1 Introduction

1.1 Motivation

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.

an economical? prospective? or business?

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.

structure not clear, rephrase

on?

on?

grammar!s

too strong claim

sentence not well connected to previous one

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.

Turing Learning consists of two populations. A population of 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 distinguish between them. The models, on the other hand, are evolved to mimic the behavior of the agents and mislead the judgment of the classifiers. The fitness of the models depends solely on their ability to 'trick' the classifiers into categorizing them as agents. Unlike other methods for system identification, *Turing Learning* does not require any predefined metrics to quantitatively measure the difference between the models and agents.

The merits of *Turing Learning* are demonstrated using three case studies. In the first case study, a 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. 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.~~ 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 all of the agent's entire behavioral repertoire and help reinforce the learning process. This interactive approach proves superior to learning only through observation.

this is the case for all 3 case studies and is repetitive — remove?

Say instead that the case study is conducted with both simulated and physical robots.

groups of

in which

cation 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 an agent-based model [13], which 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 existing system identification methods is that they rely on predefined metrics, such as the sum of squared 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 much prior information about the systems. Moreover, an unsuitable metric may not well distinguish between good and bad models, or even bias the identification process. This thesis solves these problems by introducing a system identification method that does not rely on predefined metrics.

a bit strange that you solve problem before stating them

1.2 Problem Statement

The investigated (agent) behaviors in this thesis are simulated by ~~using~~ computers or physical robots. The agent 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). 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. In general, one could monitor a range of variables including the agent's motion, heart rate, morphology, body temperature, etc. Furthermore, the machine ~~could~~ also control the environmental stimuli that the agent responds to. The system identification task is to infer the observed behavior, in other words, the agent's behavioral rules.

a ?

Three case studies are presented in this thesis. The first case study is about inferring swarm behaviors, which are emergent behaviors that arise from the interactions

It is

may

can, (not could)

A particular scientific area ~~that~~ [in which] 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 and 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.

[do these question relate to ethology on its own, or gaining knowledge for applicat. (structure of paragraph not clear)]

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 accomplish 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~~ [that] is able to automatically conduct scientific research without (or with minimal) human intervention? Recently, the development 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 method.

System identification is a process of modeling natural or artificial systems with observed data. It has drawn a large interest among researchers for decades [10, 11]. One appliof 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 difficult 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]. Comparing 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.

(comparing

s

s

(nice and consistent)

the

may

The second case study is about inferring deterministic behaviors of a single agent, and investigating how the agent responds to ~~the~~ environmental stimuli. The machine has full control over the environmental stimuli that the agent responds to, and at the same time observes the agent's motion. We investigate a particular type of agent behavior; from the perspective of system identification, the agent behavior has low observability. That is, ~~only~~ randomly generating sequences of inputs (stimuli) ~~is~~ not sufficient to reveal all the hidden information of the agent and ~~to~~ 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 stimuli that help reinforce the learning process.

s

(particular?

ed

thus

are

In the third case study, we investigate how to infer stochastic behaviors of a single agent. The agent still responds to the environmental stimuli; however, its behavior is not only determined by the environmental stimuli but also ~~some~~ probability. In other words, constructing a fixed sequence of stimulus may not trigger all the agent's behavioral repertoire as ~~mentioned above~~. 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. Inferring such stochastic behaviors through predefined metrics are challenging, as given the same sequence of inputs (stimuli), the agent ~~would probably~~ exhibit different behaviors.

but

case study 2:

is there true interaction? only via evolution?

is

is likely to

true ✓

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 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 specific 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 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 by both simulated and physical robot swarms.

directly

physical system?

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. counterpart

in the physical world?

metric-free

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.

OK

OK!

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.

Remove. (you already say passive in existence)

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

version

IEEE submission? Worth it check as you still can demote if needed!

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~~ version 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.

The!

Preliminary version of

included in?

~~A part of~~ Chapters 3 and 4 ha~~s~~ve been ~~written as~~ a paper ~~and~~ submitted to the following journal:

I space

- **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.

space

- 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 a on

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.

Space!

1.5 Thesis Outline

The!

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~~ a 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]) ~~only~~ 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)~~. 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. In particular, 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~~ studies sensitivity to noise. Section 3.4 summaries the findings in this chapter.
- Chapter 4 presents a real-world validation of *Turing Learning* to infer the behavior of a swarm of physical robots. Section 4.1 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

the

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.

cap~ ?

