VBA package (Daunizeau et al., 2014), the $P_{t-1}^{op,k}$ used in this equation is not stored from previous trials, but is instead approximated using the fixed form equation, derived by Daunizeau (2017a) as it is computationally more efficient:

$$P_{t-1}^{op,\kappa} \approx s\left(\frac{\mu_{t-1}^{k,\kappa} + b \cdot \left(\Sigma_{t-1}^{k,\kappa}\right)^{c}}{\sqrt{1 + a \cdot \left(\Sigma_{t-1}^{k,\kappa}\right)^{d}}}\right) \tag{9b}$$

Where, $\mu_{t-1}^{k,\kappa}$ is k-ToM's estimated probability of its opponent choosing 1 from the previous trial, for each possible opponent level κ. $\Sigma_{t-1}^{k,\kappa}$ is k-ToM's subjective uncertainty about it's estimate of opponent choice probability μ, also for each possible opponent level κ. The fixed variables have the values a = 0.205, b = -0.319, c = 0.781, d = 0.870. The quality of this approximation have been validated by comparing it to Monte Carlo estimates (Daunizeau, 2017a).

In eq.9b, μ is calculated by simulating the opponent's learning and decision process for each level κ. Σ, on the other hand, is a composite of the uncertainties Σ^{θ} for each parameter θ, which is calculated using the following equation:

$$\Sigma_{t-1}^{k,\kappa} \approx \left(\Sigma_{t-1}^{k,\kappa,\theta}\right)^{T}\left(W_{t-1}^{k,\kappa,\theta}\right)^{2} \tag{9c}$$

Where $\Sigma_{t-1}^{k,\kappa,\theta}$ are k-ToM's uncertainties on the previous trial about each estimated parameter θ, and for each opponent level κ. $W_{t-1}^{k,\kappa,\theta}$ is then the gradient from last trial of the relation between each parameter estimate μ^{θ} and the choice probability estimate μ, again for each level κ. $\Sigma_{t-1}^{k,\kappa,\theta}$ is transposed for each level κ separately before being multiplied with the squared gradient, resulting in a single variance Σ for each level κ, weighted by all the uncertainties related to estimates for that level.

The composite variance Σ is inserted into eq. 9b, the output of which is inserted into eq. 9a to update the probabilities λ for each possible opponent level κ.

After this update, k-ToM now must update it's estimates of its opponent's parameter values θ, which includes both a volatility σ and a behavioral temperature β. This is similar to 0-ToM's update rule, but to account for the non-linear effect of parameter estimates μ^{θ} on the estimated opponent choice probability μ, the updating is weighted by the gradients W^{θ} between μ^{θ} and μ from last trial. First the uncertainty of the parameter estimates is updated with the following equation:

$$\Sigma_t^{k,\kappa,\theta} \approx \frac{1}{\frac{1}{\Sigma_{t-1}^{k,\kappa,\theta}+\sigma^k} + s(\mu_{t-1}^{k,\kappa,\theta})(1-s(\mu_{t-1}^{k,\kappa,\theta}))\lambda_t^{k,\kappa}\left(W_{t-1}^{k,\kappa,\theta}\right)^2} \tag{10}$$

This is similar to equation 4, with the exceptions that now multiple parameters θ is being estimated, and that the estimation happens for each possible level κ. Additionally, the term is weighted by the probability λ of the opponent's level κ and the squared gradient W^2 of each parameter θ, resulting in a smaller updating of uncertainties Σ^θ for uncertain levels κ and for parameters θ with little influence on behaviour. Note that, drawing from the VBA package (Daunizeau et al., 2014), volatility σ is set to 0 when estimating the behavioural temperature β.

After updating the uncertainty of each parameter estimate, k-ToM updates its mean estimate of each parameter θ using the following equation:

$$\mu_t^{k,\kappa,\theta} \approx \mu_{t-1}^{k,\kappa,\theta} + W_{t-1}^{k,\kappa,\theta}\Sigma_t^{k,\kappa,\theta}\lambda_t^{k,\kappa}(c_{t-1}^{op} - s(\mu_{t-1}^{k,\kappa,\theta})) \tag{11}$$

Again, the updating term is similar to eq. 5, but differs in the updating for multiple parameters θ and multiple opponent levels κ, and the weighting by λ and W, resulting in smaller updating of estimates μ^θ for unlikely opponent levels κ and for parameters θ with little effect on behaviour.

Followingly, k-ToM simulate its opponent's learning an decision processes. This is done recursively by repeating the whole learning function once for each possible opponent level $\kappa \in [0, k-1]$. If $k > 1$ this entails simulating the opponents of the simulated opponents, and if $k > 2$, also the simulated opponents of those, etc. For every simulation, the highest level simulated is reduced by one, which eventually makes the recursion end in simulations of 0-ToM agents who do not simulate opponents. The last step of the decision process of each simulated agent is the softmax decision rule (eq. 8), which outputs a choice probability for choosing 1, which is used as the updated mean for k-ToM's mean estimate of the opponents choice probability μ. This can be written as the following equation:

$$\mu_t^{k,\kappa} = l \circ v(\mu_t^{k,\kappa,\theta}) \tag{12}$$

Where l is the logit function and v is the mapping of the relation between θ values and observable behaviour, in this case the recursive simulation of k-ToM's possible opponents. v then reverses the perspective by applying the opponent's payoff matrix and reversing c^{self} and c^{op}. Note also that v also uses the simulated opponents' prior beliefs, which are stored from last trial.

The last part of k-ToM's learning function is to calculate the gradient W between parameter estimates μ^θ and choice probability estimates μ. The gradient is calculated by numerically approximating the following differentiation:

$$W_t^{k,\kappa,\theta} = \frac{d\mu_t^{k,\kappa}}{d\mu_t^{k,\kappa,\theta}} \tag{13}$$

Importantly, the differentiation is calculated for parameters one at a time while all other parameters are held constant, yielding one gradient W for each parameter θ. The numerical approximation is done by slightly

incrementing parameter estimates μ^{θ} one at a time, and then simulating the learning and decision process of the opponent again by insertion into eq. 12 with the incremented parameter value. The difference in choice probability estimates μ is then divided by the increment size i to estimate a slope of the relation between parameter value estimates μ^{θ} and choice probability estimates μ. The increment size is set to be $0.001 \cdot \mu^{\theta}$, and as minimum 0.001. The approximated differentiation is a local linearization of the nonlinear relation between parameter estimates μ^{θ} and choice probability estimate μ, which serves to estimate how important each the estimation of each parameter θ is to the overall choice probability estimate μ. The use of the gradient W in equations 9c, 10 and 11 makes parameters be updated and uncertainties about them be weighted relative to how much they affect the predictions, so that parameters are updated appropriately in relation to each other, and updated in the appropriate direction. This allows for estimating the opponent's parameter values based on their dynamic and nontrivial effect on his observed behaviour. This concludes k-ToM's learning process.

k-ToM's decision process is largely similar to that 0-ToM. First k-ToM uses its updated mean choice probability estimates $\mu^{k,\kappa}$ to calculate the choice probability estimate $P_t^{op,\kappa}$ where the effect of variance has been included. This is done using an equation similar to eq. 6b:

$$p_t^{op,\kappa} \approx s\left(\frac{\mu_t^{k,\kappa}}{\sqrt{1 + a \cdot \Sigma_t^{k,\kappa}}}\right) \tag{14a}$$

Where a=0.36, $\mu_t^{k,\kappa}$ is the mean of the opponent choice probability estimation on trial t, and $\Sigma_t^{k,\kappa}$ is a composite of the variances Σ^{θ} of the parameter estimations. The composite variance Σ is calculated using the following equation:

$$\Sigma_t^{k,\kappa} = \sum_{\theta} \Sigma_t^{k,\kappa,\theta} \left(W_t^{k,\kappa,\theta}\right)^2 \tag{14b}$$

Where $\Sigma_t^{k,\kappa,\theta}$ is k-ToM's subjective uncertainty at trial t of the estimations of each parameter θ, for each possible opponent level κ, and $W_t^{k,\kappa,\theta}$ is the gradient on trial t of the relation between parameter estimations μ^{θ} and estimated choice probabilities μ. The calculation is done separately for each possible opponent level κ, resulting in a composite variance Σ for each level κ.

Once the opponent choice probability estimates $P_t^{op,\kappa}$ are calculated, k-ToM calculates a single estimate of the opponent's choice probability P_t^{op}. This is done by taking a mean of the choice probability estimates $P_t^{op,\kappa}$, weighted by the probability $\lambda_t^{k,\kappa}$ of the opponent having each level κ. This can be seen in the following equation:

$$P_t^{op} = \sum_{\kappa} \lambda_t^{k,\kappa} P_t^{op,\kappa} \tag{15}$$

Note that since $\lambda_t^{k,\kappa}$ sums to one, this is a weighted mean. Now that the final composite estimate of the opponents choice probability is calculated, all that remains is to insert it the expected payoff function (eq. 7) and then the softmax rule (eq. 8) for k-ToM to calculate its own choice probability $P(c_t^{self} = 1)$. Lastly, the probability is evaluated for k-ToM to make its choice, after which the next trial can commence.

Using defaults drawn from the VBA package (Daunizeau et al., 2014), the SiRToM package uses default values for σ and β of -2 and -1, respectively, sampled from a normal distribution with standard deviation 0.1. Parameter values are exponentiated before insertion into the equations. The size of σ creates an effective lower bound on variance values at about 0.7, and the size of β results in the softmax returning a probability of approximately 94 % given an expected payoff difference of 1.

