THE REFLEXIVE INSTRUCTOR WITH DELIBERATE APPRENTICE ARCHITECTURE

by

Alexander Ferworn

A thesis

presented to the University of Waterloo

in fulfilment of the

thesis requirement for the degree of

Doctor of Philosophy

in

System Design Engineering

Waterloo, Ontario, Canada, 1997

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

ABSTRACT

The Reflexive Instructor with Deliberate Apprentice Architecture

A framework allowing a discourse in autonomy applied to autonomous mobile robots is developed based on human autonomy. This framework is extended to mobile robotics and is used to evaluate the level of autonomy in a novel approach for constructing autonomous controllers called the Reflexive Instructor (RI) with Deliberate Apprentice (DA) architecture. We claim that the RI/DA architecture supports the construction of first-order autonomous learning agents restricted only by their ability to interact with their environments.

The architecture uses simple reinforcement signals provided by the RI component to train the DA. The DA is responsible for providing control signals to the agent's actuators based on received sensor input. Like most reinforcement learning systems it is not likely to do this very well until it has learned to avoid collisions and obstacles in its environment. The RI provides a measure of safety in this respect as it is responsible for taking over control of the agent if the DA makes a mistake as well as providing an appropriate signal to the DA so it might learn from the mistake.

The RI/DA interaction is advantageous because it protects the vehicle from its own ignorance and helps accelerate learning in the DA.

ACKNOWLEDGEMENTS

I would like to acknowledge the patient insistence, loving care and effective support provided to me by my wife Charmaine and the benevolent, no-nonsense direction of my supervisor and friend Deb Stacey. I also acknowledge the inspiration provided by my cat Dave.

DEDICATION

This work is dedicated to my wife Charmaine--without her there would be no work, and to Charlotte, my daughter.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

1. Introduction

Rene Descarte was relaxing in a bar one day.
When an old friend came in and offered to buy him a drink.
Rene said "I think not".
And disappeared.

Anon.

1.1 Introduction

This chapter introduces many of the concepts that will be discussed and illustrated in later sections of this work. It presents a framework for the thesis and serves to introduce many of the motivations behind this research.

1.2 Reflexes

Descartes, is often cited as the person responsible for the first descriptive statement of involuntary action which we have come to know as a reflex. He makes reference to it explicitly in the "Passion of the Soul" and it is implicit in his theory of the automatism of brutes. Speaking of the escape response...

> For in certain persons that previous associations disposes the brain in such a way that the spirits reflected from the image thus formed in the gland proceed thence to take their places partly in the nerves which serve to turn the back and dispose the legs for flight ...[1]

The basic structural element within any nervous system is the so-called reflex arc. This consists of a series of impulse-transmitting neurons connecting a sense organ to a control centre such as a brain or spinal cord and then to an actuator such as a muscle or gland.

[1] Cited by [Fearing 30]

Each of the actions elicited by a reflex response serves the creature exhibiting it in some well defined way. In the majority of instances, they are actions that are essential for the protection and continued survival of the animal. Take for example our own response to unexpected heat. We withdraw our hand when it accidentally touches a hot surface. The response is involuntary and entirely appropriate as the action saves us from a painful burn.

Many invertebrates such as earthworms, crayfish and roaches exhibit an escape response when threatened. This is elicited by a sensory stimulus such as seeing an object move too close or touching something unfamiliar, and initiates an entirely involuntary sudden synchronous contraction of groups of muscles that move the animal away from the perceived threat--potentially saving its life.

In each case, the reflex action is involuntary, sudden and in some cases, the animal may not even be consciously aware of its response. In addition, each reflex, on its own, is highly predictable and specific. Each stimulus elicits a specific response which can be relied upon for regularity and continuity.

Our knees, for example, involuntarily jerk when an appropriate area is stimulated with sufficient force. A receptor sends a signal via a sensory neuron directly to a motor neuron attached to a muscle which causes our knee to jerk. This will happen each time this area is stimulated as the entire apparatus is, in some sense, "hard-wired".

Reflexes serve various purposes. Food on our tongues stimulates the release of saliva from the saliva gland. Fear triggers the adrenal gland to send adrenaline into our blood stream. Blinking is a reflex. We, and every other animal having some form of nervous system, are born into the world a jumble of reflexes activated by particular stimuli. We have seen that we use reflexes to survive but, perhaps more subtly, we use reflexes to learn.

Consider the automatic sitting reflex. If pressure is placed on a new-born's thighs and its head is flexed, the child will display the automatic sitting reflex as it rights itself. The child has no way of knowing how to sit yet it ends up sitting. The automatic sitting reflex persists for six to eight weeks. This is about the same length of time a child requires to learn to right itself by grasping stationary objects around it. When an object is placed in a baby's hand, the child will flex its fingers and maintain a strong grip on the object. The grasping reflex lasts from six to eight months during which time the child learns to locate, target and hold on to objects of it own accord. In these cases, the child's reflexes serve as valuable training tools. They provide consistent responses which address certain stimuli and aid in the progressive orderly sequence of normal motor development; from the apedal, to the quadrupedal, to the bipedal level of maturation.

As the child matures, these reflex responses disappear and are replaced with deliberate action involving the voluntary participation of the child. They learn to do things deliberately. Let us return, for the moment, to our reaction to pain from an open flame. When we are young, we learn from this reflex, we begin to associate "hot" with pain. We determine that we do not like pain and that we should avoid performing activities which result in it. We deliberately avoid touching hot objects. We choose not to touch hot objects.

1.3 Mobile Robotics

Autonomous mobile robotics researchers were quick to pick up on the benefits of reflex actions if only to support a more plodding planning scheme which could not react to changing situation quickly. Robots were developed with built-in reflexes and called reactive systems.

Certain actions are performed by a robot's actuators when a certain stimulus or stimuli are detected by the robot's sensors. The robot responses are preplanned to

be appropriate in a given situation. This mode of construction was first suggested by [Braitenberg 86] and systematized by [Brooks 86] in his subsumption architecture utilizing augmented finite state machines (AFSMs) to create functional units of behaviour. The intent of subsumption was to provide a framework for designing the interactions of various behaviors in order that the entire robot would display emergent behavior from the interaction of its sub-systems.

The development of subsumption was a great leap forward in terms of describing a systematic means of creating systems which could perform useful work in real time in an unstructured environment. Unfortunately, the architecture does not lend itself very well to learning new tasks or to providing a systematic means for designing subsumption control systems to accomplish those tasks. While AFSMs provide a good description mechanism, they essentially support only "hard-coded" behavior. This, in many cases is inadequate for adaptation to changing environments. In addition, the complexity of the sub-system interactions make the creation of operational systems problematic.

1.4 Reinforcement Learning

It is possible to build a robot to sense nearly anything. Unfortunately, sensing alone is insufficient. As raw data arrives it must be processed into a form which can be utilized. Traditional learning algorithms require input in a relatively rich form to interact with an internal world model. This is again problematic as the time to process the input data often mitigates the robot's ability to cope with the consequences of it.

The standard reinforcement learning model provides a much simpler mechanism. An agent (the robot) is connected to its environment via sensors which produce reinforcement signals indicating the utility of the action performed by the robot which elicited the signal in the first place. Over time, the learning

mechanism is rewarded for actions it selects which produce positive reinforcement signals and punished for negative signals.

Reinforcement learning offers a tantalizing solution to the problem of processing complex, information-rich signals, as a sparse reinforcement signal can be extracted from much simpler sensors in real time. Intuitively this is quite easy to understand as a video image of an approaching obstacle must be converted and interpreted into some internal representation while the simple on or off signal of a contact switch can give immediately useful reinforcement.

1.5 Synergy

There have been various mobile robot architectures suggested which have made use of one or several of these concepts. However, taken together, these areas form the basis of a powerful new technique for interacting with the environment and learning from it in real time in an elegant manner.

1.6 Autonomy

A crucial aspect which is sadly lacking in most discussions concerning autonomous mobile robotics is what one means by autonomy.

