Towards a "Theory of Mind" in Simulated Robots

Kyung-Joong Kim
Mechanical & Aerospace Engineering
Cornell University, Ithaca, NY 14853, USA
Department of Computer Engineering
Sejong University, Seoul,143-747, Korea
kimkj@sejong.ac.kr

Hod Lipson
Mechanical & Aerospace Engineering
Computing & Information Science
Cornell University
Ithaca, NY 14853, USA
hod.lipson@cornell.edu

ABSTRACT
The psychology term *Theory of Mind* (ToM) refers to the ability of an agent to recognize that an observed actor acts according to intentions and plans. In humans and some primates, ToM is fundamental to effective cooperation and competition, and is a key component of high-level cognition. In this paper, we explore the use of evolutionary robotics methods to create a robotic ToM. We use a co-evolutionary setup to evolve controllers that retrospectively explain an observed actor's behavior, and new actions that elicit new and more revealing behaviors. Evolved controllers can then be used to predict, manipulate and exploit the observed actor's behavior for cooperation or competition. Experimental results are shown in a physically-realistic simulation environment, and demonstrate an significant performance improvement compared to a direct estimation baseline.

Categories and Subject Descriptors
I.2.0 [Artificial Intelligence – General]: Cognitive Simulation

General Terms
Algorithms, Performance, Design, Reliability, Experimentation, and Verification

Keywords

Robotics, Evolutionary Computation, Estimation-Exploration Algorithm, Theory of Mind, Neural Network, Simulation

1. INTRODUCTION
The psychological term "Theory of Mind" (ToM) refers to the cognitive capacity that makes us understand others' internal states (intentions, goals and beliefs) and predict their future behaviors [1]. More precisely, it refers to the ability of one agent to have an explicit model of the mind of another agent. Such a model allows an agent to anticipate the other agent's behavior in both collaboration and competition. In competition, knowledge of another agent's mind is key to anticipating its actions, thereby allowing for development of strategies that counter offensives and exploit weaknesses. In collaboration, knowledge of another agent's mind is key to planning coordinated strategies, as well as compensating for potential weaknesses and taking advantage of mutual opportunities, while reducing communication. Theory of mind can be extended to multiple agent scenarios and group-level estimation, as well as to higher orders (such as, "I know what you know I know…"). Efficient algorithms for ToM could lead to better machine-machine interaction as well as machine-human interactions in cooperative and adversary situations.

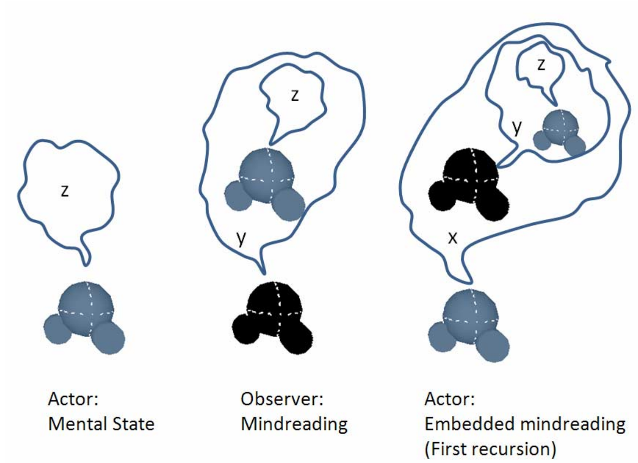


Figure 1. Robotic "Theory of Mind" (ToM)

Theory of mind is often cited as the key to higher level cognition tasks. From the observation of an actor's behavior, body language, facial expression, and speech, we can infer the person's internal states (emotions, thought, decision making, and plans). The mental states can be seen as part of one's self model. This function is supported by widely distributed areas in the human brain [2][3] and has been observed in chimpanzees [4].

ToM has gained great interest for human-computer interaction applications. Scassellati built functions for humanoid robots that enable finding faces and eyes and distinguishing animate from inanimate stimuli [5]. Buchsbaum *et al*. developed an anthropomorphic animated mouse character that uses his own behavior repositories to interpret other's behavior [6]. Hegel *et al*. studied human's theory of mind for different shapes of robots [7].

