Abstract

This thesis concerns the automated reverse engineering of agent behaviors. It proposes a metric-free coevolutionary approach—*Turing Learning*, which allows a machine to infer the behaviors of agents (simulated or physical ones), in a fully automated way. In our approach, a population of candidate models competitively coevolves with a population of classifiers. The classifiers observe the models and agents. The fitness of the classifiers depends solely on their ability to discriminate between the models and agents. Conversely, the fitness of the models depends solely on their ability to 'trick' the classifiers into categorizing them as agents. As such, the approach does not require any predefined metrics to quantify the difference between the models and agents.

The merits of *Turing Learning* are demonstrated using three case studies. In the first case study, the machine automatically infers the behavioral rules of a group of homogeneous agents only through observation. A replica, which resembles the agents under investigation in terms of behavioral capabilities, is mixed into the group. The models are to be executed on the replica. The classifiers observe the motion of each individual in the swarm for a fixed time interval. Based on the individual's motion data, a classifier makes a judgment indicating whether the individual is believed to be an agent or the replica. The classifier gets a reward if and only if it makes the correct judgment. The models, on the other hand, are evolved to mimic the behavior of the agents and mislead the judgment of the classifiers. In the second and third case studies, *Turing Learning* is applied to infer deterministic and stochastic behaviors of a single agent through controlled interaction, respectively. In particular, the machine is able to modify the environmental stimuli (to which the agent responds) and thereby interact with the agent. This allows the machine to reveal the agent's entire behavioral repertoire. This interactive approach proves superior to learning through passive observation.

1 Introduction

Over the last 50 years, robotic and automation systems have transformed our world and greatly enhanced the quality of our daily life. With the development of science and technology, many intelligent systems that integrate machines, electronics, control and information technologies have emerged. Such systems can accomplish numerous tasks originally performed by humans and often prove superior in terms of precision, speed and cost. They can replace humans in the tasks that require repetitive and monotonous operations. For example, in the automotive industry, robotic and automation systems have been widely used for machining, welding, painting and assembling. From the point of engineering, these systems have lowered the product cost and improved the efficiency of production, thus greatly increasing the speed of industrialization.

Robotic and automation systems also contribute to scientific research, especially in situations that require to conduct experiments in dangerous environments (e.g., nuclear factory) which are hazardous to human beings or some operating environments that may be beyond humans' capabilities of reach (e.g., other planets). In 2004, two famous robots—*Spirit* and *Opportunity* were sent to Mars by NASA in a mission to explore the geology of this planet [1]. The primary goal of this mission is to analyze the rocks and soils in Mars to seek potential exist of water. With the help of automation systems, researchers can collect data much faster than ever before. For instance, high-throughput screening (HTS) systems [2], which are widely used in drug discovery, allow the researchers to conduct millions of experiments in a very short time. Such systems consist of several components, including sensing units, robotic manipulator, control system, etc. Besides data collection, these systems can also help analyze the data using intelligent software, which provides an ideal tool for data analysis in scientific research and frees researchers from the tedious and monotonous process of data analysis if done manually. This accelerates the development of scientific research to a great extent.

1.1 Motivation

A particular scientific area that robotic and automation systems play a significant role is *ethology*, which is the study of animal behavior [3]. Ethology is pursued not only because it is a subject of interest in itself, but also because the knowledge gained from it has several practical applications. For instance, models of animals' decision-making process can be used to predict their behavior in novel environments, which can help in making ecological conservation policy [4]. Knowledge about animal behavior has also been applied for solving computational problems [5], and for constructing biologically-inspired robotic agents [6]. There are four types of questions to be investigated in ethology: questions concerning causes, functions, development and evolution [3]. Causes refer to the mechanisms of animals that are innate as well as the external/internal stimuli that affect such behavior. Functions concern what is the purpose of this behavior such as mating, aggregation or foraging. The development of animal behavior concerns how animals learn such behavior during their life time as well as how such behavior is affected by experience, while evolution relates to how the behavior changes over generations in the course of natural evolution. Over centuries, these four questions have been investigated by ethologists primarily in well-controlled laboratories or outdoor environments.