Consider the following abstracts;

- A distributed, heterogeneous network of fuzzy control agents has been developed for reactive behavior-based control of an autonomous mobile robot...The aim of our mobile robot project is to develop a complete autonomous creature, with 100% on-board computation and a behavioral repertoire capable of accomplishing non-trivial tasks. [Goodridge, et al. 94]

- This paper describes an autonomous dump trucks system called Hazama Intelligent Vehicle Automatic Control System (HIVACS) being developed to overcome worker shortage problems and to prevent accidents in heavy construction sites (i.e. dam construction, road construction, etc.). If this system applies at the aggregate carriage in RCD (Roller Compacted Dam) construction site, this system enables two operators to manage five dump trucks without drivers. It is anticipated that this system enables for 17% of labor-saving in total workers at a certain dam construction site. Algorithm of unmanned operation control is based "Yamabico" (sic) robot that is currently under study at University of Tsukuba. A test vehicle for autonomous dump truck is developed in this study. [Saito, et al. 95]

The first abstract suggests that the paper will describe an independent robot vehicle with characteristics similar to living creatures while the second abstract seems to associate autonomy with some form of loose tele-operation.

Of the thirty nine papers published in the IEEE International Conference on Robotics and Automation between the years 1988 and 1995 specifically mentioning autonomy in their titles, not a single one ventures to define what they mean by "autonomous".

This is not particularly surprising since the concept of autonomy has provided perplexing, varied and often heated discussions in fields as diverse as psychology, philosophy and biology for centuries. While there is certainly no agreement on how autonomy should be defined there is a rich body of work which can be used to develop tests for what can be called autonomous behaviours. The tests developed using the concepts of autonomy from other fields will be applied to the architecture presented in this thesis.

1.7 Proposition

This document presents a description, implementation details and evaluation of a novel autonomous agent control architecture known as the Reflexive Instructor with Deliberate Apprentice (RIDA). The architecture supports the ability of an agent to exhibit aspects of autonomy related to mobility which improves its survivability by learning from experience over time.

The architecture makes use of biologically inspired reflex reactions designed to respond appropriately to a given situation while at the same time providing necessary training signals to a learning system. Each control sub-system is essentially independent of any other yet communicates with other modules in a strictly hierarchical manner.

In this dissertation;

1. We develop a framework for autonomous mobile robots based on the human concept of autonomy. This framework is then used to validate the RIDA architecture in terms of how the architecture supports the framework.

2. We demonstrate how this architecture supports the training of a learning system while at the same time allowing real time interaction with the environment while training.

3. We show that the architecture supports the graceful degradation of performance as control subsystems are removed or become damaged and how the architecture supports learning in novel situations.

4. We demonstrate how the architecture can be expanded and scaled within its stated domain to support additional control modules without substantially changing the existing modules.

5. A simple control module interaction scheme is developed and demonstrated.

1.8 Organization of this Thesis

Chapter two of this document will revisit the areas of reflex, mobile robotics and reinforcement learning with an examination of salient terminology and implementations. The concept of autonomy will be examined from a multi-disciplinary perspective with the goal of applying the meaning of this concept from other disciplines to this work.

Chapter three will introduce and discuss the RIDA architecture, its sub-systems and features. Chapter four will revisit the topic of learning and how it promotes adaptation. In addition, we revisit the theoretic contributions of this work. Chapter five will show how a reliable RI was selected and present evidence that reflexes can be used to improve the learning performance of simple vehicles with arbitrary intelligent controllers. Chapter six will introduce the framework for autonomy within which RIDA will be examined and tested. Chapter seven will discuss several RIDA implementations and examine their performance with reference to the autonomy framework developed in chapter five. Finally, chapter eight will discuss certain potential non-traditional applications for the architecture and will draw conclusions from this work.

A set of appendices have been included which contain technical data and specifications for the various RIDA implementations.

2. Terminology, Theory and Literature examination

Philosophy recovers itself when it ceases to be a device for dealing with the problems of philosophers and becomes a method, cultivated by philosophers, for dealing with the problems of men.

John Dewey,
"The Need for a Recovery in Philosophy", 1917

2.1 Introduction

This chapter is intended to provide the reader with appropriate background information concerning relevant research applicable to this work. It introduces various terminology and notation which will be used throughout the remainder of the work. Specifically, the chapter will address issues in;

- Mobile Robotics, including the introduction of various reactive and other systems devised to address the issue of independent mobility,
- Biological reflexes, including a brief theoretical foundation providing various examples relevant to biological and mechatronic systems,
- Learning, what it means and how it happens in a biological context,
- Reinforcement learning,
- Neural learning systems, including the theoretical development of the various learning mechanisms addressed in this work such are reinforcement learning,
- Rapid Reinforcement Learning Using Neural Networks, and
- Autonomous systems from a multi-disciplinary approach. This section will introduce appropriate terminology and concepts from philosophy, psychology and biology which will later (chapter 5) be applied to robotics.

2.2 *Mobile Robotics*

The dream of constructing an artificial device capable of independent motion and control is not a new one. Through time, many investigators, entrepreneurs and schemers have attempted to devise systems which would fend for themselves.

As a classic robotic system, autonomous vehicles were simply a class of robot formed from a mechanical system designed to perform a specific function (such as an end-effector or locomotor) but also intended to move through space independently via the use of an on-board (or close to on-board) controller. Sub-classes of this type of device have been designed to address land, liquid and aerial mobility.

Mobile robotics research has essentially fallen into two camps. Those who adhere to the notion that it is necessary to implement a very hierarchical control structure based on simplified models of the world and those who believe that a distributed controller--where sensing is closely followed by action-- is more appropriate.

2.2.1 Hierarchical Model Builders

Generally, a hierarchical controller makes decisions based on an internal model of the environment the controller finds itself in. The model is constructed by abstracting from direct sensory input. Features are extracted from what is sensed in hopes of simplifying, what is very often, a large data stream into much simpler symbols.

Once constructed, plans are made based on the model by yet another sub-component of the controller that is able to reason using these symbols. Because the model is now much simpler, reasoning can be carried out at a high level and broad directives issued like "move to the wall". This form of reasoning was heavily influenced by [Simon 69] and his symbol system hypothesis that suggests that intelligence operates by manipulating symbols of entities in the real world.

If we believe this, then it is possible to remove the reasoning process from the actual environment in which the actions must happen and endow it with domain independence. This is significant as the reasoning can take place without reference to context, the controller is more likely to succeed in an unpredictable environment where context is constantly changing.

The broad directives issued from the reasoning system must be converted into actual commands--"move forward, turn right, etc." and these, in turn, are converted into extremely low level control signals which manipulate the vehicle's actuators.

The first example of a vehicle employing this type of control was "Shakey", developed at the Stanford Research Institute [Hart, et al. 68]. Shakey employed a television camera and a touch sensor as inputs to its controller.

The on-board processor was connected to a much larger off-board computer through a radio link. Vision and planning were computed off-board while actual motor controls were generated internally. Relatively complex models were constructed and movement decisions made and implemented based on these models. Shakey's world representation consisted of sets of well-formed formulas of predicate calculus. Simple English commands could be entered which would then be converted into formulas for resolution through a generated plan of action.

Shakey's position within its environment was determined by keeping track of various markers in conjunction with measuring motion based on wheel rotation. Slippage often caused it to miscalculate its position. This often caused its modeling system to miscalculate the placement of objects detected through its vision system thus resulting in an incorrect model and eventual plan failure.

Shakey's ability to control itself through a series of layers of software constituting a hierarchical control structure heavily influenced many who would build such vehicles in the future.

Between 1973 and 1981, research was conducted at the Stanford University Artificial Intelligence Laboratory [Moravec 81] on the construction of an off-board computer controlled, camera equipped, mobile robot. This became known as the "Stanford Cart".

The Cart used stereo imaging to locate objects and to determine its next movement. Nine different views were taken during the model building/decision making process. The system was reliable for short movements, but reliability came at a heavy cost--speed. Motion occurred in lurches of one meter every ten to fifteen minutes.

Of most recent interest was the highly-publicized Dante II experiment [Kaspar 94]. Jointly conducted between NASA and Carnegie-Mellon University, this experiment in tele-operated and autonomous walking systems attempted to show the possibility of allowing at least partial autonomous hierarchical control in the highly unstructured environment of the volcano Mt. Spurr in Alaska.