The current state of the art consists of modeling simple discrete mental states of an observed actor (for example, true/false or other finite number of states). Evolutionary robotics methods, however, are uniquely suitable for creating (synthesizing) complete controller models from scratch. However, instead of seeking controllers that exhibit some new behavior, we are interested in evolving a controller that reproduces existing behavior observed in an actor. If the controller is evolved correctly, we can use that controller to predict responses in future situations, thereby having an open-ended mechanism for forming a ToM

GECCO'09, July 8-12, 2009, Montréal, Québec, Canada.
ACM 978-1-60558-505-5/09/07.

Table 1. Verifying theories of "Theory of Mind" with neuroscience knowledge

References	Modular	Simulation	Theory-Theory
Gallese *et al.*[13]		Support	Opposition
Blakemore *et al.* [14]		Support	
Ramnani *et al.* [15]		Opposition	
Siegal *et al.* [3]	Support		
Saxe *et al.* [16]	Support		

Table 2. ToM applications in simulated environments

References	Other's Self	Modeling Methods	Tasks
Peters [17]	-	Symbolic Memory	Conversation
Kaliouby *et al.* [18]	Six Discrete States	Bayesian Networks	Mind Reading Dataset for Individuals with Autism
Buchsbaum *et al.* [6]	-	Action Recognition based on Simulation Theory	-
Bosse *et al.* [19]	BDI model	BDI (Belief-Desire-Intention) Modeling	Employer's Task Avoidance Scenario
Pynadath *et al.* [20]	Three Discrete States	Nested Belief Modeling on Agent-based Simulation	School Violence Scenario
Takano *et al.* [21]	-	Predicting Other's Velocity Vector	Collision-Avoiding
Zanlungo [22]	-	Predicting Other's Velocity Vector	Collision-Avoiding
Kondo *et al.* [23]	Eight Discrete Actions	Predicting Other's Discrete Action by a Neural Network	Carrying a Stick
Bringsjord *et al.* [24]	Two Discrete States	Logical Inference based on a ToM Statement	False-Belief Test

In this paper, we consider ToM in a robot-robot interaction setting. In this setup, each robot has its own controller and tries to construct a model of another robot's controller purely from observation, using only information of the other's movements. Each robot's controller is hidden from the other robot. This reconstruction process can be seen as reverse engineering of a black box. Figure 1 shows a conceptual diagram of robotic theory of mind with a recursive property. The motivation of this setup twofold: First, in many robotic environments access to an observed robot's controller is limited either by design, due to unanticipated failure, or due to legacy constraints. In adversarial situations, inferring a controller from observation may provide strategic value. Second, developing controller inference algorithms in robots may help in interaction with non-robotic actors such as humans, and may even shed light on fundamental questions in human-human ToM.

We used an incremental evolutionary algorithm is used to reverse engineer an accurate controller model from observations. While a variety of other machine learning techniques could be used, evolutionary algorithms have proven to be an effective method to train controllers for complex tasks, such as matching observed signals. To accelerate the learning, we used an active-learning processes where the observer robot actively collects training trajectories based on the disagreement between multiple candidate models, as per the Estimation-Exploration Algorithm (EEA) often used for reverse engineering problems [8][9].

The proposed method was tested in a simulated robot environment using on two physics engines. The first is PhysX and the second is EnKi for E-Puck robots from K-Team. The experimental results suggest that our approach can discover successful controller models for simulated robots, opening the door to future hardware implementations.

2. RELATED STUDIES

2.1 Theory of Mind

The first paper on ToM was published in 1978 by Premack and Woodruff asking "Does the chimpanzee have a theory of mind?" [1]. Subsequently there have been many articles on the ToM of human and non-human primates.

After over 30 years of research since the initial question, it is generally agreed that chimpanzees have a theory of mind but do not understand each other as humans do [4]. Herrmann *et al.* compared ToM ability among humans, chimpanzees, and orangutans with gaze following and intention understanding tasks [10]. Childhood autism is also related to the lack of theory of mind [11]. Baron-Cohen compared normal, autistic, and Down's syndrome subjects using a belief question to test theory of mind. The results for Down's syndrome and normal subjects were similar but 80% of autistic children failed the test.