- Chapter 5 investigates how to infer the deterministic and stochastic behaviors of an agent through *Turing Learning* with interaction. The machine not only observes the behavior of the agent but also interacts with it through changing the stimulus that influences the agent's behavior. This chapter is organized as follows. Section 5.2 describes the methodology, illustrating how *Turing Learning* is extended to have interactive ability. The deterministic and stochastic behaviors under investigation are presented as two case studies (Sections 5.3 and 5.4). Section 5.3.1 describes the deterministic behavior. Section 5.3.2 presents the simulation setup. Section 5.3.3 presents the results of inferring the deterministic behavior. Section 5.4.1 describes the stochastic behavior for the general case using a state machine. Section 5.4.2 presents the simulation setup for inferring the stochastic behavior. Section 5.4.3 and 5.4.4 present the obtained results for the case of 2 states and 3 states, respectively. Section 5.5 summaries the chapter.
- Chapter 6 concludes the thesis and discusses the future work.

of?

s

2 Background and Related Work

repetitive

This chapter presents the background and related work. In Section 2.1, we introduce the background of this thesis. This includes a historical perspective of AI, how AI and robotics are combined, and the research that has been done in recent years in the areas of automation science. In Section 2.2, we review the field of evolutionary computation that is the main technique used in this thesis. This section includes an introduction of natural evolution/coevolution, principles of evolutionary algorithms and coevolutionary algorithms, and their applications. As the theme of this thesis is to show how machine intelligence can be used for learning animal behaviors, Section 2.3 introduces how AI/robotics and animal behavior study benefit from each other. Section 2.3.1 gives some examples of single and swarm behaviors observed in nature. Section 2.3.2 reviews how animal behavior can be used as inspiration for AI and robotics, and introduces one such example—swarm optimization and swarm robotics. Section 2.3.3 introduces two methods to study animal behaviors using AI/robotics techniques and reviews the related work.

2.1 Background

2.1.1 AI and Robotics

2.1.1.1 The Development of AI

Intelligence is a natural part of life. Humans can exhibit many behaviors such as pattern recognition and decision making that can be considered as intelligence. However, intelligence is not a property that can only be limited to biological creatures. It should be equally applicable to computers or machines. The term of Artificial Intelligence related with machine intelligence emerged in a conference in 1956 at Dartmouth College, where several pioneers of this filed including Marvin Minsky, John McCarthy, etc., discussed the development of digital computer and the future of AI. The definition of AI is still a disputed topic. Some researchers argue that AI is to simulate the intelligent behaviors which are observed in humans and other creatures using computers or machines. That is, an intelligent machine should be able to exhibit the behavior similar to that of humans when encountering the same problems [23]. Others gave the following definition: "Artificial Intelligence is the study of mental faculties through the use of computational models" [24]. According to Fogel [25], an intelligent system should know how to make decision in order to fulfill a goal (e.g. solving a problem). In other words, instead of pre-programming the machine using human's knowledge, the machine should be able to learn and adapt. In [26], Minsky even argues, "Why can't we build, once and for all, machines that grow and improve themselves by learning from experience? Why can't we simply explain what we want, and then let our machines do experiments or read some books or go to school, the sorts of things that people do?" In 1950, Turing [27] proposed a imitation game which is nowadays known as *Turing test* to discuss a question: "Can machine think?". Although whether a machine could pass the *Turing test* or not is beyond the consideration at that time, it was accepted as a notion that a machine could mimic human behavior. Many promising achievements have been made to enable machines to do a variety of intelligent things since then.

In the 1970s, the emergence of expert system [28], which mimics a human expert's decision-making capability, significantly promoted the development of AI. These expert systems can solve complicated problems through reasoning about the knowledge encountered mainly based on *if-else* rules. One of the most representative examples is IBM's chess program (Deeper blue). It defeated the champion of the world chess (Gary Kasparov) in 1997 [29], which gives impression that the artificial intelligence system can even outperform a human expert. Expert systems were the first commercial systems ranging from manufacturing, process monitoring to financial decision making, etc. An expert system consists of two components: knowledge base and inference engine. Knowledge base denotes the knowledge of the world, which can be represented by a set of rules. Inference engine use the knowledge to make decision. The rules in expert systems are absolute. Unlike expert systems, fuzzy system first introduced by Zadeh [30] is used to to describe element in a range rather than give a absolute value. This is common in our daily life. For example, an old saying in stock market is: "buy low, sell high". However, what can be considered as low or high depends on the stock curves in a particular situation. There is no absolute value. Fuzzy systems have many commercial applications in why do you not mention IL in the text anywhere?