Dante II was provided with a wide range of vision systems including one for each if its eight legs, a laser range finder and several cameras mounted on helicopters hovering overhead. When Dante II was allowed to move in "autonomous mode" it used the input from these sources to build a model on which it based decisions on leg position.

Dante II moved more than two hundred feet over several days before losing its footing and crashing to the crater floor. Despite its ultimate destruction, Dante II was considered a limited success because of the amount of data collected and experience that was accumulated through the trial. In addition, Dante II confirmed

that hierarchical control could be made to work in real world environments. For further discussion of this area see [Antsaklis 89].

2.2.2 Distributed Reactive Systems

The second camp, of more recent origin, consists of those who argue that action must happen shortly after sensing. If this does not occur, much of the useful information in the sensing data will change and acting on it will result in inappropriate or even dangerous behaviour by the robot. Reactive controllers take action based directly on what is sensed in hopes of reacting quickly enough to address the situation. This speed is fostered by relatively simple control mechanisms which are closely tied to both sensors and actuators.

Undoubtedly, the most influential contribution to reactive systems came from [Brooks 86]. His controversial subsumption architecture has both inspired many and infuriated others. The next section will provide an overview of this architecture.

2.2.2.1 *Subsumption*

The subsumption (or "Brooksian") architecture is modeled after the close interaction between sensing and actuation in lower animals like the cockroach. Brooks argues that instead of building complex agents in simple worlds, we should follow an evolutionary-inspired path and construct simple agents in the real, complex and unpredictable world.

From this argument, a number of key features of subsumption result:

- No explicit knowledge representation. Brooks often refers to this as "The world is its own best model".
- Behaviour is distributed rather than centralized.

- Response to stimuli is reflexive -- the perception-action sequence is not delayed by deliberation on the part of a higher controller.
- The agents are organized from the bottom-up. Thus, complex behaviors are created from the combination of simpler, underlying ones.
- Individual agents are inexpensive, allowing a domain to be populated by many simple agents rather than a few complex ones. These simple agents individually consume few resources (such as power) and are expendable, making the investment in each agent minimal.

2.2.2.1.1 Subsumption Architecture Description

The Subsumption architecture is constructed in layers. Each layer provides the system a set of pre-wired behaviours. The higher levels build upon the lower levels to create more complex behaviors. The behavior of the system as a whole is the result of many interacting simple behaviors.

Each layer of the Subsumption architecture is composed of networks of finite state machines (FSM) augmented with timers. A FSM is a device composed of a set of states, a finite set of signals it understands (tokens) and a transition function to map received tokens to acceptable states [Hopcroft and Ullman 79]. The timers enable state changes after preprogrammed periods of time. In effect, they provide a degree of fault tolerance when expected signals/tokens are not received.

Each Augmented (A) FSM has an input and output signal. When the input of an AFSM exceeds a predetermined threshold, the behavior of that AFSM is activated (i.e. the output is activated). The inputs of AFSMs come from sensors or other AFSMs. The outputs of an AFSM are sent to the agent's actuators or to the inputs of other AFSMs.

Each AFSM also accepts a suppression signal and an inhibition signal. A suppression signal overrides the normal input signal. An inhibition signal causes

output to be completely inhibited. These signals allow behaviors to override each other so that the system can produce coherent action when contradictory control information is applied .

The use of AFSMs results in a tight coupling of perception and action, producing the highly reactive response characteristic of subsumption systems. However, all patterns of behavior in these systems are pre-wired.

2.2.2.2 *Hierarchical and Distributed Control and the Problem with Change*

The problem with both these methods of robot construction is that they are brittle to the problem of mobility--but in different ways. While hierarchical controllers support the notion of learning in a changing environment, they typically have no way of doing so quickly. Their preponderance of modeling, reasoning and translation subsystems makes their ability to respond to a changing environment quite slow.

Subsumption systems, while avoiding the problem of speed, cannot adapt to an environment which they were not designed to interact with. Some work has been done with a Planning and Learning Extension to the basic subsumption architecture [Mataric 92]. These extensions are known as behavior-based architectures. Mataric describes some modifications to the subsumption architecture to allow it to recognize by "remembering" features previously encountered. While the basic architecture is extended, the interactions tend to be difficult to design and prone to many of the faults attributed to more traditional hierarchical architectures.

In addition, while the architecture is quite clean, the interaction of the AFSMs is usually very complex making the creation of a viable controller in a commercial product impractical. Such controllers typically require many hours of painstaking "tweaking" to get right.

2.2.2.3 *Other Reactive Approaches*

In reality, many problems related to mobility can be solved using much simpler mechanisms than subsumption. Examples of simple reactive systems which avoid some of the problems of subsumption are not difficult to find. [Zapata, et al 94] suggests several vehicles capable of collision avoidance using what he calls deformable virtual zones (DVZ) and simple neural networks. Although learning is not supported, the vehicles are relatively straightforward to construct.

[Nehmzow, et al. 89, 92] suggest methods for extending the behavioural repertoire of a mobile robot through the selection of "instinct rules" to an existing controller. The neural network-based controller learns to apply the available instinct rules to a given situation through its interaction with the environment, although the rules themselves are supplied by the designer.

[Ram et al 94] and [Dorigo and Schnepf 93] have suggested using genetic algorithms as a means for a controller to learn reactive control policies in a given environment. This method has proven successful but does not address how the controller is to be supported while it is learning.

2.3 *Reflexes*

In animals, reflexes are involuntary acts that represent the lowest level of nervous response to a stimuli and underlie all animal behavior. The intensity and pattern of stimuli largely shape the strength and type of the reflex that is elicited. Increasing intensity and frequency of impulses to a nerve centre will reach a threshold, at which point the response is triggered. Often sensory input is used in a distributed fashion for various responses at different levels where a reflex depends on the timing of an incoming signal or perhaps other elements which modify its activation or perhaps eliminate it altogether.

The most familiar example of a reflex response is the monosynaptic stretch reflex or "knee jerk". The sensory receptors for this reflex arc are located in muscle spindles embedded in skeletal muscles. When the kneecap tendon is tapped the muscle is stretched and the muscle spindles in the thigh muscle are excited. Because this reflex response involves sensory neurons that directly connect via synapses with motor neurons in the spinal cord, transmission time is short, and the thigh muscle quickly contracts, extending the leg.

Important reflex nerve centres related to posture, balance, and eye position are located in the brainstem. Receptors for these reflexes are found in the vestibule of the inner ear. These reflexes serve two functions: to stabilize the position of the head and provide information about its angular and linear acceleration, and to maintain visual image by stabilizing the eyes during head movement. Individuals suffering from motion sickness usually experience disturbances of these vestibular receptors. [Fisher 88]

2.4 Learning

Learning is supported by reflexes. Of particular relevance is the work of [James 1890] who proposed that consciousness conferred on its possessor the ability to move beyond the confines of mere instinct and respond in a more flexible way to novel situations. Learning was simply the way that animals, including humans, adapted to their changing environments.

A significant consequence of James' and others work was the notion that animal behavior, and especially learning, could be studied in animals. Of particular interest to us is the notion of learning behaviours. The term "conditioning" is used, in psychological parlance, to designate the forms of behavioral learning that humans share with animals.

One way in which forms of learning can be categorized is as either associative or nonassociative. The latter kind of conditioning occurs when an organism's response to a single stimulus changes with repeated experience. Associative learning occurs when an organism learns to associate two or more stimuli, changing its response to one or both stimuli as a result of their being experienced together. At present all learning is considered associative to some degree. For example, if a person lives near an airport, that person will habituate to the noise of aircraft. That is, they will cease to notice the constant roar of jets. However, if that person is then taken to a much quieter area and a jet flies overhead that person will probably notice it. This suggests the person's earlier habituation was caused by their association of the jet sound with their background and indicates that the response of lack of response is plastic and is even reversible.

Associative learning has been the focus of much research. It is usually described in two forms: Pavlovian conditioning, and instrumental or operant conditioning. The first primarily involves modification of innate reflexive behaviour. The second, of more interest to us, involves modification of behavior by reward and punishment.