Based on [12], theories for "theory of mind" are classified into four categories: Modular, simulation, theory-theory and executive function theories. In the modular view, the ToM is functionally dissociable from other cognitive functions and it is assumed that there are one or more neural structures specifically dedicated to this function. In the simulation perspective, there is no general theory guiding the ToM but the human brain mentally simulates another person's situation by placing itself into the other person's place. According to the theory-theory school, a child has a theory about how other minds operate and it evolves over time. Some theorists argue that a distinct ToM does not exist and executive functions are sufficient for the skills. We believe that developing and testing ToM models in robotics may help shed light on some of these complex and opaque questions.

2.2 ToM in Simulated Environments

There is a significant body of work on the use of ToM in simulated environments; however, the actor's self models representations in past research have been intentionally simple, typically allowing only a few discrete states.

Peters developed synthetic vision, memory, and theory of mind module for embodied conversational agents [17]. In his work, an agent has three states of ToM: "Have they seen me?", "Have they seen me looking?", and "interest level." Kaliouby *et al.* developed a "mind-reading machine" that recognizes six humans' discrete mental states from video input of the person's facial expression [18]. Buchsbaum *et al.* developed synthetic mouse characters that recognize other mouse's behavior based on their own repositories [6]. Bosse *et al.* proposed a two-level BDI (Belief, Desire and Intention) model for ToM [19]. The first level was used to model self's BDI and the other was for reasoning about other agents. Pynadath *et al.* developed a social simulation tool, PsychSim whose agents have beliefs about other agents [20]. Takano *et al.* [21] and Zanlungo [22] applied ToM to complex agent-based simulations and discussed about the effect of the level of ToM. Kondo *et al.* used the ToM in "carrying a stick task" for the cooperation of two computer programs [23]. Bringsjord *et al.*

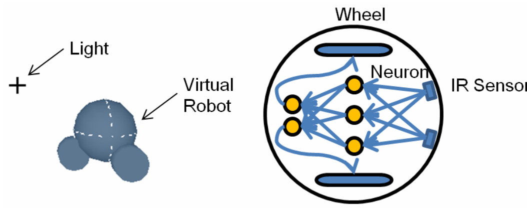


Figure 2. A virtual robot and neural controller

created a virtual character with a reasoning engine and they demonstrated that the character can pass the false-belief task by inserting "If someone sees something, they know it and if they don't see it, they don't" statement [24].

3. MODELING AN ACTOR

We performed our experiments using simple wheeled robots. Each robot has two sensors and wheels and detects light levels around the robot and uses them to control wheel velocities. An innate neural network processes the sensor inputs and generates outputs (wheel speed and/or steering angle), as shown in Fig 2.

We consider two types of robots: Actor robots and Observer robots. The goal of the observer robot is to learn a neural network (NN) model that is equivalent to the "innate" NN of actor. The process comprises four steps.

- **Step 1: Actor learning**
  - The first robot (the *actor*) learns to move towards light source by evolving an "innate" NN. This is not necessarily an efficient motion; it can be any arbitrary complex behavior.
- **Step 2: Observer learning**
  - A second robot (the *observer*) observes the actor's trajectory and uses the path to reverse engineer actor's innate NN. Additional paths help the observer refine or refute models of the actor's NN.
- **Step 3: Actor manipulation for learning**
  - The observer determines where to place the light to better expose actor's NN in order to best refine or refute models of the actor's NN.
  - If predictions are inaccurate, go to Step 2
- **Step 4: Actor exploitation**
  - The observer determines where to place the light source to elicit desired behavior from the actor (e.g. making the actor reach a specific target location).

3.1 Actor Learning

For the first step, we used an evolutionary algorithm to learn a model of the "innate" NN of observed actors. Initially, P neural networks were generated randomly. Weights (including bias weights) were generated from a uniform distribution over [-0.2, 0.2]. Each weight had a corresponding self-adaptive parameter initialized as 0.05. The mutation operator was $\sigma_i'(j) = \sigma_i(j)\exp(\tau N_j(0,1))$ $w_i'(j) = w_i(j) + \sigma_i'(j)N_j(0,1)$

where N_w is the number of weights. $\tau = 1/\sqrt{2\sqrt{N_W}}$. $\sigma_i(j)$ and $w_i(j)$ are a self adaptive parameter and value of jth weight of ith neural network. $N_j(0,1)$ is a standard Gaussian random variable re-sampled for every j. Each neural network generated one offspring using mutation yielding $2\times P$ neural networks (parents + offspring). The fitness of each NN was evaluated based on the distance to the light source. We allow only the fittest P NN to survive to the next generation.