Similarly, the simulated ToM agents in the SirToM package use agnostic priors about their opponent's level probabilities λ and choice probabilities μ, while parameter estimation means μ^θ are set to 1. All variances Σ^θ and Σ for parameter and choice probability estimation, respectively, are also set to 1. Gradients for all parameters are 0 on the first trial, which means that no parameter estimates and variances are updated during the first trial. The SiRToM package uses priors and parameter values similar to the ones used in the VBA toolbox (Daunizeau et al., 2014) to enable comparison, but other settings are possible, and are also explored further in *4 Simulation Results*.

4 Simulation Results

KCE & PTW

In the following is shown the visualized results of three case simulations, acting as an example of how the SiRToM package could be used. They all include a Random Bias agent (RB), a Win-stay Loose-shift agent (WSLS), and Theory of Mind agents with sophistication levels $k \in [0,5]$, all playing the matching pennies game in a round robin tournament. Agents' parameter values were re-sampled from their distribution at the beginning of each simulation. The first case consists of 100 simulations of 200 trials per game under default conditions, hereby investigating how ToM's performance and parameter estimates changes over time. The second uses 200 simulations of only 30 trials, testing ToM performance in a setting with limited time. And in the third, *k*-ToM agents use optimally accurate priors instead of the arbitrarily chosen default priors, which was done using 40 simulations.

4.1 Case 1: Default settings

In the first case, all agents except RB used SiRToM's default parameter values. The probability parameter of the RB strategy value was specified to be sampled around 0.8, so that 0-ToM's estimation of it could be evaluated better.

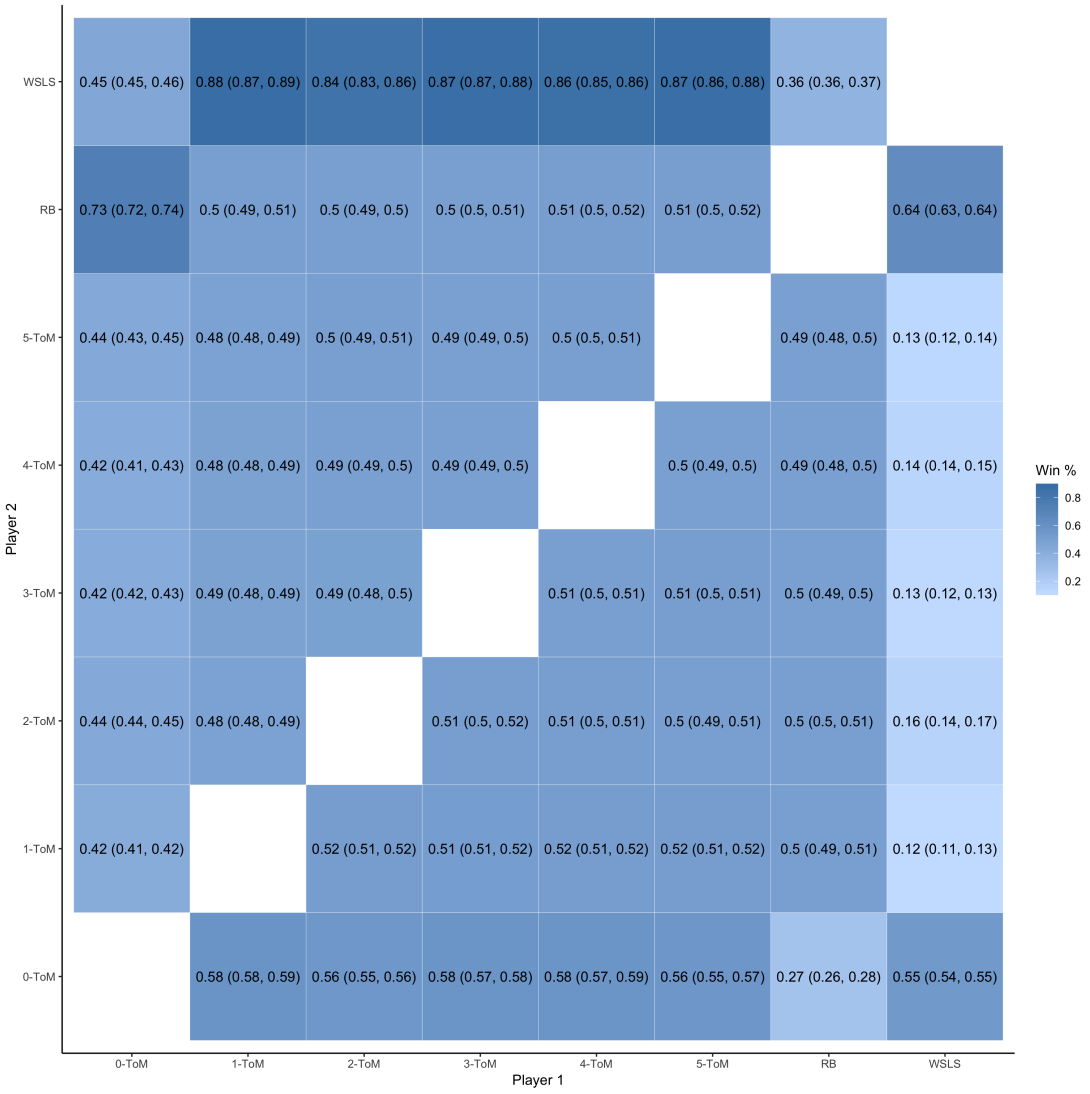

Figure 3: A heatmap showing the average win rate of agents, over 200 trials and 100 simulations. The values in the cells indicate the win rate and the 95% confidence interval.

Examining figure 3 it is seen that, as intended, the 0-ToM agent outperforms the Random bias (RB) agents with a win rate of 0.73 (95% CI, 0.72, 0.74). Similarly to the 0-ToM it is also seen that the simple WSLS heuristic also manages to capitalize on the RB agent's bias, albeit to a lower extent with a win rate of 0.60 (95% CI, 0.58, 0.62). As expected it is seen that the k-ToM agents with a sophistication level $k > 0$ all outperform the 0-ToM with a win rate of 0.56-0.58 and a lower 95% confidence interval (CI) above 0.55. Notably the 1-ToM outperforms the higher level k-ToM agents against 0-ToM, as it need not estimate its opponent's sophistication level. In figure 3 and it is also seen that 1-ToM and 2-ToM agents outperform a k-ToM with a lower sophistication level, while k-ToM agents with $k > 2$ see no apparent increment in performance is seen when using the default priors. Strikingly k-ToM agent with $k > 0$ perform at chance level when competing against a RB with a clear bias.

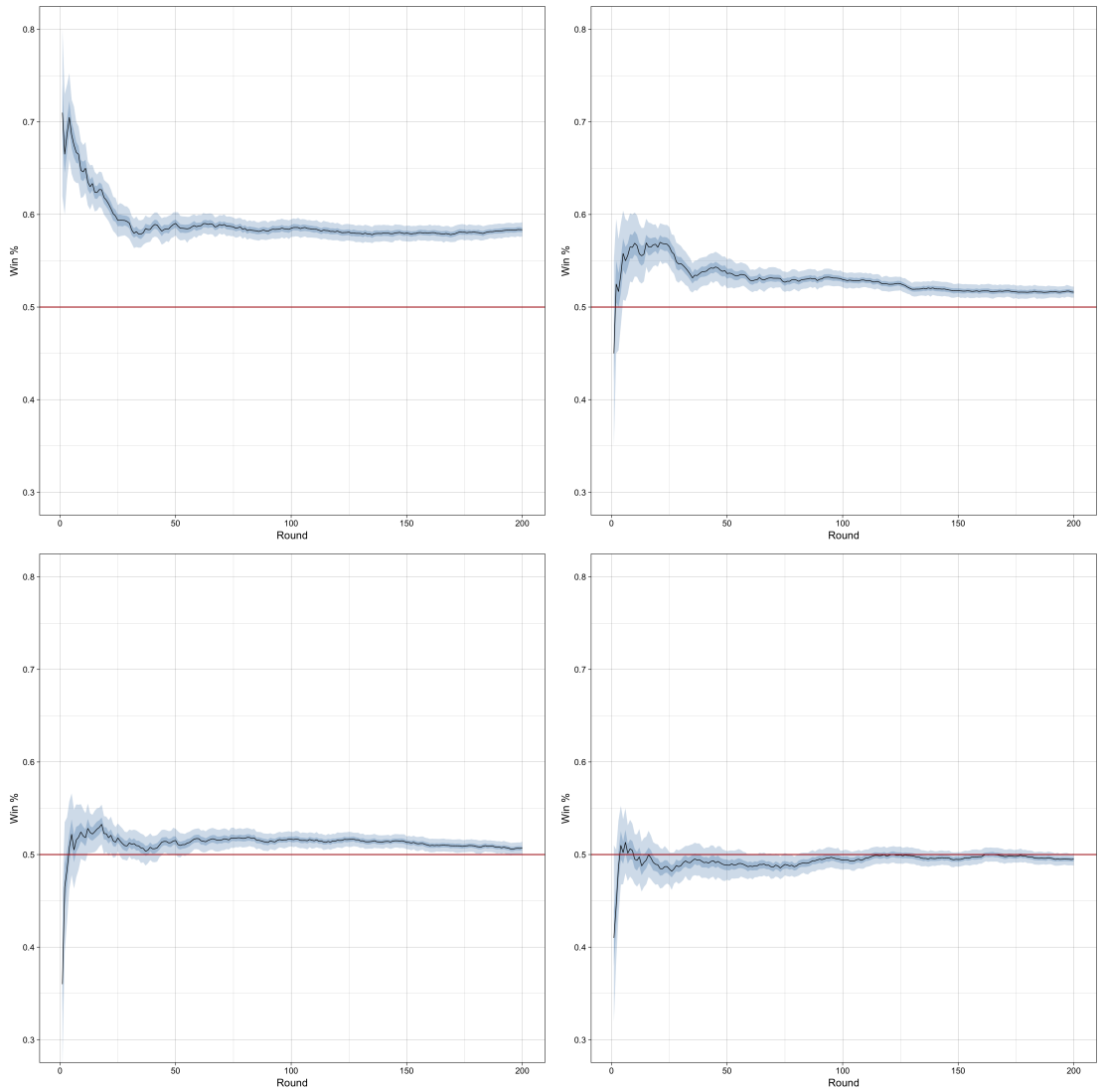

Figure 4: The win rate of agents over 200 trials and 100 simulations. From left to right, top to bottom, 1) a 1-ToM agent versus a 0-ToM, 2) a 2-ToM versus a 1-ToM, 3) a 4-ToM versus a 2-ToM, and 4) a 5-ToM versus a 4-ToM. The light and dark blue intervals indicate 95% and 50% non-parametric bootstrapped confidence intervals (CI).