The basic law of instrumental learning is that when a response is followed by a reward, its probability of recurring in the same circumstances is increased. Rats press a bar more frequently when they receive food afterward. [Leahey 93]

Punishment can also be used; this involves following a behavior with pain to eliminate it. The term "negative reinforcement" refers to applying pain and then removing it when a desired behavior occurs. The effectiveness of such "negative" methods for the learning process in animals is open to debate. Punishment can in fact be effective in suppressing behavior, but only if the punishment is severe, immediate, and inescapable. In any event, punishment does not cause a habit to be unlearned but only to be suppressed--later to return [Martinez 91]. One need only own a cat to understand this perfectly.

2.4.1 Reinforcement Learning

A deficiency implicit in many artificial learning schemes is the dependence on supervision. Input/output pairs are presented to the algorithm creating a "training by example" paradigm of learning.

Unfortunately, it may not be possible to ensure invariant training examples to the control mechanism of an autonomous agent. In fact, it may not be possible to provide richer reinforcement than a very rough error signal. Consider a simple agent wandering through some environment. Perhaps the agent is equipped with only rudimentary sensing capability. What can be discerned from one of its sensors unexpectedly coming in contact with an obstacle? The only data available is that the agent has hit something. It has no means of ascertaining what actions caused it to strike the object. The error signal simply indicates one has been struck.

In the standard reinforcement-learning model, an agent is connected to its environment via perception and action. On each step of interaction the agent receives as input-- (i), some indication of the current state-- (s) of the environment; the agent then chooses an action-- (a) to generate as output.

The action changes the state of the environment, and the value of this state transition is communicated to the agent through a scalar reinforcement signal--(r). The agent's behavior--(B) should choose actions that tend to increase the long-run sum of values of the reinforcement signal. It can learn to do this over time by systematic trial and error, guided by a wide variety of algorithms

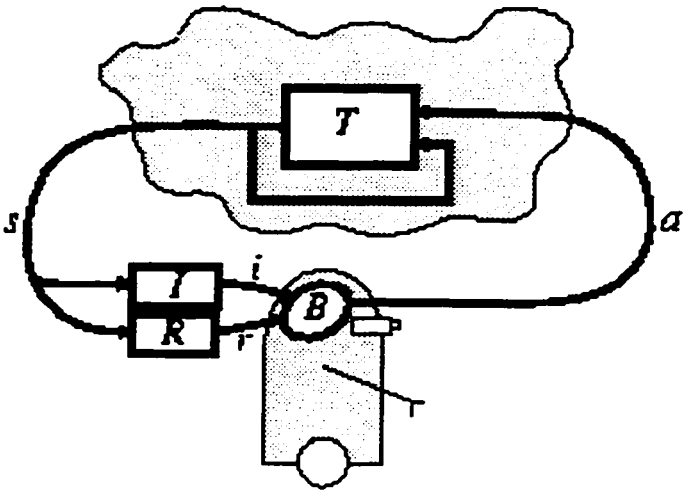


Figure 2-1 Reinforcement Learning [Kaelbling et al. 96]

Reinforcement learning, while very promising, is somewhat problematic. The controller may perform an action but not receive any reinforcement whatsoever – the sparse signal problem. For example, a learning controller might be given control of a vehicle and given the goal of not hitting obstacles. It will have to make many decisions as it moves through its environment, then after acting on many of these decisions, the vehicle might strike a wall. What should the controller learn from this experience? Which of its many actions were responsible for the collision? It is this problem of assigning responsibility--the credit assignment problem--to individual actions that makes successful reinforcement learning an elusive goal.

In [Kaelbling et al. 96]'s words, "Its promise is beguiling--a way of programming agents by reward and punishment without needing to specify how the task is to be achieved. But there are formidable computational obstacles to fulfilling the promise."

2.5 *Artificial Neural Systems*

Artificial Neural Systems (ANS) have misleadingly been referred to as "Brain-like". There is no evidence that biological neurons store information in the form

of weighted connections. For the most part, no similarity at all has been observed in the processes of thought in the human brain and the calculations within an ANS [Hinton 87]. However ANSs are brain-like in that they,

- are adaptive and can be trained,
- are highly parallel processing structures, having large numbers of neuron-like processing elements, and
- exhibit distributed control.

A basic neuron consists of,

- a *soma*, or nerve cell, which is the large central body of the neuron,
- the *axon*, attached to the soma and electrically active, producing the pulse emitted by the neuron (output),
- the *dendrites* which receive inputs from other neurons by means of a contact called a
- *synapse*, which occurs where the dendrites of two nerve cells meet.

The synapse is capable of changing a dendrite's local potential positively or negatively depending on the transmitted pulse. The transmissions occur in large numbers but are very slow, being caused by chemical reactions.

The structure of an ANS consists of layers of processing elements which take a given set of inputs, perform some form of summation and produce an output. There is usually an input layer where information is presented to the network. There may be one or more hidden layers, where this information is manipulated and an output layer where the network output is presented. The nodes of various layers are attached by connections. Most commonly this is done by feed-forward connections, where values move from an input layer to one or more hidden layers and finally to an output layer.

Each connection from one node to another has a weight associated with it. The value of the weight is usually between zero and one and may vary over time. Each node is considered to have a net input which is defined as the sum of all weights of connection multiplied by the activation value or output of the preceding layer. In short, the connection with the greater weight has more impact on the activation of the node being processed. If the activation of a node has a negative value then it will tend to inhibit the activation of the node it is connected to otherwise it will excite it.

When values have passed through a network and have modified activation values and produced an output, it is possible to modify the weights of connections by backward propagating the error associated with each output unit. The error may be defined in many ways but usually has something to do with the amount the actual output was different from the desired output. The change in weight values is termed learning in the network.

2.5.1 ANSs and Robotics

ANSs are not new to robotics and have been applied to such diverse areas as the inverse kinematics problem [Iberall 87], trajectory and path planning [Jorgenson 87], sensing [Pati 88], and control [Elsley 88].

2.5.2 Rapid Reinforcement Learning Using Neural Networks

The traditional problems associated with Reinforcement Learning are,

- sparsity of reinforcement information,
- credit/blame assignment, and
- slow learning rates.

To overcome these difficulty, several novel approaches have been suggested. [Shavlik 96] has suggested the use of human intervention as a means of speeding learning by increasing the availability of reinforcement information by allowing an outsider to input "suggestions" to a reinforcement ANS. This has some advantages but still requires a human operator to be close by to make these suggestions.

[Nehmzow 92] reports of increasing the behavioural repertoire of a mobile agent using a connectionist approach to learn certain "instinct rules". The instinct rules are provided by the system's designer. The selection of instinct rules is determined through interaction with the environment. The vehicle "Alder" was constructed and could learn the interaction of several rules in a few tens of learning steps. Nehmzow points out that fast learning is essential because certain functions like obstacle avoidance must be learned quickly in order for the robot to remain functional. However, the automatic creation of instinct rules remains problematic. It is also unclear whether striking an obstacle could be considered acceptable behaviour, even if it only happens tens of times.

Let us digress for a moment and examine what happens when a human infant learns to avoid obstacles. They often learn only after many encounters--sometimes colliding quite violently with objects. This is quite a different situation than is applicable to a robot. Not only are infants capable of correcting their actions, but they are self-healing--an important advantage that artificial systems do not normally possess.

Ideally, reinforcement learning would take place quickly based on very limited information which could be attached to specific events in time. The literature provides several suggestions for making these, somewhat contradictory, characteristics a reality.

2.6 Autonomy

The term autonomy came into favor in robotic literature about 1986. Any examination of the literature from that time until the present will reveal the fact that there is essentially no agreement about what is meant when one says that a robot is "autonomous" or displays some form of "autonomy".