3.2 Observer Learning

The observer learning (step 2) was also done by using the same evolutionary strategy as the actor learning. The merit (fitness) of each neural network was evaluated based on the similarity between the actor's trajectory and the one generated by the candidate network from the same initial condition. We used derivatives of the trajectories to measure similarity, where, Δx and Δy at each time step are compared. **Figure 3** shows the details of the distance measuring.

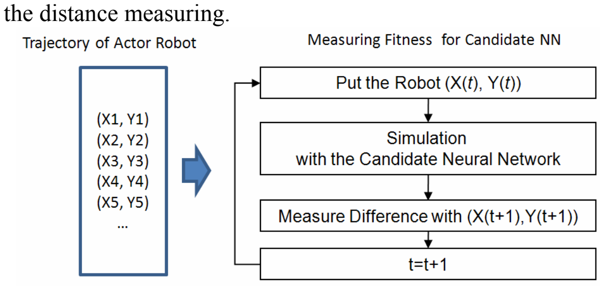


Figure 3. Distance measuring between two trajectories from the original and a candidate neural network

3.3 Actor Manipulation for Learning

In this step, the observer robot can manipulate the actor robot to obtain new trajectories for accelerated learning. In this case, the observer can change the position of the light source to elicit a different trajectory from the actor robot. Figure 4 shows the overall learning algorithm using manipulation. The disagreement between predicted trajectories is measured based on Euclidean distances of the (x,y) position vectors. The initial population of the Estimation Step is copied from the last Exploration Step. Each iteration adds one trajectory to the estimation bank (see Figure 4).

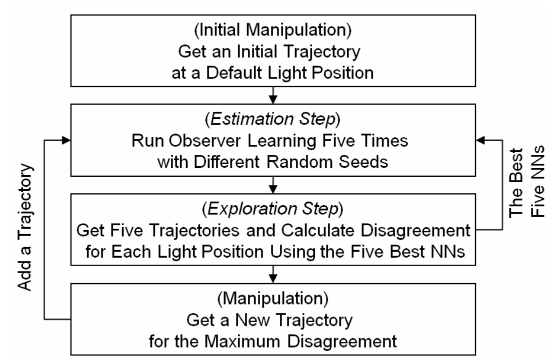


Figure 4. Estimation-Exploration Algorithm for Other's Self Modeling

The EEA returns the five best neural networks of the actor's self model. We can then choose the best one from the five NNs to predict the other's behavior. Alternatively, the ensemble of the five candidates can be used to provide more robust predictions as well as confidence estimates.

3.4 Actor Exploitation

Once the actor's self model is discovered, there could be several strategies to exploit that knowledge. For example, a robot could change the position of the light source to elicit a desired behavior. The goal of this experiment is to find a light position that will force the actor robot to go into a "trap" location. In the absence of any knowledge about the actor's controller, it is possible to make a straightforward guess that the actor will move towards the light source in a straight line. We call this strategy the "Straight Line Estimation" and it serves as our baseline comparison. With a "ToM" it is possible to make an educated guess that may be superior to the straight line estimation. The two approaches were compared for 100 different trap positions by changing the angle between the trap and the robot exhaustively. For each trial, the distance between the virtual robot and the possible light source position was kept constant.