In figure 4.1 which shows a 1-ToM agent's win rate against a 0-ToM , it is seen that a 1-ToM win rate starts out well above 60% but settles slightly below a 60% win rate. Similarly we see that (figure 4.2) that a 2-ToM obtain a higher win rate in the initial 25 trials. In figure 4.3, in which a it is seen that a 4-ToM performs only slightly above chance level against a 2-ToM over 30 trials and around chance level after 200 trials. When a 5-ToM is competing against a 4-ToM (figure 4.4) we see that it looses on average, although indistinguishable from chance.

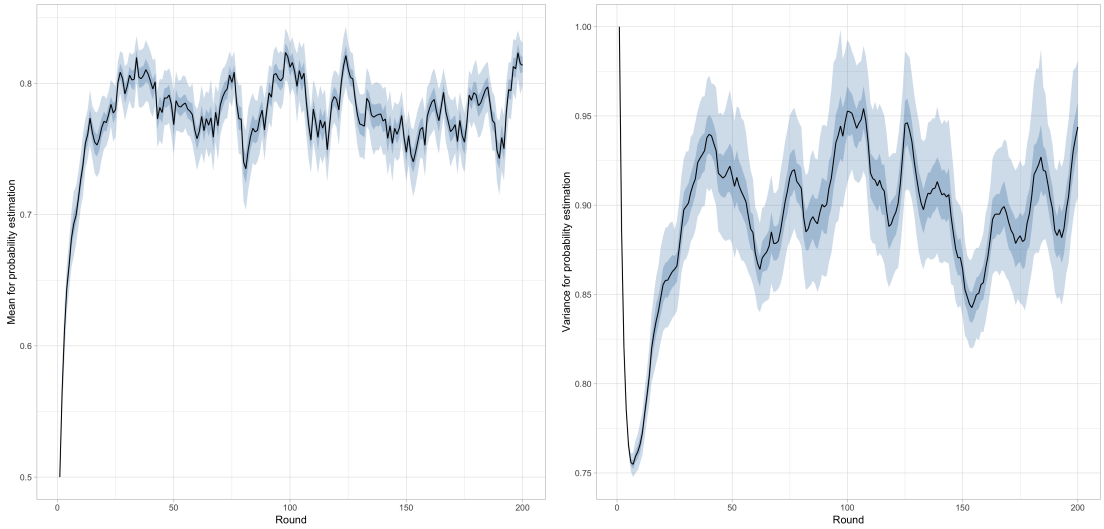

Figure 5: 0-ToM's estimate of RB's probability parameter over 200 trials and 100 simulations. The actual probability estimates were around 0.8. From left to right, the mean estimate and the uncertainty. The light and dark blue intervals indicate 95% and 50% non-parametric bootstrapped confidence intervals (CI).

Examining figure 5 of a 0-ToM agent's parameter estimate of a RB agents, we see that 0-ToM correctly approximates the probability parameter of approximately 80% after 25 trials, which is also reflected in the variance estimate. However, both estimates, retains a measure of uncertainty.

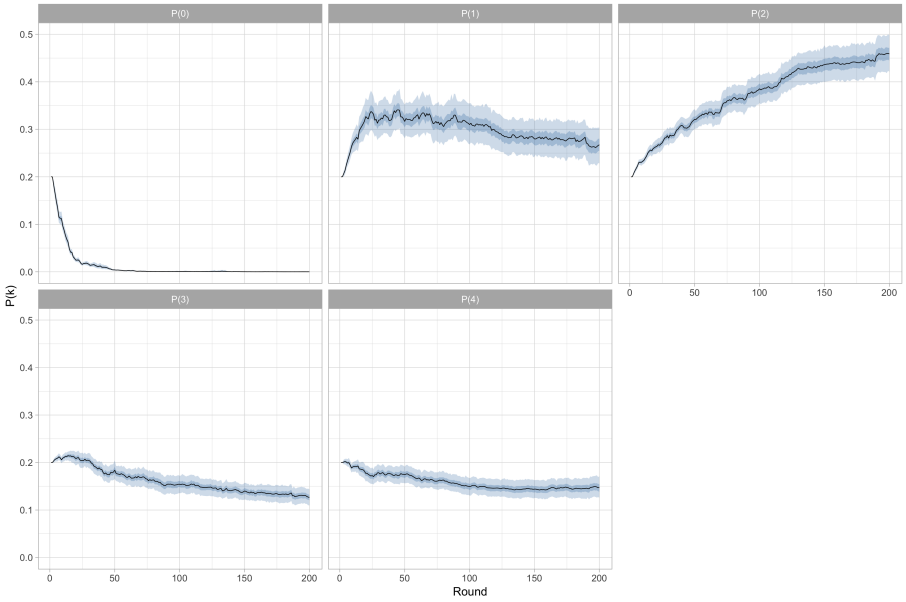

Figure 6: A 5-ToM's estimated probability λ or P(k) of its opponent's sophistication level κ when playing against a 2-ToM agent, over 200 trials in 100 simulations. The light and dark blue intervals indicate 95% and 50% non-parametric bootstrapped confidence intervals (CI).

In figure 6 we see a 5-ToM agent's probability estimate of its opponents sophistication level when playing against a 2-ToM. While a 5-ToM steadily become more certain of its opponent being a 2-ToM it never exceeds a 50%. Notably it quickly realizes that a 0-ToM agents is highly unlikely.

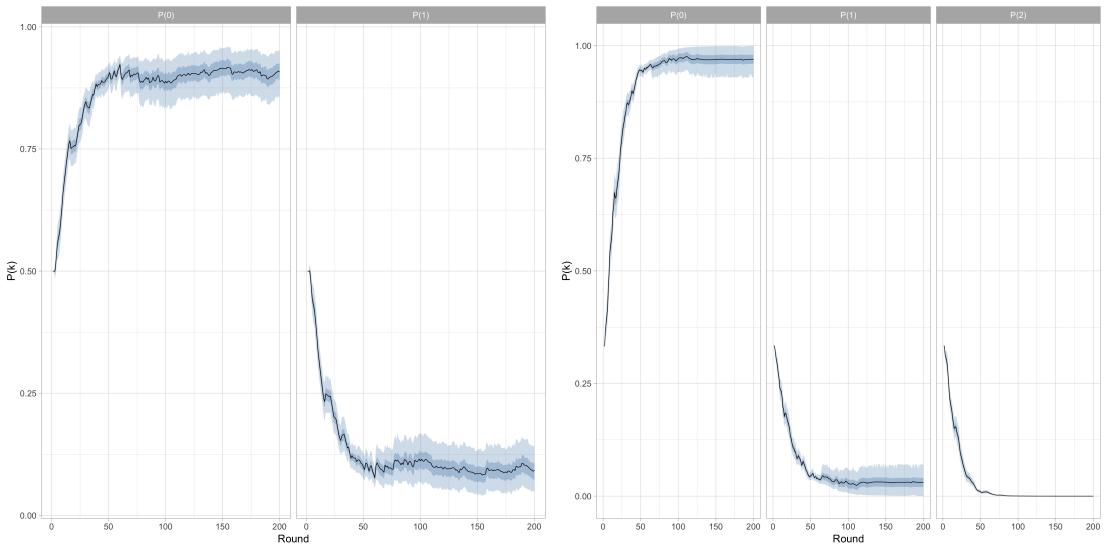

Figure 7: Left; 2-ToM's estimated probability λ or P(k) of its opponent's sophistication level κ when playing against a 0-ToM. Right; 3-ToM's estimated probability λ or P(k) of its opponent's sophistication level κ when playing against a 0-ToM. Both over 200 trials, averaged across 100 simulations. The light and dark blue intervals indicate 95% and 50% non-parametric bootstrapped confidence intervals (CI).

Figure 7 interestingly illuminate why k-ToM with $k > 0$ have similar score when playing against a 0-ToM, as they are approximately able distinguish a 0-ToM from other k-ToM opponents around the same trial. It also illuminate why they don't receive a score similar to the 1-ToM, not only do they have to estimate its opponent's sophistication level but it also never obtains certainty in its estimates, which prevents if from exploiting the 0-ToM.