The following are excerpts of abstracts taken from various papers found over several years in various Proceedings of the IEEE International Conference on Robotics and Automation. Each one of these papers uses the term autonomy or autonomous in its title;

- This paper describes the concept of the autonomous mobile robot system, which comprises a hierarchical autonomous mobile control system, a localization system based on pattern matching between the data profile from a laser range finder and an environment map, a method of constructing environment maps from 3D-CAD of a NPP, and a cableless robot system. [Igarashi, et al. 95]
- For underwater vehicles to be self-sufficient in an a priori unknown environment, reinforcement from the environment through altitude sensors is essential. [Santos, et al. 95]
- This paper presents both [sic] of the hardware and the software architectures for the multi agent robotic system. For the hardware architecture of the multi agent robotic system, we show the programmable MARS (micro autonomous robotic system). This robot can work for one of the agents of the multi agent robotic system. [Mitsumoto, et al. 1995]
- ...The autonomous mobile robot "Yamabico" is used for experiments after [sic] equipped 12 directional sonar-ring. The on-board controller of the robot decided its motion based on sonar-ring data every 3 centimeters going forward. We made many experiments with this autonomous mobile robot, and investigated the validity and the limits of this method. [Ando and Yuta, 1995]

- ...On the other hand, the most critical and challenging issue in designing and programming robotic systems working with some degree of autonomy in dynamic, unstructured environments, is related to the definition of their architecture. [Zanichelli 1994]

The concept of autonomy is used in a bewildering array of contexts and referring to a wide variety of attributes. It is used to refer to the independent completion of specific tasks, or as a characteristic of a system's cooperative effort with other agents. On occasion it seems to refer to a means of describing a system which follows a plan or is used as a phrase to emphasize the characteristics of other elements of a particular robot architecture.

[Meystel 91] provides a working, if somewhat loose, definition of autonomy;

> "autonomy" is understood as the ability to independently make intelligent decisions as the situation changes. Such an ability is possible if intelligence allows a certain level of independence, i.e. if the general goal of motion is formulated by a human-operator but the specifics of the particular motion are taken care of by the robot with no direct human involvement. Thus, we can talk about different degree(s) of autonomy; robots can be generally controlled by a human operator. However some of the operations can be planned, controlled, and executed with no human participation: they are left to the robot's discretion.

However, this is somewhat lax and open to interpretation and does not suggest means of answering the question of "How autonomous is autonomous?"

In our view the lack of common terminology or framework for discussing autonomy has significantly hampered progress in the field if for no other reason than it is impossible to make comparison between any two autonomous systems. We will now consider the question of autonomy from a psychological and

philosophical perspective. Then in a later chapter we will argue that there is strong evidence that it is feasible and necessary that robotic research should adopt the terminology, meanings and concepts used in describing human autonomy.

The notion of human autonomy has been well examined in both psychology and philosophy literature and an entire nomenclature constructed describing various aspects of it. This is not to imply that there has been agreement in either the terminology or definition of the concept;

- "The law in thus implementing its basic commitment to man's autonomy, his freedom to and his freedom from, acknowledge(s) how complex man is" [Goldstein 78]
- "To regard himself as autonomous in the sense I have in mind, a person must see himself as sovereign in deciding what to believe and in weighing competing reasons for action." [Scanlon 72]
- "As Kant argued, moral autonomy is a combination of freedom and responsibility; it is a submission to laws that one has made for oneself. The autonomous man, insofar as he is autonomous, is not subject to the will of another." [Wolff 70]
- "(Children) finally pass to the level of autonomy when they appreciate that rules are alterable, that they can be criticized and should be accepted or rejected on a basis of reciprocity and fairness. The emergence of rational reflection about rules...central to the Kantian conception of autonomy, is the main feature of the final level of moral development. " [Peters 72]
- "I am autonomous if I rule me, and no one else rules I." [Feinberg 71]
- "Human beings are commonly spoken of as autonomous creatures. We have suggested that their autonomy consists in their ability to choose whether to think in a certain way insofar as thinking is acting; in their freedom from obligation within certain spheres of life; and in their moral individuality." [Downie 71]

- "A person is "autonomous" to the degree that what he thinks and does cannot be explained without reference to his own activity of mind." [Dearden 72]
- "[A]cting autonomously is acting from principles that we would consent to as free and equal rational beings." [Rawls 71]
- "I, and I alone, am ultimately responsible for the decisions I make, and am in that sense autonomous." [Lucas 66]

It is apparent that, although not a synonym for qualities that are associated with robotic autonomy, it is used in an equal number of bewildering ways. It is sometimes equivalent to liberty, sometimes it is used to refer to self-rule or sovereignty, sometimes free will. It is equated with dignity, integrity, individuality, independence, responsibility, and self-awareness. It is identified with critical reflection, freedom from obligation, absence of external influence, determination and execution of self interest. It is related to actions, to beliefs, to reasons for acting or not acting, to rules, to the will of other persons, to thoughts, and to principles. About the only features held constant from one interpretation to another is that autonomy is a feature of people, and that autonomy is a desirable quality to have [Dworkin 88].

One definition of autonomy is self-determination. The autonomous person is one who chooses for themselves what to think and what to do. They are self-governing in that their actions are a result of interests and values that they have decided upon. Also, these beliefs are arrived at independently, by means of critical reasoning. The autonomous individual is guided by their own notion of what is right, best, or at least possible. This has been termed the Autonomy of judgment or "thinking for oneself."

In reality we do not directly ascertain the validity of most of our beliefs. A good deal of our autonomy is derived from assessing the behaviour of others--we are

taught. This requires that we have criteria by which to recognize an authority or when someone's testimony is dependable.

With the idea of dependence, we come to the matter of constraints or limitations on our autonomy. External constraints typically interfere with the exercise of autonomy, as with deception or censorship--being lied to or "kept in the dark" can severely limit autonomy. Internal restrictions are due to some condition suffered by the individual rather than outside interference. Typically, they will consist in deficiencies or "defects" in rationality. For example, stubbornness or stupidity might restrict autonomy in this way.

While autonomy of judgment is necessary for autonomous thinking and action, it is not sufficient. Because of either external or internal conditions a person may be incapable of acting or even choosing based on freely made decisions. Threats (external) or the possibility of embarrassment (internal) might restrict an action.

Efficacy of will indicates the ability to do what one wills. Deliberateness of will refers to the extent to which what it is that one wills is the fruit of deliberate choice. Efficacy of will might more colloquially be called autonomy of action, since it refers to our ability to act on our decisions or will.

It is easy to see how autonomy of action can be interfered with. Interference can range from physical limitation to coercion or exploitation. The latter limits the individual by, "attaching costs to certain forms of action that they would not otherwise carry."--For example the association of a certain action with pain (pain = wrong).

[Benn 76] example of the psychopath nicely illustrates how someone could have autonomy of judgment but lack of autonomy of action. Psychopaths cannot carry through projects requiring deferment of gratification. Only immediate consequences of action count as relevant considerations for decision-making.

The nature of one's will is relevant to the question of autonomy. Depending on why one wills what one does, or how the will is formed, it may be more or less autonomous. What one wills may be determined exclusively by the strength of one's desires and impulses. Such a person then acts in accordance with their strongest prompting. If you are both hungry and tired you will eat if the hunger is greater than the fatigue and vice versa. This sort of will consists of the most demanding, urgent force within the individual. Such a will is in some sense less one's own, hence less autonomous, than it might be.

There is a higher level at which our deliberations may proceed. We can assess and make decisions about who we are and what we wish to become. We can take up the question of what sort of person to be and what kind of life to lead. This sort of deliberation goes beyond the ordering of priorities.

"Overall" autonomy is strengthened through flexibility: the ability to respond creatively and constructively to a variety of circumstances; this includes the ability to adapt to change. John Dewey contrasts the development of a chicken's ability to that of a human's.

> The chick, which can peck accurately at food shortly after hatching, quickly develops its expertise in behavior because it stems from only a few original tendencies. Its immediate efficiency, however, is "like a railway ticket,...good for one route only." Whereas, "A being who, in order to use his eyes, ears, hands and legs, has to experiment in making varied combinations of their reactions, achieves a control that is flexible and varied." [Dewey 63]

First-order autonomy is autonomy exercised in the particular decisions which occupy us in the ordinary course of life: where to live, whom to marry, what vocation to pursue...These everyday decisions can be made more or less

autonomously, depending, as we have seen, on our resources, abilities, and freedom from restrictions. [Benn 76]

Autonomy may also be viewed as moral self-governance--the individual authoring his moral principles, obeying moral laws which are self-imposed. The autonomous individual does not simply conform to some conventional standard of conduct. Rather, they rationally ascertain for themselves what is desirable for any rational individual. This is autonomy not in the sense of being governed by contingent desires or ambitions, but governed by the rewards of a dispassionate, disinterested reason. This is what we will refer to as Second-Order Autonomy.