5 Inferring Individual Behaviors Through Interactive Turing Learning

it can be difficult to learn ..... through pass. ob.

5.1 Introduction

In the previous chapters, we demonstrate that the metric-free approach can learn swarm behaviors only through observation. This is based on an implicit assumption that the behavioral repertoire of agents in the swarm could be fully revealed. That is, from the perspective of system identification, the target system has high observability. However, when the target system has low observability, ~~only through passive observation may not learn~~ all the behavioral repertoire of agents. For example, if we try to learn the behavior of an agent's response to certain stimuli, only randomly changing the stimuli (input) may not fully ~~extract~~ all the behavioral information of the agent, as some behavioral repertoire is hidden. Instead, the system needs to interact with the agent ~~in an active way~~ to explore the rare information from the agent. Based on this idea, we aim to learn ~~such~~ agent behaviors ~~which~~ have low observability.

wrong meaning otherwise!

about

reveal

may be

that

inter-action is always active ?

Observation and interaction are widely adapted by scientists when investigating the behavior of animals. For example, when investigating animals' behavior in their natural habitat, passive observation is a preferable method as it is difficult to change the environmental stimuli. In this case, inferring the causal relations between the animal's behavior and its environmental stimuli may become challenging, because these stimuli are not under the observer's control. However, when the experiments are carried out in a controlled environment, it is possible to actively change the stimuli to interact with the animals under investigation in a meaningful way [167, 168, 169]. In [169], in order to investigate cause of the dung beetle dance for orientation, biologists designed various experiments such as the appearance of disturbance or obstacles to interact with the dung beetle and learn how it adapts to the environmental changes.

the?

not clear

these are not experiments! do you mean, testing the ...?

needed!

ions

In this chapter, we investigate whether a machine could learn such behaviors (~~demonstrated using computer simulation~~) through actively interacting with the agents under investigation, without the need of human intervention. We validate our approach using two case studies: deterministic agent behaviors (see Section 5.3.1) and stochastic agent behaviors (see Section 5.4.1). The machine outputs a model of the agent that captures its behavior in relation to the environmental stimuli.

say this in a separate sentence!

We make the following assumptions:

in particular, we

- The machine can observe the agent's actions. ~~In this work~~, we consider the agent's motion, and therefore assume that the machine can observe the agent's position as this changes over time.

in particular, we assume

- The machine is capable of simulating the actions of the agent. ~~In this work, this corresponds to~~ generating sequences of coordinates over a time interval.

it can e

- The machine is able to control the agent's environmental conditions, which in this work corresponds to the intensity of the ambient light.

is this true? not speed??

~~The advantages of our approach are twofold:~~

- Firstly, it does not rely on a pre-defined metric for gauging the resemblance of models to the agent. Rather, such metrics are implicitly defined by the classifiers, and hence incorporated into the evolutionary process.
- Secondly, the machine learns the agent's behavior by interacting with it, rather than simply observing its behavior in a passive manner. This interaction can help the machine to extract all of the agent's behavioral dynamics, as will be shown in the results section.

This chapter is organized as follows. Section 5.2 describes the methodology used, including the agent behaviors (deterministic and stochastic) and the implementation of the metric-free method. Section 5.3.3 presents the results of learning the deterministic agent behaviors, including analyses of the evolved models and classifiers, the coevolutionary fitness dynamics, the effect of noise on the algorithm's behavior. It also presents a comparison of the coevolutionary approach with a single-population evolutionary approach and a approach based on coevolution of tests (inputs) and models. Section 5.4.3 and Section 5.4.4 presents the results of learning stochastic behaviors. Section 5.5 summaries the chapter.

5.3? 5.4? missing!

metric-based???

?

We extend Turing learning such that

is it Turing L. or something else?

5.2 Methodology

This section describes the metric-free method used in this chapter to illustrate the ideas presented in the introduction. The coevolutionary algorithm is comprised of two populations: one of models, and one of classifiers, which coevolve with each other competitively. The fitness of the classifiers depends solely on their ability to distinguish the behavior of the models from the behavior of the agent, described in Section ??. The fitness of the models depends solely on their ability to mislead the classifiers into making the wrong judgment, that is, classifying them as the agent.

rephrase

This is Turing Learning. Just explain(ing) the differences!