4.2 Case 2: 30 trials

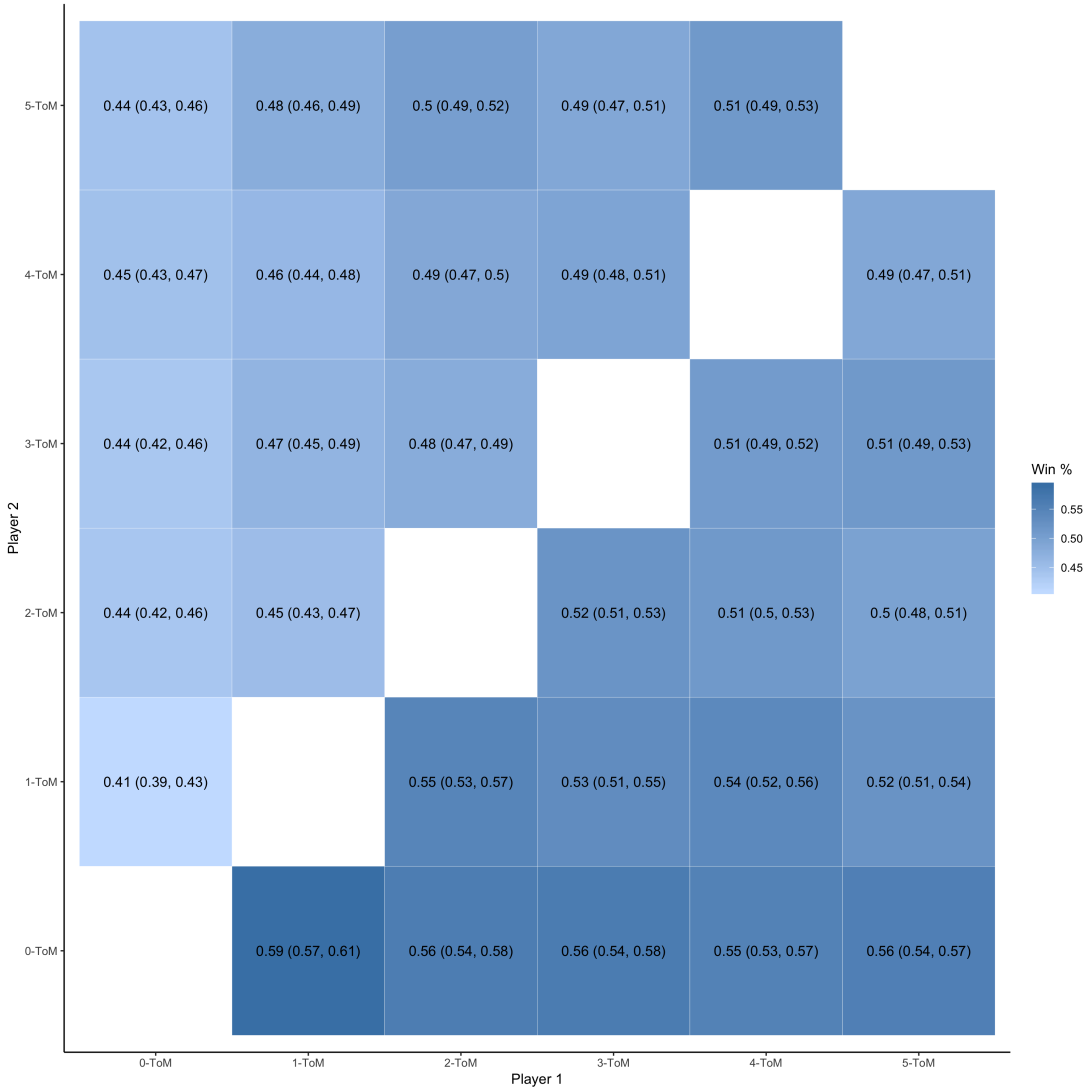

Figure 8: Heatmap showing the average win rate of agents in the first 30 rounds, across 100 simulations. RB and WSLS agents have been excluded for overview. The values in the cells indicate the win rate and the 95% confidence interval.

In figure 8, it can be seen that performance patterns are generally similar to case 1. Two differences are, however, relevant, compared to case 1, 1-ToM here has a comparatively larger advantage over the higher levels in exploiting 0-ToM, and the win rate is generally higher among k-ToMs with $k > 1$ when playing against less sophisticated k-ToM agents.

4.3 Case 3: Accurate priors

SiRToM's default priors, which were used in cases 1 and 2, were taken from the VBA package (Daunizeau et al., 2014). In order to examine the effect of *k*-ToM's priors, we ran another simulation where ToM agents had optimally accurate priors for parameter estimates μ^{θ} (σ = -2 and β = -1).

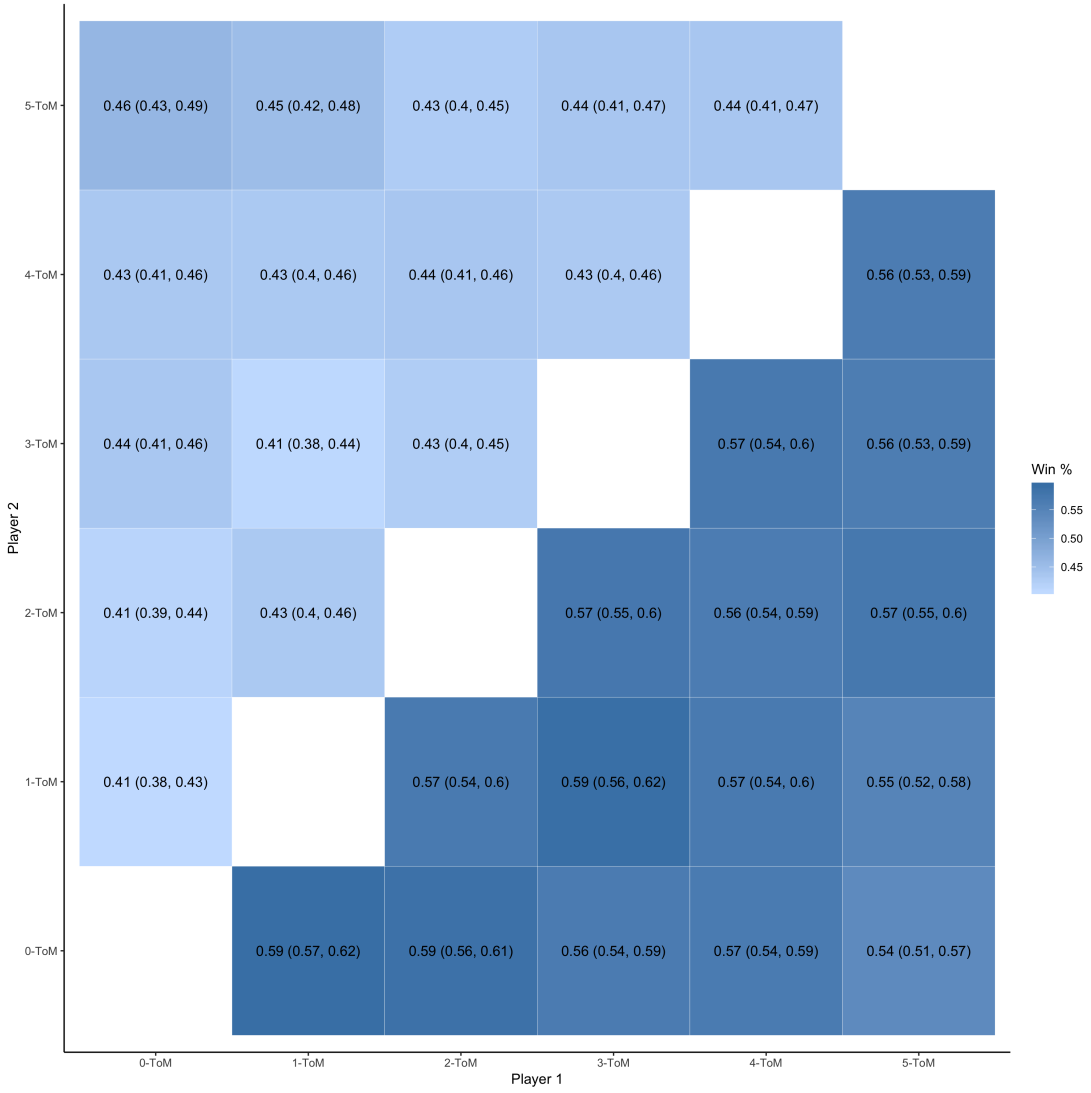

Figure 9: A heatmap showing the average win rate of ToM agents after 30 rounds using accurate priors. The values in the cells indicate the average win rate and the 95% confidence intervals over 40 simulations.

On figure 9 it can be seen that the performance of *k*-ToM has greatly improved, compared to cases 1 and 2, to a degree where even 5-ToM performs consistently well against all opponents. Seemingly a *k*-ToM of a higher level consistently beat a *k*-ToM with lower sophistication level in this setting, however the advantage of sophistication still seem to decrease slowly with higher increased complexity.