While this is all well and good for humanity there is no necessary link between autonomy of the human and the notion of autonomy we may mean for something as mundane as a mobile robot. If it were possible to associate these two, robotics could draw on a vast array of terminology with specific meaning. Clearly this would be advantageous as it would become possible to compare functionality and characteristics on an "apple for apple, orange for orange" basis. We submit that this has not been the case in the past and would be a useful addition to any discourse on autonomous vehicles.

2.7 Conclusion

This chapter has introduced the concepts and terminology which will form the basis of the arguments we use to build a framework for autonomous mobile robots in which we will place and test the RIDA architecture. In addition we have introduced the algorithms and control concepts on which the RIDA architecture is based. This will be used to develop a test-bed for the architecture in chapter six.

3. The Reflexive Instructor with Deliberate Apprentice Architecture

There ought to be a law so a man knows whether he is doing right or wrong.

Senator Thomas Dodd

3.1 Introduction

This chapter introduces, develops and explains the Reflexive Instructor with Deliberate Apprentice (RIDA) architecture. The function of each element in the architecture is introduced, formally explained and illustrated with appropriate reference to existing techniques. The behaviour of a RIDA system is discussed and an example is used to illustrate RIDA's nature. Several conventional learning mechanisms are used to illustrate how such a scheme might be selected or rejected to act in the RIDA hierarchy.

We argue that RIDA is a flexible and reliable means of controlling a vehicle. Its performance is good, reliable and straight forward in its potential implementation.

3.2 RIDA Architecture Description

RIDA consists of two independent yet related sub-systems,

- the reflexive instructors (RI), and
- the deliberate apprentice (DA).

The sub-systems interact in a way best thought of using a pedagogical analogy. The DA is, in a sense, a student attempting to learn a control task. The RI components are elements which ensure that as the DA learns by making mistakes, it is "guided" by correcting signals to eventually learn the task. The Deliberate Apprentice attempts to send control signals to the actuators it is attached to. We use the term "Deliberate" in the philosophical sense of a

deliberate choice and not in the sense of deliberation or internal examination. The RI(s), in turn, monitor the control signal and either does nothing if the control signal is appropriate or intervenes by over-riding the DA's signal and injecting its own signal which is deems to be more appropriate--much as a teacher might correct the spelling of an errant pupil--the result is correct but produced both through the action of the pupil and the teacher. Continuing the analogy, as the pupil understands they have been corrected, the activated RI informs the DA that its signal was inappropriate and was corrected.

The selection of which RI is activated is accomplished via a strict precedence hierarchy. Individual RIs are activated as the sensors they monitor send the appropriate activating signals. For example, a collision avoiding RI might be activated by a whisker sensor touching an object.

There is no restriction on which RI can be activated or how many may attempt to control the vehicle, however only one control signal is sent to the actuator and only a single reinforcement signal is provided to the DA. This is accomplished by replacing control signals from lower precedence RIs with those of higher precedence signals. Replacement is also carried out with the reinforcement signal.

Figure 3.1 depicts the relationship between modules within RIDA and how they sample sensed data. Individual RIs need not be attached to the same sensors as the DA or to the ones attached to other RIs for that matter. Note that only a single DA exists within the confines of the architecture.

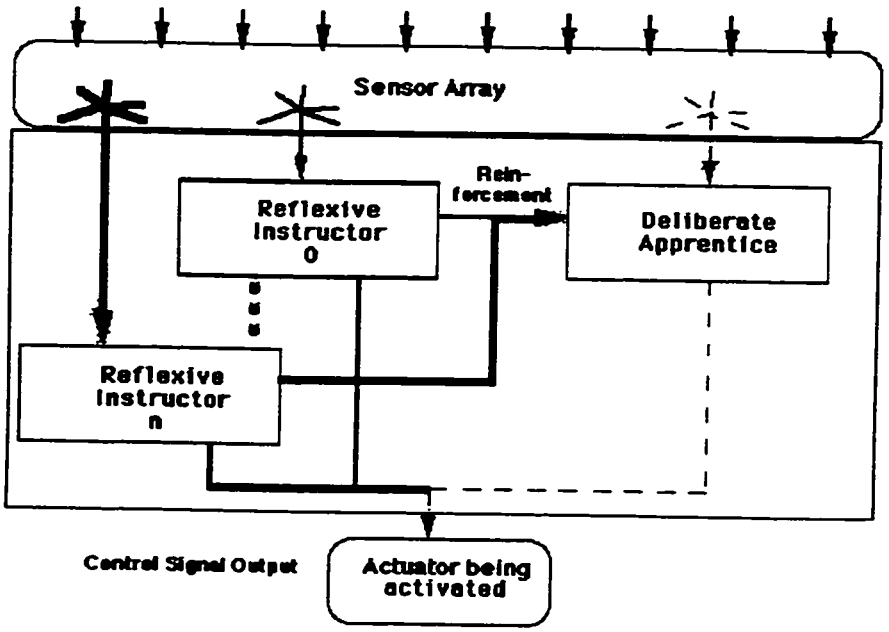


Figure 3-1 The RIDA model.

The architecture presents a hierarchical control scheme which places the RIs in a position where they are the ultimate arbiters of the control signals which reach the actuator. The RIs defer to the control signals of the DA so long as the DA's control signals do not result in any of the RIs' activation.

3.2.1 Comparison with simple Reinforcement Learning

[Sutton et al. 91] and [Barto 92] provide a working definition of reinforcement learning:

> If an action taken by a learning system is followed by a satisfactory state of affairs, then the tendency of the system to produce that particular action is strengthened or reinforced. Otherwise, the tendency of the system to produce that action is weakened.

This definition is illustrated in figure 3-2 where the environment provides reinforcement--R to some form of learning mechanism which receives input from the environment, makes decisions and consequently produces output. The cycle continues with more reinforcement.

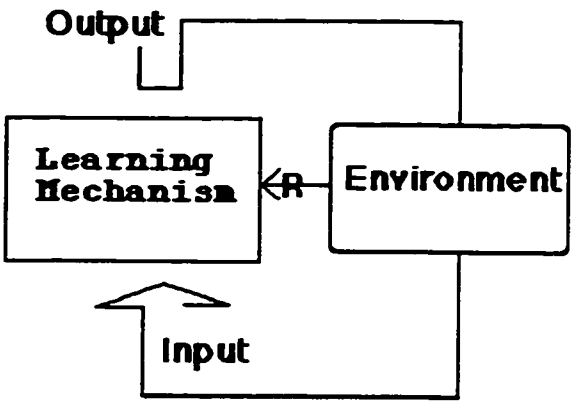


Figure 3-2 Reinforcement Learning.

The interaction between RI and DA is similar to that which underlies various reinforcement learning models where reinforcement is provided to a learning mechanism by the environment within which a particular action took place. However RIDA improves on simple reinforcement addressing two important reinforcement deficiencies.

1) Reinforcement learning models typically are capable of sending only sparse reinforcement signals indicating, at best, the degree to which a control system has done something right or wrong.

2) Simple reinforcement models, while indicating the suitability of a controller's control signals, do not intervene in the controller's "decision"--which is allowed to proceed on an erroneous path.

The RI portion of RIDA is fully capable of providing control signals when necessary. In addition, although the RI is capable of passing simple reinforcement signals to the DA it is possible to pass a much richer signal if necessary.

3.2.2 Comparison with Supervised Learning Model

Because a richer indication of appropriate behaviour can be accommodated through the RI reinforcement signal, RIDA shares some similarities with supervised learning models (SLM) as shown in figure 3-3.