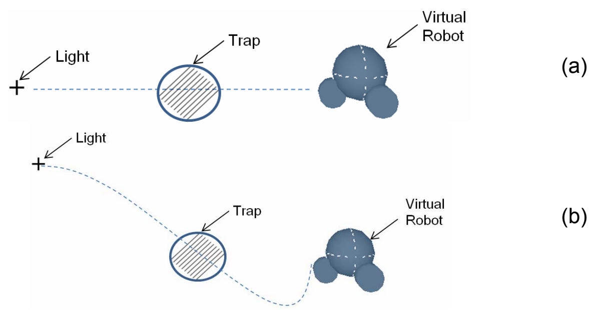


Figure 5. Two different strategies to estimate the behavior of the actor robot. (a) Straight Line Estimation assumes that the robot goes to light source following a straight line. (b) Theory of Mind places the trap based on estimates of the trajectory using the best available actor's model prediction

4. EXPERIMENTAL RESULTS

We performed two experiments: One where the actor was unable to sense the trap, and the second where the actor was able to sense the trap and was trained to avoid it, thereby making the manipulation by the observer more challenging.

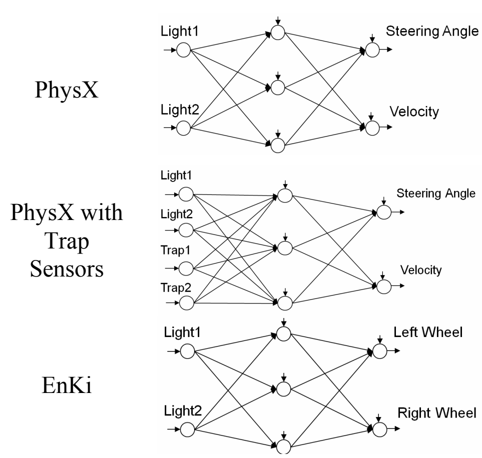
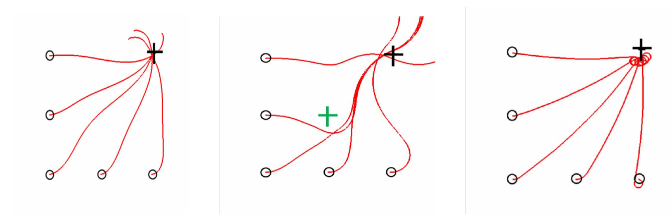


Figure 6. Architecture of neural networks

The first experiment was carried out in a PhysX simulator. The robot had two light sensors and three wheels. A neural controller outputs the speed and steering angle of a front wheel. In the second experiment the robot had additional sensors that detect a trap. Each neuron in the controller had a bias parameter and arctan function was used as a transfer function.

The parameters of experiments were as follows. In the actor learning, the population size was 20 and the maximum generation was 50. In the observer learning, the population size was 20 and the maximum generation was 30000 (PhysX) and 1000 (EnKi). The speed of PhysX simulation was accelerated using a simple heuristic equation which calculates the change of robot's angle. $\Delta\alpha=100\times|\text{speed}|\times(\text{steering angle})\times(\Delta\text{time})^2$.

Figure 7 shows the trajectories of the actor robot for different starting positions. It is interesting that the trajectory is not straight line, indicating that the theory of mind modeling is necessary to predict the behavior of the robot correctly. Figure 8 shows the total error change of the observer learning. Low total error means the behavior of the other's self model is close to the original actor's neural network. It shows that the total error gradually goes down for all environments.



(a) PhysX | (b) PhysX with trap sensors | (c) EnKi

Figure 7. Behavior of neural controllers from actor learning (Black Circle = Initial position, Black cross = Light and Green cross = Trap)

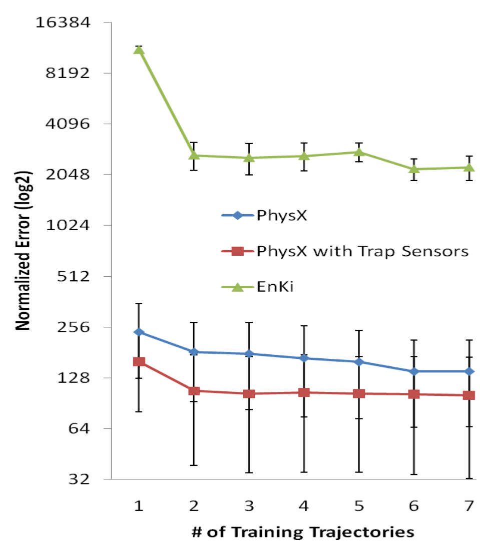


Figure 8. The progress of observer learning (averaged over 5 runs). The total error means the sum of trajectory distance between actor and other's self model for all possible light positions. Because each environment has different number of possible light positions and length of trajectory, they're normalized to show in one chart.