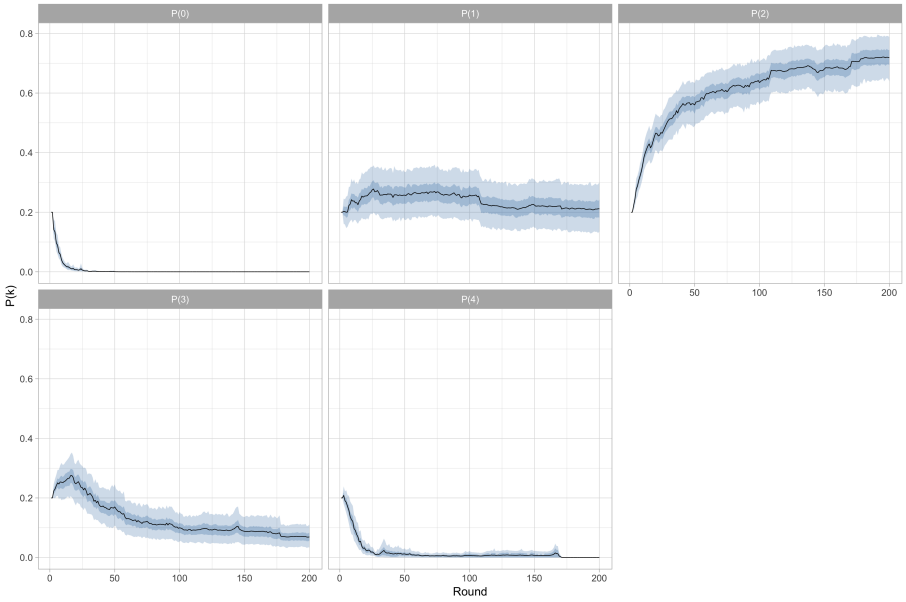

Figure 10: A 5-ToM's estimated probability λ or P(k) of its opponent's sophistication level κ when playing against a 2-ToM agent using accurate priors, over 200 trials in 40 simulations. The light and dark blue intervals indicate 95% and 50% non-parametric bootstrapped confidence intervals (CI).

On figure 10 it can interestingly be seen that accurate priors result in better and faster estimation of the opponents' sophistication level, compared to figure 6 in case 1.

5 Discussion

5.1 Model Performance

KCE & PTW

Overall, the simulation gives results similar to earlier work (Devaine et al., 2014b), validating its implementation. In general, sophisticated agents outperform simple agents, with 0-ToM correctly estimating and outperforming RB, but being outperformed by higher levels of *k*-ToM agents. Note that 0-ToM can better exploit RB agents if it has a lower behavioural temperature (see *Appendix*), but that this also makes it easier to predict and exploit by more sophisticated ToM agents. The WSLS approximates a maximum-learner 0-ToM model, making its performance similar to 0-ToM's, although its higher predictability makes it perform worse against *k*-ToM agents. Also similarly to Devaine et al. (2014b), an increase in sophistication level quickly grows less beneficial with each level, and also makes it harder to exploit 0-ToM agents efficiently. This informational cost to sophistication results in the best overall performance being by 1- and 2-ToM agents. This is congruent with findings that human ToM sophistication is usually around 2 in human populations (Devaine et al., 2014a, 2017). A curious finding is that agents with $k > 0$ cannot exploit RB agents because they do not model a consistent bias in the opponent's behaviour (see *5.2 Potential improvements on the k-ToM model* for a suggested solution).

In figure 4.1 it is notable that k-ToM performs worse over time. This happens as 0-ToM becomes more uncertain about its estimates about the opponent, since 1-ToM does the opposite of what 0-ToM expects. Consequently, 0-ToM's variance increases, leading its own choice probability to become more random, which makes it harder to exploit by the 1-ToM. The keen reader will similarly have noted that k-ToM agents with $k > 1$ perform close to chance level when competing against k-ToM with $k > 0$ for 200 rounds (see figure 3 and 4.2). Similarly to when 1-ToM competes against 0-ToM, this is caused by the increased uncertainty of the k-ToM with lowest sophistication level. This is a rather interesting feature, as the randomization of the simpler *k*-ToM is the Nash equilibrium in the matching pennies game, hereby acting as a defense mechanism against further losses. This is also why *k*-ToM agents perform better with fewer trials against simpler ToM agents. It could be interesting to examine whether humans playing against a superior opponent acts in a similar fashion.

Parameter estimation patters do not change much with priors (see *Appendix*), but that estimates are correct early on has a lasting effect on every aspect of *k*-ToM's performance (see figures 9 and 10). Accurate priors seem to enable even highly sophisticated ToM agents to perform well against other agents, even over many trials. This indicates that Devaine et al.'s (2014b) suggested upper bound on sophistication levels at $k = 2$ might depend on the quality of priors. While perfect prior beliefs are very unlikely in any natural setting, the simulation results indicate that it is necessary to examine how *k*-ToM ought to form its priors, and which priors are used by humans. The results from (Devaine et al., 2014a) could suggest that this is very context-dependent.

5.2 Potential improvements on the *k*-ToM model

PTW, KCE

The *k*-ToM model's equations are generalizable to any number of model parameters, and the implementation in SiRToM is made so parameters and changes to the model can be added as optional features. Some possible features are easier to implement than others. What follows is a discussion of how the *k*-ToM model could be expanded to perform better and in more different contexts.

Inspired by the VBA package (Daunizeau et al., 2014), adding a perturbation to learning is an easily implementable and potentially rewarding extension of the model. This would entail simply adding a number sampled from a normal distribution with a mean of 0 to all values after updating them. The standard deviation of the normal distribution then signifies the size of the perturbation of the learning, and would be decided by a perturbation parameter for *k*-ToM. A perturbation to the data like this could potentially help prevent overfitting.

Similarly inspired by the VBA package (Daunizeau et al., 2014), a bias parameter can be added to the *k*-ToM model, which simply adds a constant to the expected payoff in the softmax function after it is divided by the behavioural temperature (see eq. 8). This would allow for creating a *k*-ToM agent with a constant bias towards either option in the game, but more importantly also allow *k*-ToM to estimate a bias for the opponent. Critically, this allows ToM agents with $k > 0$ to exploit the bias in RB agents, with which they otherwise have problem (see 3). This also allows the model, when fitted to the behaviour of humans, to estimate bias in humans, where it might be more prevalent than in artificial agents (see *5.4 Future Applications*).

A last possible extension for the *k*-ToM model that is also taken from the VBA package (Daunizeau et al., 2014)

is letting k-ToM do a partial forgetting of the level probabilities λ, i.e. moving them closer to agnostic chance levels at every trial. The size of this forgetting is controlled by a dilution parameter between 0 and 1. The partial forgetting can help prevent overfitting, but it more importantly also allows k-ToM to adapt to opponents who can switch between strategies. Without the forgetting, as it is currently, once a level is very improbable, it takes many rounds and much evidence for it to become probable again (see eq. 9a and figure 6).

There is reasonable evidence to suggest that humans base prior beliefs on earlier encounters in new situations, such as when playing the ultimatum game or when competing in the prisoner's dilemma, in which humans have been shown to be predisposed to exhibit cooperative behaviour (Fehr and Fischbacher, 2004). We have also shown in this thesis that the accuracy of the parameter estimate priors has great effect on many aspects of k-ToM's performance (see *5.1 Model Performance*). Therefore, it seems reasonable that a k-ToM model, in order to explain human behaviour, should also have a way of forming prior beliefs across opponents between encounters. One option is to empirically test the distributions of parameter values found in the human population, and use those as priors for k-ToM (for an example, see Skewes et al. 2017). Some game types might also call for specific priors. As it is now, the k-ToM has difficulties playing the stag hunt game (see *Appendix*) because it is likely to defect and choose the safer option. Cooperative priors, i.e. a prior belief that the opponent will cooperate, is already well documented in humans (Fehr and Fischbacher, 2004) and implementing these will allow the ToM agents to initiate cooperation in the stag hunt. A more dynamic approach to prior beliefs might also be appropriate in environments with repeated games and multiple opponents. Other non-variational Bayes implementations of Theory of Mind models have utilized having a group-wide representation of intentions (Khalvati et al., 2018), or building their initial beliefs about opponents based on earlier encountered opponents (Rabinowitz et al., 2018). In an evolutionary setting, prior estimates of new ToM agents could be sampled from a distribution with the mean sampled from the estimates made by successful agents, which should allow the prior estimate to drift from the current arbitrary default priors. This would naturally require an implementation of a evolutionary environment structure (see *5.3 Package development*).

k-ToM could, furthermore, be expanded to be able to explicitly consider the effects of its own choices on the opponent's future choices. As it is now, k-ToM only learns and decides reactively. Sometimes this can still lead to relatively sophisticated strategies, like agents of the same level imitating two different, lower levels to be able to cooperate efficiently (Devaine et al., 2014b). But it is a problem in games like the Prisoner's Dilemma, where the dominant strategy is not the one that leads to the highest gain. In the Prisoner's Dilemma, k-ToM cannot see past the current round, and will always defect, no matter its predictions about the opponent. One solution is to let k-ToM simulate a number of future trials on behalf of itself and the opponent (what is called considering the "Long Shadow of the Future" (Axelrod and Hamilton, 1981)). k-ToM would have to simulate all possible situations, and make the choice that leads to the highest average reward, weighted by the estimated probabilities of each situation occurring. k-ToM agents would then differ on another dimension, the number of trials they are able to simulate ahead. It would be possible that a temporal discounting effect (Green et al., 1997) would emerge, as a consequence of uncertainties and probabilities accumulating with simulated trials, leading to trials in further in the future having very uncertain predictions and little effect on k-ToM's choice. An alternative to simply choosing the option with the highest probability-weighted average rewards in the future would be to allow k-ToM to work more specifically towards beneficial scenarios. This would be akin to the idea of active inference in the Free Energy Principle,

where an organism actively affects the environment to produce predicted behaviour. In some cases like the penny matching game, this might not be different, but especially in fragile cooperative settings like the Prisoner's Dilemma it might allow *k*-ToM to create a scenario where cooperation becomes possible. Another facet of performing active inference would also be for *k*-ToM to actively seek out situations where it will be able to get data that allows for better estimations of the opponent, in the long run also improving performance. It is noteworthy that more specific goals also might make *k*-ToM easier to exploit, especially by other ToM agents of higher sophistication levels, because it is willing to sacrifice more in order to get into a potentially beneficial scenario.