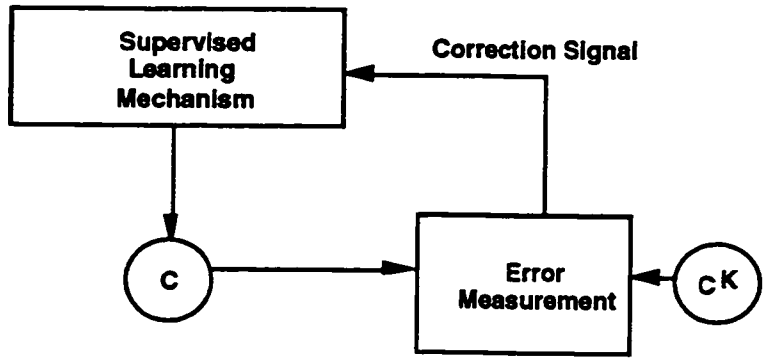


Figure 3-3 Supervised Learning.

In supervised learning, the control signal (C) generated by the learning mechanism is compared with a reference control signal (C^K) and the appropriate adjustments made to the knowledge representation within the mechanism via a correction signal. This model has the unfortunate characteristic of inevitably requiring some form of off-line learning as the initial control signals sent by the SLM will, in all probability, be quite inappropriate--possibly leading to the physical destruction of the system being controlled.

Because the RIDA architecture supports the initial "ignorant" state of the DA, this problem is avoided. The RI is always capable of sending an arguably appropriate control signal to the actuators.

3.3 *RIDA Formalism*

The following sections introduce each component of the RIDA architecture, explain their function and interaction. The RIDA description will be followed by several examples where RIDA might be applied to existing systems.

3.3.1 Assumptions

The RIDA architecture exists in an environment represented by figure 3-4.

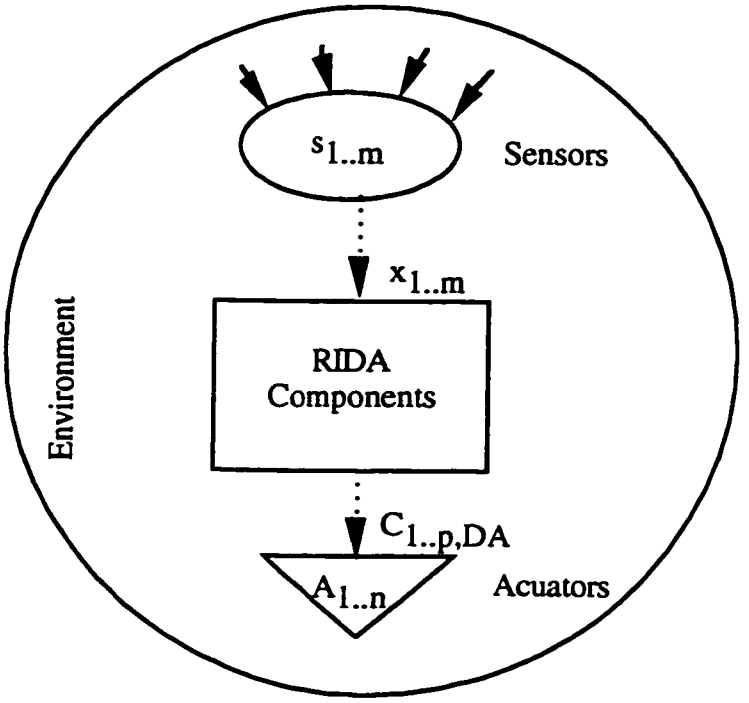


Figure 3-4 Environmental System Assumptions.

It is assumed that data is gathered from the environment by sensors s_1 through s_m where m is the number of sensors provided. The sensors need not be limited to individual units (such as single contact switches) but can be combined into subsystems as long as the subsystem is capable of producing a single sensed value per subsystem label. These individual values are x_1 through x_m.

This form of sensor arrangement is shown in figure 3-5 and has been reported by [Dorigo 93], [Brooks 91] and others. In this case the sensors s_1 through s_4 provide input to an integration function Σ which produces the final output x_1.

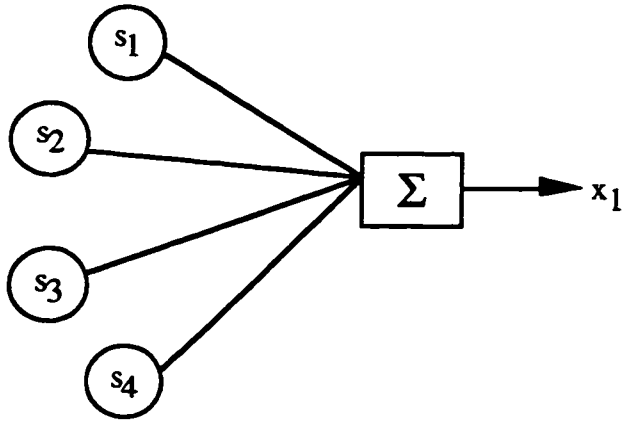


Figure 3-5 Multiple Sensor System Producing Single Output Signal.

The sensor data x is distributed to the DA and RI components capable of making use of it as determined by the system designer. Clearly, certain values of x will be inappropriate for certain RI components as they may be incapable of making use of it.

The sensed values are received by RIDA and used to make control decisions as will be discussed in the following sections. RIDA can issue $p+1$ individual control signals where p is the number of RI components in the RIDA hierarchy. C_1 through C_p are the potential control signals generated by the RI components, and C_{DA} is the control potential signal generated by the DA component. These signals govern the actuators A_1 through A_n where n is the number of actuators subject to RIDA influence.

3.3.2 General Description and Conventions

The RIDA architecture forms a cascading control system where a control system is taken to mean an interconnection of components forming a system configuration that will provide a desired system response [Dorf 92] and components of the system provide individual responses which can be overridden by responses higher in the hierarchy. A process or element being controlled can be represented by the block diagram given in figure 3-6.

Figure 3-6 Process to be Controlled.

The input elements of the system are provided by what is labeled as "environment" in figure 3-4 and sampled by "sensors". While the output is provided by "actuators" which respond to control signals.

Initial control signals are generated by a DA component. As RI components are activated they inhibit the control signals of the DA and RI components which are less trusted members of the hierarchy then themselves. Eventually only a single RI element is left in the chain and its activation inhibits all other components.

3.3.2.1 Hierarchy of Control

The concept of a hierarchy of control is derived from work involved with complexity regulation of the selection of nonlinear models of physical phenomena. Statistics literature provides various examples of limiting model complexity including [Rissanen 78] and an information theoretic criterion by [Akaike 74]. Although these criteria differ from each other

considerably, they share a common form of composition as described by [Haykin 91] in equation 3-1:

$$\begin{pmatrix}\text{Model - complexity}\\ \text{criterion}\end{pmatrix}=\begin{pmatrix}\text{log - likelihood}\\ \text{function}\end{pmatrix}+\begin{pmatrix}\text{model - complexity}\\ \text{penalty}\end{pmatrix} \quad (3\text{-}1)$$

The log-likelihood function is simply a performance measure of the model, while the penalty associated with the complexity of the model reduces the likelihood that it will be selected. The problem with this formulation lies in the selection criteria for these two elements.

A RIDA hierarchy attempts to limit the complexity of the control model by replacing the log-likelihood function with a series of empirically reliable RI components whose performance measurement is left to other RI components deemed by the system designer to be more reliable. By making this assumption we can eliminate the model-complexity penalty. However, this means that considerable effort must be exerted by the designer to ensure adequate performance of the RI and DA interactions. This is a continuing limitation of RIDA, as there are no automatic means of creating a hierarchy.

3.3.3 The Deliberate Apprentice

The following section describes the DA component of the hierarchy. While there can be many RI components only a single DA is allowed. The DA component is shown in figure 3-7.