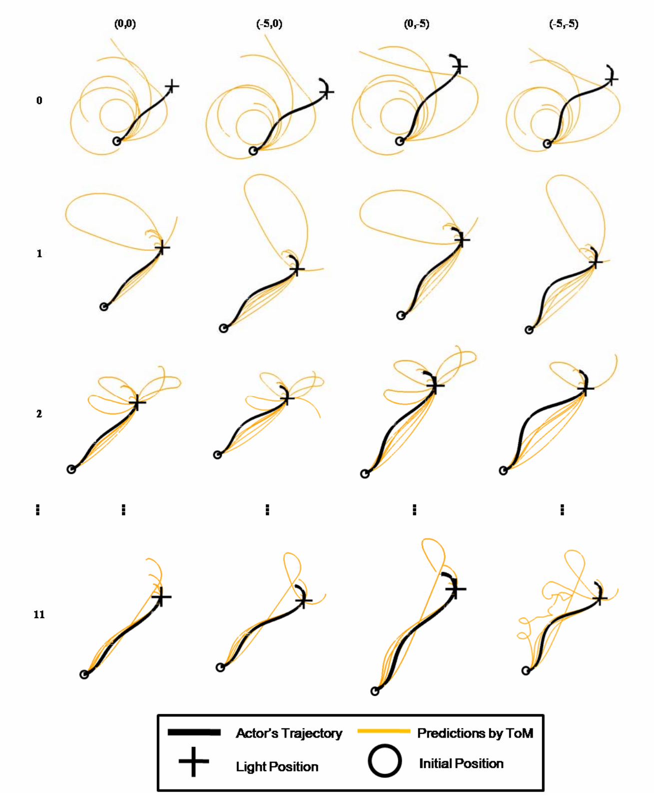


Figure 9. Progress of EEA learning for PhysX (Y-axis represents different starting positions and X-axis represents the number of training trajectories used. Initially, the trajectories of other's self model were quite different with the actor's one but it gradually became similar to the original one. When the number of trajectories was eleven, they showed quite similar behavior)

Table 3. Statistical results of experimentations. The results show the mean of total error and the standard errors. The bold means the best one among the three methods. Although the ensemble approach performs very well in PhysX environment, they fail to do so in the EnKi environment.

	Straight Line Estimation	ToM (Single neural network)	ToM (Ensemble of 5 neural networks)
PhysX	5.93 ± 0.54	3.87 ± 0.69	**1.10 ± 0.28**
PhysX with Trap Sensors	18.21 ± 2.60	18.29 ± 2.72	**11.97 ± 2.24**
EnKi	10.75 ± 1.26	**0.89 ± 0.30**	29.08 ± 5.75

Figure 9 shows the progress of other's self learning. The goal of this is to find a neural network that shows the most similar behavior to the original NN of actor. Initially, the trajectories of the candidates are nearly random and quite different with the original. This also shows the trajectories from multiple starting positions. At stage 1, their trajectories are close to the real one but there is still high error near the light source. At stage 2, their error has been reduced, and they are now closer to the original than previous stages. At stage 11, the candidates show the most similar behavior to the original one. This shows that our incremental evolutionary learning is working for this kind of problem.

Comparing the behavior and architecture between the actor's and the other's self model, we see that they have different weights but show very similar behaviors. There is a possibility that they are equivalent numerically. During the behavioral phase, they show very similar decisions (steering angle and speed) for most of sensory inputs (Figure 12).

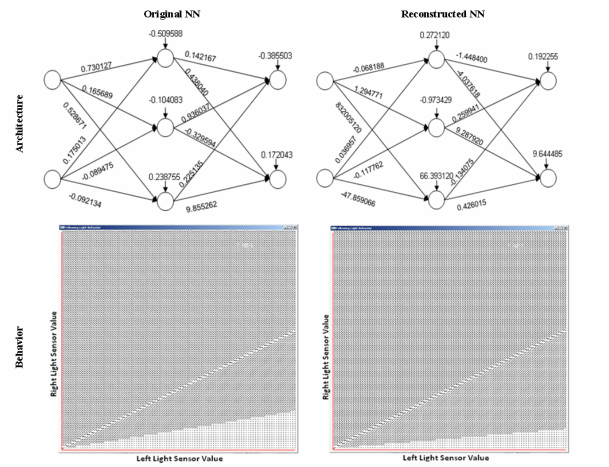


Figure 10. Comparison of actor's NN and reconstructed NN in the PhysX environment. Map shows the output of neural networks for all combination of sensory values.