Another potentially beneficial expansion of the *k*-ToM model would be to allow it to simulate other types of opponent strategies than ToM strategies of lower levels. It is implementable for such a "MetaToM" agent to be able to simulate Tit-for-Tat, Win-stay Loose-shift, Reinforcement Learning or other strategies, and the variational Bayes Laplace approximation should be able to estimate their parameters as well. It would then be simple to let the MetaToM estimate the probabilities for the opponent having each of its known strategies, in exactly the same way that the *k*-ToM estimates the probabilities of each possible opponent level κ.

Once *k*-ToM or MetaToM has access to other strategies, it would also be possible to allow it to change strategies itself, given specific circumstances. How this could be beneficial depends greatly on the context, but one example where switching strategy could be useful, would be switching to the Nash equilibrium Random Bias strategy with probability parameter 0.5 in the matching pennies game, to reduce losses when playing against a superior opponent. This would require ToM to have a representation of its chances of getting a score higher than with a RB strategy, which could be implemented in multiple ways.

It would also be possible to expand *k*-ToM to function in environments that are partially unknown. *k*-ToM could for example possibly be made to estimate the opponent's payoff matrix instead of knowing it before the game. This would perhaps resemble some social situations, where the opponent's preferences and goals are not known *a priori*. Also resembling actual social situations might be allowing the ToM model to have a sense of jealousy or altruism, that is, have a preference for the opponent to get lower or higher rewards, respectively, independently of ToM's own scores. This can often be represented in the payoff matrix rather than in the strategy, but in some contexts like multiplayer games, ToM might be incentivized to prevent certain opponents from gaining points rather than optimizing purely for its own gains. Similarily, ToM might prefer to cooperate only with those opponents not likely to end up with a higher score than itself. Being a dynamic and agent-specific preference for the opponent's score, it cannot be represented in the payoff matrix only.

These are many possible expansions on the *k*-ToM model, and while they all have possible benefits, it should be mentioned that a too-complex model might not be able to make certain or useful inferences, or at least would depend greatly on good priors. It might be preferable to use only a subset of the extensions, depending on context. It can also be a subject to future research which extensions are useful in which contexts.

Note that the implementation of ToM estimate its opponent parameters using a Gaussian distribution as it is computationally simple (Yoshida et al., 2008). This is reasonable when competing against other ToM agents and the current Random bias (RB) implementation, but it might not be a reasonable in all scenarios. However, given no other information, the Gaussian distribution is reasonable because it has maximum entropy (McElreath, 2016).

5.3 Package development

KCE, PTW

Apart from implementing extensions on the k-ToM model, future iterations of the SiRToM package should include a variety of other agent strategies, like reinforcement learning agents, Bayesian ToM agents similar to those used by Baker et al. (2017), or ToM agents employing neural networks similar to those developed by Rabinowitz et al. (2018). The influence learning model and the volatile Random Bias agents used by Devaine et al. (2017) are also candidates. As it is now, SiRToM has a very modest range of environmental structures available, but as the environment plays an important role, future iterations should include more variety. A very relevant implementation would be evolutionary ABM structures used to study the emergence of evolutionary dominant strategies, which have been widely utilized in behavioural and biological studies (Bowles and Gintis, 2011), and which has already been used by Devaine et al. (2014b) to study ToM. Another would be manipulating the environment to see if interaction structures has an effect on ToM's performance. A wide variety of options for implementing environment structures could be gained by incorporating the Netlogo extension for R, RNetLogo Thiele (2014), the R package called network developed by (Butts, 2008) to manage relational data, or the similar igraph package by Csardi and Nepusz (2006).

SiRToM currently contains a feature under development allowing humans to play against implemented agents. This feature allows for utilizing the ToM implementation in experimental settings, similarly to Devaine et al. (2014a). The current implementation simply uses a command line interface, as currently no package exists for running experiments in R. But future package iterations could integrate with a language like Python in PsychoPy better suited for running experiments (Peirce, 2007), which can be done by using the reticulate package for running Python in R (RStudio Team, 2018). Alternatively, ShinyR (RStudio, 2018), a framework for creating interactive web interfaces, could be used for collecting data in an online setting.

Also, for the development of SiRToM to have an effect, the package has to be advertised and easy to use. There are multiple ways of doing this. The SiRToM package can be made available on the Comprehensive R Archive Network (CRAN), R's publicly available collection of packages (R Core Team, 2013), which would make it easily accessible and increase trust in it (Leek, 2015). The writing of extensive and clear documentation for the package's functionalities is also important. Another way is encouraging the use of the package in different contexts. This has already been done preliminarily, by providing the package to a student research project on the relation between empathy and performance against the artificial ToM agents (Mortensen and Nymann, 2018). Making SiRToM broadly available and encouraging its use is fundamental to its societal value in itself, but the fact that many people use it and interacts with its code can also be beneficial for the development of the package itself (Ebert, 2009).

5.4 Future Applications

KCE, PTW

Computational implementations of ToM have already proved relevant in describing ToM processes in humans and animals (Devaine et al., 2014b,a, 2017) and have shown promising performance on advanced cooperative games

(Rabinowitz et al., 2018; Foerster et al., 2018). ToM have even been suggested for applications in human-computer-interfaces and advances in interpretable A.I (Rabinowitz et al., 2018). The SiRToM package provides am accessible framework for testing a waide array of ToM-related hypotheses.

The SiRToM package has already been used in a unpublished project by Mortensen and Nymann (2018) to investigate the relation between empathy and performance when playing against a k-ToM agent. Mortensen and Nymann (2018)'s project utilized SiRToM's feature which allows humans to play against implemented agents. While no effect was found, likely due to small sample size and fewer than 20 trials, it is a proof of concept of SiRToM to be useful in allowing even relatively untrained students to test hypotheses using an advanced ToM model. Developing a companion package for fitting the ToM models to human data, perhaps using the RStan interface between R and STAN (Team, 2016), would also allow for accessible but advanced analysis of data generated by the SiRToM package.

The model of Theory of Mind currently implemented in SiRToM does not adhere specifically to either of the conceptualizations of Theory of Mind discussed by Goldman et al. (2012). It uses simulation to predict the opponent, but also assumes the opponent to be rational, and learns the opponent's parameter values statistically. As such, the model unifies the perspectives of that debate, but by expanding it, it could be used to differentiate the views as well. Similarly, it could be used to differentiate cognitive and affective ToM, but perhaps also be a groundwork for modelling affective ToM computationally.

Given the rise of online social media such as twitter and facebook, agent-based models using a network structure is useful for modelling these environments Gilbert and Hamill (2009). ToM is an especially interesting agent in a social setting as it has been shown to exhibit behaviour similar to humans who believe they interact with other humans (Devaine et al., 2014a). Being highly flexible, the use of network structures also allows for modelling of situations such as developing social circles, or creating classical grid-like structures like those used by Epstein (1998) in his outstanding study showing that even zero-memory cooperation strategies thrive in a demographic prisoner's dilemma. This could also be used to further investigate Devaine et al.'s (2014a) findings that a mixture of 1- and 2-ToM is evolutionary stable in a cooperative setting. A possible hypothesis is that, in a grid-ike structure, we would see 2-ToM agents surrounded by 1-tom and vice-versa. Such a study might prove relevant for the current literature of organizations of groups. It can also be investigated in general which conditions affect the upper bound on feasible sophistication levels, and if there are any conditions in which the upper bound disappears entirely.

6 Conclusion

KCE & PTW

The SiRToM package for R is a tool, currently under development, for agent-based models in game theory with a strong emphasis on a variational Bayes implementation of a computational model of Theory of Mind. The implemented model has already been used in earlier research on Theory of Mind, but has not been publicly accessible in an open source software. Initial results using the SiRToM package indicate priors to be highly influential for the inference and performance of the Theory of Mind model, and warrant further research on how the models priors should be formed. The package is intended for further research on Theory of Mind, either using agent-based models or in human behavioural experiments. It has, indeed, already been used for a preliminary student project. For it to be

influential, the package should be made broadly and easily available, so as to scaffold research using computational models of Theory of Mind. Future package development should include further integration with other established packages in R to leverage well developed features like advanced environment structures, or an interface integration for experimental use. The Theory of Mind model can be expanded upon in various ways, and a companion package could enable fitting the model to human behavioural data. All in all, there are many future potential uses for the SiRToM package, which warrants developing it further.

References

Axelrod, R. and Hamilton, W. D. (1981). The evolution of cooperation. *Science (New York, N.Y.)*, 211(4489):1390–6.

Axelrod, R. M. (2006). *The evolution of cooperation*. Basic Books.

Axtell, R. L., Epstein, J. M., Dean, J. S., Gumerman, G. J., Swedlund, A. C., Harburger, J., Chakravarty, S., Hammond, R., Parker, J., and Parker, M. (2002). Population growth and collapse in a multiagent model of the Kayenta Anasazi in Long House Valley. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7275–7279.

Baker, C. L., Jara-Ettinger, J., Saxe, R., and Tenenbaum, J. B. (2017). Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, 1(4):64.