Before the emergence of computers, ethologists needed to observe the animals and analyze the data manually. They also needed to learn how to control the environmental conditions in a meaningful way to extract most of the information from the animals under investigation. Such process of analysis sometimes is time-consuming and tedious. With the help of intelligent and automation systems, nowadays researchers can conduct experiments much more efficiently. However, these systems are often secondary, and in most of the cases they are merely accomplishing mechanical and repetitive work. The question is whether we can build a machine/system that can dominate the whole process of scientific investigation and automatically analyze experimental data, search for correlations between different elements, generate new hypotheses and devise new experimental conditions to be investigated. In other words, can we build a system which is able to automatically conduct scientific research without (or with minimal) human intervention? Recently, the emergence of "robot scientists" shows that such systems may be within reach [7, 8, 9]. Following this motivation, this thesis aims to pave the way for further development in science automation [8], especially in the area of ethology. The ultimate goal of this thesis is to contribute to the study of animal behavior through developing an automated system identification/modeling system.

System identification is a process of modeling natural or artificial systems through observed data. It has drawn a large interest among researchers for decades [10, 11]. One application of system identification is the reverse engineering of agent behavior (biological organisms or artificial agents). Many studies have investigated how to deduce rules of agent behavior using system identification techniques based on various models [12], one of which is agent-based model [13] that simulates the complex behavior of a group of agents with relatively simple behavioral rules. The global behavior emerging from interaction within agents and between agents and environments is used for refining the model. Evolutionary computation which draws inspiration from biological evolution has proven to be a powerful method to automate the generation of models, especially for behaviors that are hard to formulate [14, 15, 16]. Evolutionary computation also provides a potential realization for automation science, as models evolve in an autonomous manner. It is the main technique that is investigated in this thesis for performing system identification.

A limitation of current system identification methods is that they rely on predefined metrics, such as the square error, to measure the difference between the output of the models and that of the system under investigation. Model optimization then proceeds by minimizing the measured differences. However, for complex systems, defining a metric can be non-trivial and case-dependent. It may require significant prior information about the systems. Moreover, an unsuitable metric may not distinguish well between good and bad models, or even bias the identification process. This thesis overcomes these problems by introducing a system identification method that does not rely on predefined metrics.

1.2 Problem Statement

The agent (animal)[1] to be studied is put in an environment. Its behavior depends on interaction with the environment and with other agents in a group (if any). In general, one could monitor a range of variables including the agent's motion, heart rate, morphology, body temperature, etc. In this thesis, the machine will observe the agent's motion, and assumes that it is possible to track the position and orientation of the agent at discrete steps in time. The system identification task is to infer the observed behavior, in other words, the agent's behavioral rules through observation. In the following, we present the three case studies investigated in this thesis.

---

[1]Since the behaviors under investigation in this thesis are simulated using computer simulation or physical robots, throughout the thesis, unless otherwise stated, we used the term agent.

The first case study is about inferring swarm behaviors, which are emergent behaviors that arise from the interactions of numerous simple individuals [17]. Learning about behaviors that are exhibited in a collective manner is particularly challenging, as the individuals not only interact with the environment but also with each other. Typically their motion appears stochastic and is hard to predict [18]. For instance, given a swarm of simulated fish, one would have to evaluate how close its behavior is to that of a real fish swarm, or how close the individual behavior of a simulated fish is to that of a real fish. Characterizing the behavior at the level of the swarm (that is, an emergent behavior) is challenging [19]. It may require domain-specific knowledge and not discriminate among alternative individual rules that exhibit similar collective dynamics [20]. Characterizing behavior at the level of individuals is also difficult, as even the same individual fish in the swarm is likely to exhibit a fundamentally different trajectory every time it is being looked at. In this case study, the machine observes the motion of each individual in the swarm and needs to automatically infer the behavioral rules of the swarming agents.