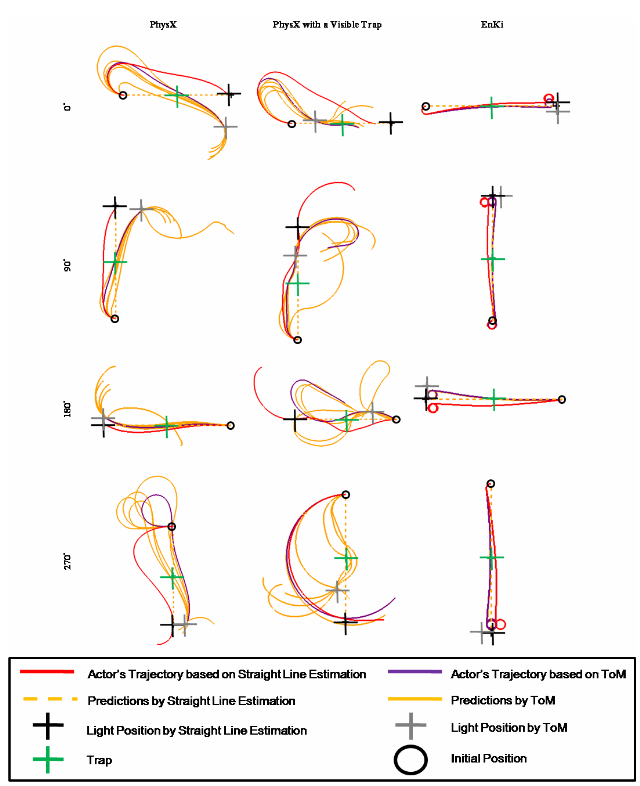


Figure 11. The results of exploitation

Figure 12. Comparison of actor's NN and reconstructed NN in the PhysX environment. Although they have different weight values, they're approximating a very similar function. Investigation on the scaling factors and topological symmetry could be interesting to understand their similarity. In behavioral part, the figure shows the output of neural networks for all combination of sensory values. Each output (speed and steering angle) is depicted as an arrow. They showed very similar output patterns. The narrow diagonal white line shows transition points of robot's decision.

Figure 11 shows the exploitation results using the neural networks found. If the prediction of other's self models is close to the real trajectory, it is possible to correctly estimate the position of light to lure the robot into the trap. When the prediction works well, the robot was forced to go into the trap. The error was measured as the distance to the trap when the actor moves the light source using its ToM. Our approach is compared with straight line estimation (SLE), which assumes the robot goes to the light source straightly. The error caused by ToM is much smaller than one by SLE. Table 3 shows statistical results for the three environments. The error of ToM is substantially lower than the SLE for all cases.

5. CONCLUSIONS & FUTURE WORK

In this paper, we have proposed computational approaches for theory of mind in simulated robots. We used both active and incremental learning using evolutionary computation. The experimental results show that our approach can reconstruct actor's neural network successfully in various configurations.

Our next step is to test our approach with real physical robots (e.g., E-Puck robots). This includes a number of challenges: 1) The accuracy of simulator 2) Tracking of robot's movement 3) Uncertainty in real world and 4) Finding movable light source.

6. ACKNOWLEDGMENTS

his work has been supported in part by the U.S. National Science Foundation (NSF) Creative-IT program, grant #0757478. K.-J.K. was supported by the Korea Research Foundation Grant (KRF-2007-357-D00220) funded by Korean Government (MOEHRD) and Korea Health 21 R&D Project, Ministry for Health, Welfare and Family Affairs (A040163)

7. REFERENCES

[1] Premack, D. G., and Woodruff, G. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1 (1978), 515-526.