Bartlema, A., Lee, M., Wetzels, R., and Vanpaemel, W. (2014). A Bayesian hierarchical mixture approach to individual differences: Case studies in selective attention and representation in category learning . *Journal of Mathematical Psychology*, 59:132–150.

Beal, M. J. (1998). VARIATIONAL ALGORITHMS FOR APPROXIMATE BAYESIAN INFERENCE. Technical report.

Binmore, K. (2007). *Game theory: a very short introduction*, volume 173. Oxford University Press.

Bloom, P. and German, T. P. (2000). Two reasons to abandon the false belief task as a test of theory of mind. *Cognition*, 77(1):B25–B31.

Bowles, S. and Gintis, H. (2011). *A cooperative species: Human reciprocity and its evolution*. Princeton University Press.

Bowles, S. and Gintis, H. (2013). A Cooperative Species. *Introductory Chapters*.

Butts, C. T. (2008). network: a Package for Managing Relational Data in R. *Journal of Statistical Software*, 24(2):1–36.

Camerer, C. F. (2003). Behavioural studies of strategic thinking in games. *Trends in Cognitive Sciences*, 7(5):225–231.

Chubaty, A. M. and McIntire, E. J. B. (2017). SpaDES: Develop and Run Spatially Explicit Discrete Event Simulation Models. *R package version*, 2(0).

Corts, K. S. (1998). Third-Degree Price Discrimination in Oligopoly: All-Out Competition and Strategic Commitment. *The RAND Journal of Economics*, 29(2):306.

Csardi, G. and Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695(5):1–9.

d'Arc, B. F., Marie, D., and Daunizeau, J. (2018). A reverse Turing-test for predicting social deficits in people with Autism. *bioRxiv*, page 414540.

Daunizeau, J. (2017a). Semi-analytical approximations to statistical moments of sigmoid and softmax mappings of normal variables. (Icm).

Daunizeau, J. (2017b). The variational Laplace approach to approximate Bayesian inference.

Daunizeau, J., Adam, V., and Rigoux, L. (2014). VBA: A Probabilistic Treatment of Nonlinear Models for Neurobiological and Behavioural Data. *PLoS Computational Biology*, 10(1):e1003441.

Daunizeau, J., den Ouden, H. E. M., Pessiglione, M., Kiebel, S. J., Friston, K. J., and Stephan, K. E. (2010a). Observing the Observer (II): Deciding When to Decide. *PLoS ONE*, 5(12):e15555.

Daunizeau, J., den Ouden, H. E. M., Pessiglione, M., Kiebel, S. J., Stephan, K. E., and Friston, K. J. (2010b). Observing the Observer (I): Meta-Bayesian Models of Learning and Decision-Making. *PLoS ONE*, 5(12):e15554.

Daunizeau, J., Friston, K., and Kiebel, S. (2009). Variational Bayesian identification and prediction of stochastic nonlinear dynamic causal models. *Physica D: Nonlinear Phenomena*, 238(21):2089–2118.

Dean, J. S., Gumerman, G. J., Epstein, J. M., Axtell, R. L., Swedlund, A. C., Parker, M. T., and McCarroll, S. (2000). Understanding Anasazi culture change through agent-based modeling. *Dynamics in human and primate societies: Agent-based modeling of social and spatial processes*, pages 179–205.

Devaine, M., Hollard, G., and Daunizeau, J. (2014a). The Social Bayesian Brain: Does Mentalizing Make a Difference When We Learn? *PLoS Computational Biology*, 10(12):e1003992.

Devaine, M., Hollard, G., and Daunizeau, J. (2014b). Theory of Mind: Did Evolution Fool Us? *PLoS ONE*, 9(2):e87619.

Devaine, M., San-Galli, A., Trapanese, C., Bardino, G., Hano, C., Saint Jalme, M., Bouret, S., Masi, S., and Daunizeau, J. (2017). Reading wild minds: A computational assay of Theory of Mind sophistication across seven primate species. *PLoS Computational Biology*, 13(11):e1005833.

Durrett, R. and Levin, S. (1994). The Importance of Being Discrete (and Spatial). *Theoretical Population Biology*, 46(3):363–394.

Ebert, C. (2009). Guest editor's introduction: how open source tools can benefit industry. *IEEE software*, 26(2):50–51.

Ehrentreich, N. (2007). *Agent-based modeling: The Santa Fe Institute artificial stock market model revisited*, volume 602. Springer Science & Business Media.

Epstein, J. M. (1998). Zones of cooperation in demographic prisoner's dilemma. *Complexity*, 4(2):36–48.

Epstein, J. M. (2002). Modeling civil violence: an agent-based computational approach. *Proceedings of the National Academy of Sciences of the United States of America*, 99 Suppl 3(suppl 3):7243–50.

Epstein, J. M., Cummings, D. A. T., Chakravarty, S., Singa, R. M., and Burke, D. S. (2002). Toward a containment strategy for smallpox bioterror: an individual-based computational approach. *Brookings Institution, CSED Working Paper*.

Fehr, E. and Fischbacher, U. (2004). Social norms and human cooperation. *Trends in cognitive sciences*, 8(4):185–190.

Foerster, J. N., Song, F., Hughes, E., Burch, N., Dunning, I., Whiteson, S., Botvinick, M., and Bowling, M. (2018). Bayesian Action Decoder for Deep Multi-Agent Reinforcement Learning.

Friston, K. (2010). The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience*, 11(2):127–138.

Friston, K., Kilner, J., and Harrison, L. (2006a). A free energy principle for the brain. *Journal of Physiology-Paris*, 100(1-3):70–87.

Friston, K., Mattout, J., Trujillo-Barreto, N., Ashburner, J., and Penny, W. (2006b). Variational free energy and the Laplace approximation.

Friston, K., Thornton, C., and Clark, A. (2012). Free-Energy Minimization and the Dark-Room Problem. *Frontiers in Psychology*, 3:130.

Gauthier, D. (1986). *Morals by agreement*. Oxford University Press on Demand.

Gilbert, G. N. and Hamill, L. (2009). Social circles: A simple structure for agent-based social network models. *Journal of Artificial Societies and Social Simulation*, 12(2).

Gintis, H. (2006). The emergence of a price system from decentralized bilateral exchange. *Contributions in Theoretical Economics*, 6(1):1–15.

Gintis, H. (2007). The Dynamics of General Equilibrium. *The Economic Journal*, 117(523):1280–1309.

Goldman, A. I., Margolis, E., Samuels, R., and Stich, S. (2012). *Theory of Mind Oxford Handbook of Philosophy and Cognitive Science*.

Green, L., Myerson, J., and Mcfadden, E. (1997). Rate of temporal discounting decreases with amount of reward. *Memory & Cognition*, 25(5):715–723.

Hamilton, W. D. (1967). Extraordinary sex ratios. *Science*, 156(3774):477–488.

Huth, A. and Wissel, C. (1992). The simulation of the movement of fish schools. *Journal of theoretical biology*, 156(3):365–385.

Kahneman, D. (2003). A perspective on judgment and choice: mapping bounded rationality. *American psychologist*, 58(9):697.

Kalbe, E., Grabenhorst, F., Brand, M., Kessler, J., Hilker, R., and Markowitsch, H. J. (2007). Elevated emotional reactivity in affective but not cognitive components of theory of mind: A psychophysiological study. *Journal of Neuropsychology*, 1(1):27–38.

Khalvati, K., Park, S. A., Mirbagheri, S., Philippe, R., Sestito, M., Dreher, J.-C., and Rao, R. P. N. (2018). Bayesian Inference of Other Minds Explains Human Choices in Group Decision Making. *bioRxiv*, page 419515.

Kidd, D. C. and Castano, E. (2013). Reading literary fiction improves theory of mind. *Science (New York, N.Y.)*, 342(6156):377–80.

Leek, J. (2015). How I decide when to trust an R package.

Marvin, R. S., Greenberg, M. T., and Mossler, D. G. (1976). The Early Development of Conceptual Perspective Taking: Distinguishing among Multiple Perspectives. *Child Development*, 47(2):511.

Mathys, C., Daunizeau, J., Friston, K. J., and Stephan, K. E. (2011). A Bayesian foundation for individual learning under uncertainty. *Frontiers in Human Neuroscience*, 5:39.

Maynard Smith, J. (1982). *Evolution and the theory of games*. Cambridge University Press.

McElreath, R. (2016). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, volume 122. CRC Press.

Mortensen, M. D. and Nymann, K. (2018). An investigation of 'Theory of Mind' and empathy when playing against artificial agents. *Aarhus University (unpublished exam assignment for cognition an commication at Cognitve Science)*.

Mossler, D. G., Marvin, R. S., and Greenberg, M. T. (1976). Conceptual perspective taking in 2- to 6-year-old children. *Developmental Psychology*, 12(1):85–86.

Nash, J. (1951). Non-Cooperative Games. Technical Report 2.

Nowak, M. and Sigmund, K. (1993). A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game. *Nature*, 364(6432):56–58.

Osborne, M. J. (2004). *An introduction to game theory*, volume 3. Oxford university press New York.

Peirce, J. W. (2007). PsychoPy—psychophysics software in Python. *Journal of neuroscience methods*, 162(1-2):8–13.

Premack, D. and Woodruff, G. (1978). Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1(04):515.

R Core Team (2013). R: A language and environment for statistical computing.

Rabinowitz, N. C., Perbet, F., Song, H. F., Zhang, C., Eslami, S. M. A., and Botvinick, M. (2018). Machine Theory of Mind.

RStudio (2018). Shiny - RStudio.