5.2.1 Model

The models are represented by a set of parameters govern the rules of the agents. The details of these parameters will be described in Section 5.3.1 and Section 5.4.1 As we have argued in the previous chapters, explicit representation (i.e., evolving only the parameters) makes it feasible for us to gauge the quality of the models obtained. In principle, the method will still work using implicit representation such as artificial neural networks.

an

classifier

now has an additional input, same as before!

5.2.2 Classifier

The structure of ~~the classifiers~~ is similar to the one used in Chapter 3 (see Fig. 3.1). ~~The only difference is~~ the neural network takes the environmental stimuli and ~~the agent's response~~ as inputs and the outputs can be used to control the environmental stimuli and ~~make decision~~. For the proof-of-concept study, we only use one environmental stimulus and stimulate the agent behaviors in one-dimensional environment. Fig. 5.1 shows the structure of the classifiers ~~used in this chapter~~.

can

Suppose the agent responses to the level of light intensity in the environment, ~~and~~ we assume that the system could observe the agent's speed corresponding to the light intensity. ~~One of the inputs to the network (classifier)~~ is the light intensity in the environment at time step t, $I^{(t)} \in [0, 1]$, and the other input is the speed $v^{(t)}$. In order to make this setup more feasible to implement, it is assumed that the system cannot directly measure the speed of the individual, but rather its position. The speed of the individual

rate

of the agent

This is not the right place to talk about the agent 1-D movement! 5.3.1?

has 2 inputs. One represents

However,

Moreover, the environmental stimulus is also fed into the classifier neural network as an additional input.

has an additional output that is used to control an environmental stimulus.

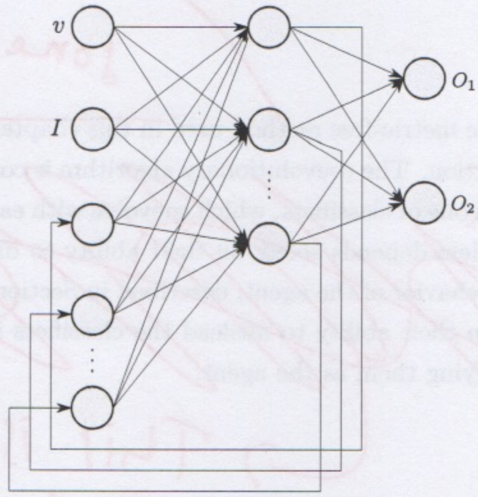


Figure 5.1: This diagram shows the structure of the classifiers used in this chapter. It is a recurrent Elman neural network [151] with two inputs (agent speed, v and environmental stimulus, I), three hidden neurons, and two output neurons (O_1 and O_2). O_1, which controls the stimulus, is fed back into the input; O_2 is used for making a judgment. Two bias neurons with a constant input of 1.0 are connected to each neuron of the hidden and output layers. See the text for details.

for the classifier's input is then calculated by subtracting the previous estimated position from the current estimated position, and dividing the resulting number by the time interval between two measurements.

In order to make a judgment between a model and the agent, the classifier observes the behavior (speed) over a period of time. In addition, the classifier is also in control of the light intensity in the individual's environment. At time $t = 0$, the value of the light intensity is chosen randomly with a uniform distribution in the range $[0, 1]$. The neural network is then updated, using $I^{(0)}$ and $s^{(0)}$. The value of the light intensity for the next time step is obtained from the classifier's output neuron O_1, and the process repeats. After having iterated through all the time steps (a single trial), the final value of output neuron O_2 is used to make a judgment: the network decides on a model if $O_2 < 0.5$, and on the agent if $O_2 \geq 0.5$. The memory (value of hidden neurons) of the classifiers is reset at the end of every trial.

5.2.3 Optimization

The algorithm used here is based on a $(\mu + \lambda)$ evolution strategy with self-adaptive mutation strengths [170, 153]. It is the same as the one used in Chapter 3. For the details of the implementation, see Section 3.1.1.3 of Chapter 3.

5.2.4 Fitness Calculation

Suppose the population sizes for the model and classifier are N and M, respectively. The fitness of each model is obtained by evaluating it with each of the classifiers in the competing population (N in total). For every classifier that wrongly judges the model as being the agent, the model's fitness increases by $\frac{1}{N}$. The final fitness is in $[0, 1]$.