The second case study is about inferring the deterministic behavior of a single agent. In particular, we investigate how the agent responds to an environmental stimulus. The agent behavior under investigation has low observability. That is, only randomly generating sequences of input (stimulus) in a passive way is not sufficient to reveal all the hidden information of the agent and infer its behavior. Instead, in order to extract all the agent's behavioral repertoire, the machine needs to interact with the agent in an active way and construct complex patterns of stimulus that help reinforce the learning process. The machine has full control over the environmental stimulus, and at the same time observes the agent's motion.

In the third case study, we investigate how to infer the stochastic behavior of a single agent. In this case, the agent still responds to the environmental stimulus; however, its behavior is not only determined by the environmental stimulus but also some probability. In other words, constructing a fixed sequence of stimulus may not trigger all the agent's behavioral repertoire as in the case of investigating the deterministic behavior. The machine needs to interact with the agent during the experimental process and dynamically change/control the environmental stimulus based on the agent's observed motion to reveal its hidden behavior. Allowing a machine to exhibit such 'intelligent' behavior and infer the agent's behavioral rules in an autonomous manner is challenging. It is hence investigated in this thesis as a case study.

1.3 Contributions

The main contribution of this thesis is a novel system identification approach—*Turing Learning* which allows a machine to infer agent behavior in an autonomous manner. *Turing Learning* uses a coevolutionary algorithm comprised of two populations. A population of candidate models competitively coevolves with a population of classifiers. The classifiers observe the models and agents. The fitness of the classifiers depends solely on their ability to discriminate between them. Conversely, the fitness of the models depends solely on their ability to 'trick' the classifiers into categorizing them as agents. Unlike other system identification methods, *Turing Learning* does not rely on predefined metrics to gauge the difference between the behaviors of agents and models.

The particular contributions are as follows:

1) A metric-free approach to automatically infer the behavioral rules of a group of homogeneous agents. Both the inferred model parameters and emergent global behaviors closely matched those of the original swarm system.

2) A metric-free approach to automatically produce a classifier system that can be used for detecting abnormal behaviors (e.g., faulty agents in a swarm). This was validated for both simulated and physical robot swarms.

3) A physical metric-free coevolutionary system for performing system identification. The system was validated through successfully inferring the behavioral rules of a physical robot swarm.

4) A metric-free approach to automatically infer deterministic behavior of a single agent by interacting with it, rather than simply observing its behavior in a passive manner. This interactive approach proves superior to learning through passive observation.

5) A metric-free approach to automatically infer stochastic behavior of a single agent through controlled interaction. The machine dynamically changes the environmental stimulus to interact with the agent on the fly. The results are shown to be better than those obtained using two metric-based methods.

1.4 Publications

This thesis presents the author's own work. Some parts of the thesis have been published as original contributions to the scientific research. A preliminary work of Chapter 3 was orally presented in a conference by the author:

- **W. Li**, M. Gauci and R. Groß, "Coevolutionary learning of swarm behaviors without metrics," *Proceedings of 2014 Genetic and Evolutionary Computation Conference (GECCO 2014)*. ACM Press, Vancouver, Canada, 2014, pp. 201–208.

A preliminary work of Chapter 5 was orally presented in a conference by the author of this thesis:

- **W. Li**, M. Gauci and R. Groß, "A coevolutionary approach to learn animal behavior through controlled interaction," *Proceedings of 2013 Genetic and Evolutionary Computation Conference (GECCO 2013)*. ACM Press, Amsterdam, Netherlands, 2013, pp. 223–230.

A part of Chapters 3 and 4 has been written as a paper and submitted to the following journal:

- **W. Li**, M. Gauci, J.Chen and R. Groß, "Reverse Engineering Swarm Behaviors Through Turing Learning," *IEEE Transactions on Evolutionary Computation*, under review.