RStudio Team (2018). RStudio 1.2 Preview: Reticulated Python — RStudio Blog.

Schelling, T. (1978). Micromotives and Macrobehavior. *WW Norton & Company, New York,*.

Schneider, D., Nott, Z. E., and Dux, P. E. (2014). Task instructions and implicit theory of mind. *Cognition*, 133(1):43–47.

Shultz, T. R. and Cloghesy, K. (1981). Development of recursive awareness of intention. *Developmental Psychology*, 17(4):465–471.

Skewes, J., Håkonsson, D. D., Bilde, T., and Roepstorff, A. (2017). Informational Openness Enhances Decentralized Decision-Making: A Cognitive Agent Based Study.

Smith, J. M. and Price, G. R. (1973). The Logic of Animal Conflict. *Nature*, 246(5427):15–18.

Steingroever, H., Wetzels, R., and Wagenmakers, E.-J. (2013). Validating the PVL-Delta model for the Iowa gambling task. *Frontiers in Psychology*, 4:898.

Team, S. D. (2016). RStan: the R interface to Stan. *R package version*, 2(1).

Thiele, J. C. (2014). R Marries NetLogo: Introduction to the RNetLogo Package. *Journal of Statistical Software*, 58(2):1–41.

Woodruff, G. and Premack, D. (1979). Intentional communication in the chimpanzee: The development of deception. *Cognition*, 7(4):333–362.

Yoshida, W., Dolan, R. J., and Friston, K. J. (2008). Game theory of mind. *PLoS computational biology*, 4(12):e1000254.

Github links

Github link to package, liable to change: https://github.com/KennethEnevoldsen/SiRToM

Github link to bachelor project: https://github.com/KennethEnevoldsen/Bachelor

Appendix

Prisoner's Dilemma

The following is the classic payoff matrix for the prisoner's dilemma. This game have been of special interest in game theory, since both agents can obtain more by defecting as opposed to cooperating, as such dominant strategy is to defect although both a higher reward could have been obtain from cooperating, herein lies the dilemma (Axelrod and Hamilton, 1981).

Prisoner's Dilemma		Player 2	
		Cooperate	Defect
Player 1	Cooperate	5,5	0,3
	Defect	3,0	3,3

Battle of the sexes

The following is the payoff matrix for the battle of the sexes implemented in the package SiRToM. This game have been of interest in game theory, as it required coordination of behavior and good performance are contingent on the agent's ability to predict its opponent next move (Devaine et al., 2014b).

Battle of the sexes		Player 2	
		Opera	Football
Player 1	Opera	5,10	0,0
	Football	0,0	10,5

0-ToM's Behavioural Temperature

The following figure shows the performance of a 0-ToM against a RB over 10 simulations. The 0-ToM's used default parameters (see *3.2 Computational model*), with the exceptions of its behavioural temperature which were sampled from a distribution with mean $exp(-10) \approx 0$. The Random bias (RB) agents probability parameter was sampled from a distribution with a mean 1.39 and a standard deviation of 0.1 in logodds, i.e. it the probability of it choosing one was approximately 80%.

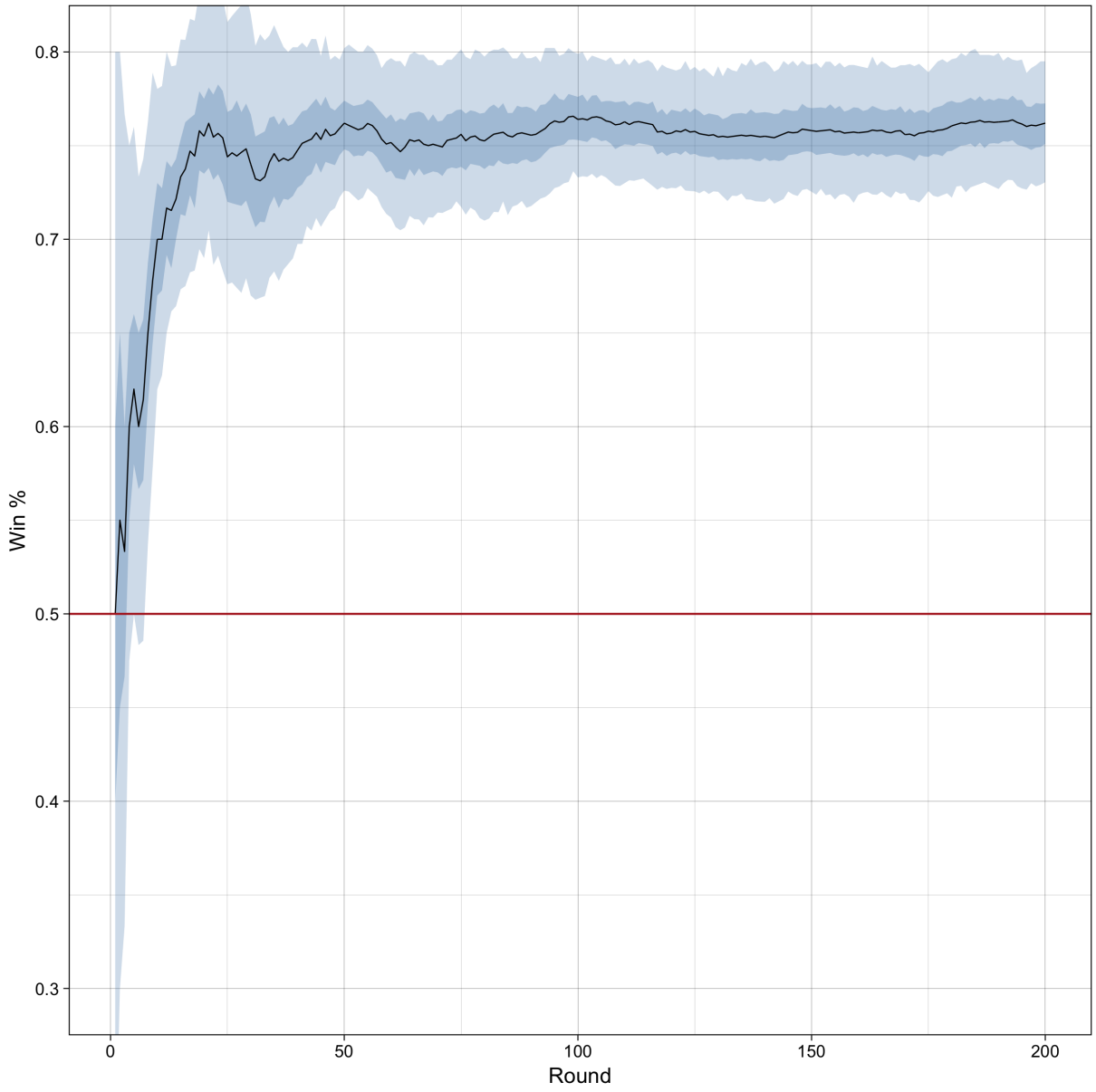

Figure 11: The win percentage of 0-ToM against a RB over 10 simulations.

1-ToM's parameter estimates of a 0-ToM

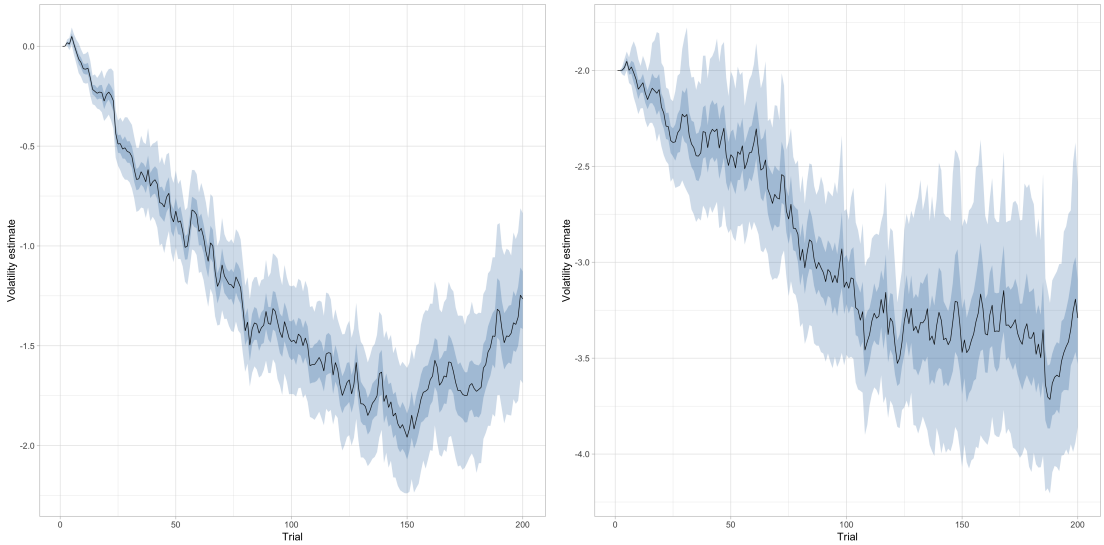

Figure 12: Left; 1-ToM's estimate of 0-ToM's volatility σ = -2, using default priors over 100 simulations. Right; 1-ToM's estimate of 0-ToM's volatility σ = -2, using optimally accurate priors over 40 simulations. Both over 200 trials. The light and dark blue intervals indicate 95% and 50% non-parametric bootstrapped confidence intervals (CI).

On figure 12, it can be seen that the updating patterns in 1-ToM's estimation of 0-ToM's parameters both in size and in direction are similar with or without accurate priors, but that the starting value makes the estimates be closer to the actual values.