The fitness of each classifier is obtained by using it to evaluate (i) each model in the competing population (M in total) once, and (ii) the agent L times with different initial light intensities. For each correct judgment of the model and the agent, the classifier's fitness increases by $\frac{1}{2 \cdot M}$ and $\frac{1}{2 \cdot L}$, respectively. The final fitness is in $[0, 1]$.

5.3 Case Study One

one with

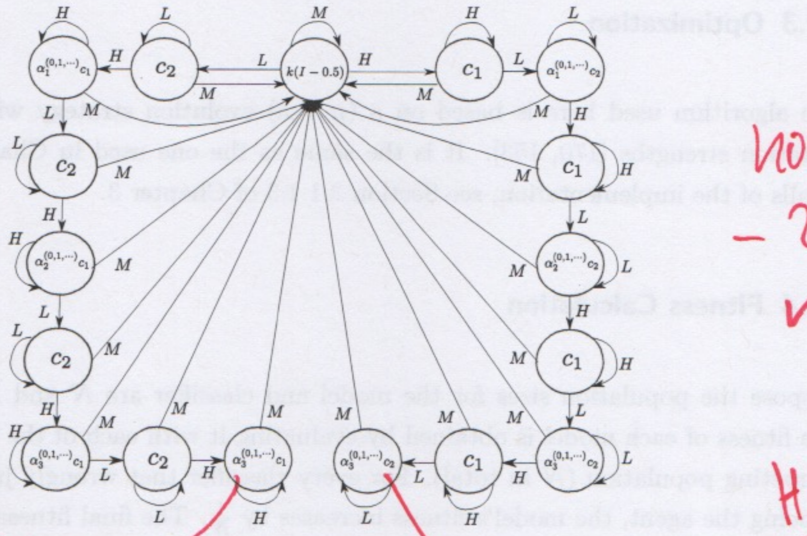
To validate our method, we present two case studies: deterministic behaviors and stochastic behaviors. The behaviors to be identified were chosen to serve as a ~~tractable test-bed for~~ proof-of-concept ~~study~~. While it may loosely correspond to how some 'real' agents react to the stimuli in their environment, it is not intended to mimic any specific animal. In these behaviors, non-trivial interaction with the agent is critical for leading the agent to reveal all of its behavioral repertoire. one with

44

5.3.1 Deterministic Behaviors

We simulate a one-dimensional[1] environment in continuous space. The simulation advances in discrete time steps $t \in \{0, 1, 2, \dots\}$. The (ambient) light intensity in the envi-

[1]In principle, the system will still work in higher dimension. As the main focus is to show how a machine interacts with the agent, for the sake of simplicity, we chose one-dimension.



not complete – 2 transitions missing!

?

L

H

?

Figure 5.2: The deterministic agent behavior under investigation. It shows how the agent responses to the level of light intensity (L, M and H) in its environment. Each state represents the agent's speed. See texts for details.

Table 5.1: This table shows the change of the agent's speed (shown in Fig. 5.2), for an example sequence of light levels.

level	M	H	L	H	L	L	L	H	H	L	L
speed	$k(I-0.5)$	c_1	c_2	c_1	$\alpha_1^0 c_2$	$\alpha_1^1 c_2$	$\alpha_1^2 c_2$	c_1	$\alpha_1^1 c_1$	c_2	$\alpha_2^1 c_2$

L	H	H	L	L	H	H	H	M	H	L	L
$\alpha_2^2 c_2$	c_1	$\alpha_2^1 c_1$	c_2	$\alpha_3^1 c_2$	c_1	$\alpha_3^1 c_1$	$\alpha_3^2 c_1$	$k(I-0.5)$	c_1	c_2	$\alpha_1^1 c_2$

ronment, I, can be varied continuously between 0 and 1 (see Fig. 5.2). The agent distinguishes between three levels of light intensity, low ($0 \leq I < I_L$), medium ($I_L \leq I \leq I_H$), and high ($I_H < I \leq 1$). Hereafter, these levels will be referred to as L, M, and H.

If the light intensity is at level M at time t, the speed of the agent, $s^{(t)} \in \mathbb{R}$, varies linearly with $I^{(t)}$ as:

$$s^{(t)} = k\left(I^{(t)} - 0.5\right), \tag{5.1}$$

where k is a constant.

The agent's behaviors for levels L and H depend on the previous levels of light intensity

this shows the behavior of the agent, but you say ~~that~~ that I can be varied from 0 to 1. You need to refer to Fig 5.2 separately – makes no sense otherwise.