[2] Zimmer, C. How the mind reads other minds. *Science*, 300 (16 May 2003), 1079-1080.

[3] Siegal, M., and Varley, R. Neural systems involved in 'theory of mind.' *Nature Reviews – Neuroscience*, 3 (June 2002), 463-471.

[4] Call, J., and Tomasello, M. Does the chimpanzee have a theory of mind? 30 years later. *Trends in Cognitive Sciences*, 12, 5 (2008), 187-192.

[5] Scassellati, B. *Foundations for a Theory of Mind for a Humanoid Robot*. Ph.D. Thesis, Massachusetts Institute of Technology (2001).

[6] Buchsbaum, D., Blumberg, B., Breazeal, C., and Meltzoff, A. N. A simulation-theory inspired social learning system for interactive characters. *IEEE International Workshop on Robots and Human Interactive Communication* (2005), 85-90.

[7] Hegel, F., Krach, S., Kircher, T., Wrede, B., and Sagerer, G. Theory of mind (ToM) on robots: A functional neuroimaging study. *Proceedings of the 3rd ACM/IEEE International Conference on Human Robot Interaction* (2008), 335-342.

[8] Bongard, J., and Lipson, H. Automated reverse engineering of nonlinear dynamical systems. *Proceedings of the National Academy of Science*, 104, 24 (2007), 9943-9948.

[9] Bongard, J., and Lipson, H. Resilient machines through continuous self-modeling. *Science*, 314, 5802 (2006), 1118-1121.

[10] Herrmann, E., Call, J., Hernandez-Lloreda, M. V., Hare, B., and Tomasello, M. Humans have evolved specialized skills of social cognition: The cultural intelligence hypothesis. *Science*, 317 (2007), 1360-1366.

[11] Baron-Cohen, S. *Mindblindness*. MIT Press (1997).

[12] Youmans, G.L. *Theory of Mind in Individuals with Alzheimer-Type Dementia Profiles*. Ph.D. Thesis of College of Communication at the Florida State University (2004).

[13] Gallese, V., and Goldman, A. Mirror neurons and the simulation theory of mind-reading. *Trends in Cognitive Sciences*, 2, 12 (1998), 493-501.

[14] Blakemore, S. J., and Decety, J. From the perception of action to the understanding of intention. *Nature Reviews- Neuroscience*, 2 (2001), 561-567.

[15] Ramnani, N., and Miall, R. C. A system in the human brain for predicting the actions of others. *Nature Neuroscience*, 7, 1 (2004), 85-90.

[16] Saxe, R., Carey, S., and Kanwisher, N. Understanding other minds: Linking developmental psychology and functional neuroimaging. *Annual Review of Psychology*, 55 (2004), 87-124.

[17] Peters, C. A perceptually-based theory of mind for agent interaction initiation. *International Journal of Humanoid Robotics*, 3, 3 (2006), 321-339.

[18] Kaliouby, R. E., and Robinson, P. Mind reading machines: Automated inference of cognitive mental states from video. *IEEE International Conference on Systems, Man and Cybernetics* (2004), 682-688.

[19] Bosse, T., Memon, Z. A., and Treur, J. A two-level BDI-agent model for theory of mind and its use in social manipulation. *Proceedings of the Artificial and Ambient Intelligence Conference* (2007), 335-342.

[20] Pynadath, D. V., and Marsella, S. C. PsychSim: Theory of mind with decision-theoretic agents. *Proceedings of the International Joint Conference on Artificial Intelligence*, (2005), 1181-1186.

[21] Takano, M., and Arita, T. Asymmetry between even and odd levels of recursion in a theory of mind. *Proceedings of ALIFE X* (2006), 405-411.

[22] Zanlungo, F. A collision-avoiding mechanism based on a theory of mind. *Advances in Complex Systems*, 10, 2 (2007), 363-371.

[23] Kondo, K., and Nishikawa, I. The role that the internal model of the others plays in cooperative behavior. *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication* (2003), 265-270.

[24] Bringsjord, S., Shilliday, A., Werner, D., Clark, M., Charpentier, E., and Bringsjord, A. Toward logic-based cognitively robust synthetic characters in digital environments. *Proceedings of the First Artificial General Intelligence* (2008), 87-98.