Apart from the work presented in this thesis, the author has also contributed to some other projects. This led to the following publications:

- M. Gauci, J. Chen, **W. Li**, T. J. Dodd, and R. Groß, "Self-organized aggregation without computation," *The International Journal of Robotics Research*, vol. 33, no. 8, pp. 1145–1161, 2014.

- J. Chen, M. Gauci, **W. Li**, A. Kolling and R. Groß, "Occlusion-based cooperative transport with a swarm of miniature mobile robots." *IEEE Transactions on Robotics*, vol.31, no.2, pp. 307–321, 2015.

- M. Gauci, J. Chen, **W. Li**, T. J. Dodd, and R. Groß, "Clustering objects with robots that do not compute," in *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014)*. IFAAMAS Press, Paris, France, 2014, pp. 421–428.

During his PhD studies, the author has also been a Marie Curie Research Fellow with the Department of Mechanical Engineering, University of Western Ontario, Canada, where he contributed to the project of Mechanical Cognitivization. This led to the following publication:

- G. Avigad, **W. Li**, A. Weiss, "Mechanical Cognitivization: A kinematic system proof of concept" *Adaptive Behavior*, vol.23, no.3, pp. 155–170, 2015.

1.5 Thesis Outline

This thesis is structured as follows:

- Chapter 2 describes the background of the thesis as well as the related work presented in the literature.

- Chapter 3 introduces the metric-free system identification method—*Turing Learning*. It is applied to learn two swarm behaviors (self-organized aggregation [21] and self-organized object clustering [22]) through observation. This chapter is organized as follows. Section 3.1 describes the implementation of *Turing Learning* (Section 3.1.1) and the two swarm behaviors (Section 3.1.2) investigated in this thesis. Section 3.2 introduces the simulation platform (Section 3.2.1) and simulation setups (Section 3.2.2) for performing coevolution runs. Section 3.3 presents the results obtained from the two case studies. Section 3.3.1 systematically analyzes the evolution of models, through objectively measuring the quality of the evolved models in terms of local and global behaviors. Section 3.3.2 investigates the coevolutionary dynamics. Section 3.3.3 systematically investigates the evolution of classifiers, showing how to construct a robust classifier system to potentially detect abnormal behaviors in the swarm. Section 3.3.4 studies the effect of observing only a subset of agents in the swarm and the results obtained. Section 3.3.5 presents a study where an aspect of the agents' morphology (their field of view) and brain (controller) are inferred simultaneously. Section 3.3.6 shows the results of using *Turing Learning* to learn other swarm behaviors.. Section 3.3.7 presents a noise study. Section 3.4 summaries the findings in this chapter.

- Chapter 4 presents a real-world validation of *Turing Learning* to infer the behaviors of a swarm of physical robots. Section 4.4 introduces the physical platform,

which includes the robot arena (Section 4.1.1), the robot platform and sensors implementation (Section 4.1.2). Section 4.2 details the tracking system, including motion capture (Section 4.2.1) and video processing (Section 4.2.2). Section 4.3 describes the programs executed by each component (machine, agent and replica) during the coevolutionary learning process. Section 4.4 describes the experimental setup. Section 4.5 discusses the results obtained. This includes the analysis of the evolved models (Section 4.5.1) and the analysis of the evolved classifiers (Section 4.5.2). Section 4.6 analyzes the sensitivity of *Turing Learning* for individual failure during the experimental process. Section 4.7 summaries the results obtained and discusses the findings in this chapter.

- Chapter 5 presents two case studies to infer deterministic and stochastic behavior of a single agent, respectively. In these case studies, the machine not only observes the behavior of the agent but also interacts with the agent through changing the stimulus that influences the agent's behavior. Section 5.3.1 describes the deterministic behavior under investigation. Section 5.3.3 discusses the results obtained for inferring the deterministic behavior. Section 5.4.1 presents the stochastic behavior under investigation. Sections 5.4.3 and 5.4.4 discuss the results obtained for inferring stochastic behavior. Section 5.5 summaries the findings in this chapter.

- Chapter 6 concludes the thesis and discusses the future work.