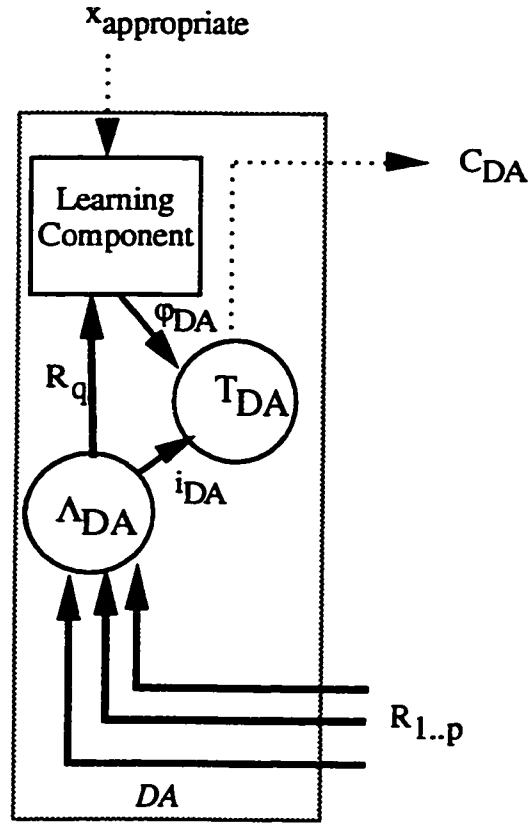


Figure 3-7 The DA Component Architecture.

The signals $x_{appropriate}$ are a subset of the available sensors signals $x_{1..m}$ consisting of those signals which can be interpreted by the learning component of the DA which will be discussed in the next sub-section.

The learning component makes control decisions based on the sensor data and issues a tentative control signal φ_{DA}. The signal is tentative because it is subject to being overridden by the inhibition signal i_{DA}. The inhibition signal is provided in order to prevent the learning component from issuing a control signal which would be inappropriate as determined by the activation of one or more RI components. The reinforcement and inhibition function Λ_{DA} is governed by the reinforcement signals R_1 through R_p coming from activated RI components, where p is the number of RI components.

Λ_{DA} passes R_q to the learning component, where q is the number of the RI component which has inhibited all lower ranking activated RI components as will be discussed in the next section. Λ_{DA} also produces the inhibition signal i_{DA} based on equation 3-2.

(3-2)

$$i_{DA} = \begin{cases} \text{active if } R_q \text{ active} \\ \text{inactive otherwise} \end{cases}$$

T_{DA} is the DA control signal transfer function. It acts as a gate for the passage of the tentative learning component control signal φ_{DA}. This signal will become the actual DA control signal to reach the actuators based on equation 3-3.

(3-3)

$$C_{DA} = \begin{cases} \varphi_{DA} \text{ if } i_{DA} \text{ inactive} \\ \text{inhibited if } i_{DA} \text{ active} \end{cases}$$

The success of the DA component hinges largely on its ability to learn the control task. This is usually limited by the speed of convergence to a solution state of the learning component of the DA. The following section discusses the issues associated with choosing an appropriate learning component.

3.3.3.1 The Learning Component

While great strides have been made in purely reactive systems by Brooks and others, their inadequacies become apparent when they are actually constructed. In order for such an agent to be effective in its intended environment, its designer must have anticipated everything that the vehicle is likely to encounter.

Inevitably, the designer must redesign after observation in an iterative process until the vehicle functions effectively. While this may be considered an advantage by some [Brooks 86], it is also a severe limitation. One cannot send an autonomous agent to a distant planet and continue to tweak it into correct behaviour.

Most autonomous systems researchers have, at least tacitly, seen the advantage of a system which is capable of adapting through learning. In some cases--subsumption for example--this has lead to an existing architecture undergoing considerable revision in order to accommodate learning [Mataric 92].

RIDA has not taken this approach. While a RIDA vehicle can function with merely its reflexive components, inherent learning allows much better performance and allows a path for graceful degradation of service.

3.3.3.1.1 Motivation for selecting a Learning Component

In order to change the behaviour of an agent and allow it to adapt successfully, it must learn from its experience within its environment. For the learning component to be considered successful it must meet two criterion.

1. It must learn quickly otherwise the system's performance would be little better than that of the RI's. Thus the system would be doing nothing more than reacting to its environment most of the time.
2. The learning component must learn the task adequately [Kaelbling 96] as opposed to perfectly [Watkins 92] since the price for perfect performance might be extended learning time--revisiting the concern of the first criterion. Of course, inadequate learning might actually lead to even worse

performance than a controller with the same task relying solely on a reactive strategy.

Slow learning is a common thread running through much of the literature concerning learning algorithms. In a comprehensive review, [Lin 92] performed simulations involving eight distinct learning methods. While each successfully learns to adapt to an environment, they did not do so in real time. Each learning technique required between tens and hundreds of trials before a behaviour was successfully learned.

To accomplish the goals of rapid learning and adequate performance, several learning algorithms were examined as potential candidate learning components. These included,

- Single Layer perceptron [Rosenblatt 62]
- Multi-layer perceptron [Bryson and Ho 69]
- Associative Reward Penalty (ARP) [Barto and Anandan 85]
- Rapid Reinforcement Network [Fagg, et al. 94]

We will now examine these in some depth.

3.3.3.1.2 Single Layer Perceptron

The perceptron relies on the simple delta learning rule [Widrow and Hoff 60] to update its weight matrix. This is given by equation 3-4,

$$w_i(t+1) = w_i(t) + \eta \Delta x_i(t) \tag{3-4}$$

where w_i is an individual weight at time t, η is a gain function controlling the learning rate and x_i is an input signal.

While a perceptron is capable of solving only linearly separable problems, it can usually do this very rapidly with few presentations of training data if the learning rate is set relatively high. Several simulations were run in attempting to teach a perceptron an input pattern which represented a wall at various aspect angles. Typically the perceptron learned these within ten iterations.

The perceptron's speed was considered a significant advantage even though it is well understood that the perceptron has severe architectural limitations [Minsky and Papert 69] which do not allow it to address some of the more difficult aspects of credit assignment.

3.3.3.1.3 Multi-Layer Perceptron

The multi-layer perceptron, or the backward error propagation algorithm, has been applied to a wide variety of learning tasks. The network architecture consists of two or more layers of weight matrices which are capable of finding non-linear mappings between an input and a goal in finite time.

The model of each artificial neuron in such a network includes a nonlinearity at the output. The nonlinearity is a smooth and differentiable at all points. A commonly used form of nonlinearity that satisfies this requirement is the sigmoidal nonlinearity as exemplified by the logistic function shown in equation 3-5,

(3-5)

$$y_j = \frac{1}{1+\exp(-v_j)}$$

where v_j is the net internal activity level of neuron j, and y_j is the output of the neuron. The learning rule is derived from [Rumelhart et al. 1986] and given in equation 3-6.

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta \delta(n) y_i(n) \qquad (3\text{-}6)$$

where α is the momentum of learning, η is the learning rate, δ is the error signal and Δw_{ji} is the change in the weights of a neuron in the network's memory matrix. This equations is usually termed the generalized delta rule.

In simulation, it was found that a multi-layer perceptron would be quite unsuitable since its learning rate required was too slow to be tenable, requiring several hundred repetitions of training data to learn of the existence of a single wall. Since each iteration implied a collision with a wall, this was clearly unacceptable.

3.3.3.1.4 Associative Reward Penalty

A learning procedure attributable to [Barto 85] [Barto and Anandan 85] [Klopf 82] applicable to stochastic environments is the Associative Reward Penalty algorithm (*ARP*). The algorithm relies on a set of stochastic output units governed by the Ising spin model shown in equation 3-7 [Peretto 84],

$$\text{Prob}(S_i = \pm 1) = \frac{1}{1 + \exp^{\pm 2Th_i}} \qquad (3\text{-}7)$$

Where S_i represents an output unit activation, $h_i = \Sigma w_{ij} V_j$ representing the weighted sums of the inputs V filtered through the weight matrix w, and T

the pseudo-temperature of the unit controlling the noise level within the system as a whole [Glauber 63].

There is no exact error measurement, simply the reinforcement signal--r, received from the environment. If we assign r = +1 as a positive reinforcement (reward) and r = -1 as a negative reinforcement (penalty). From the reinforcement signal it is possible to construct a target output pattern. Each Processing element in an *ARP* network can be thought of as similar to the one shown in figure 3-8.

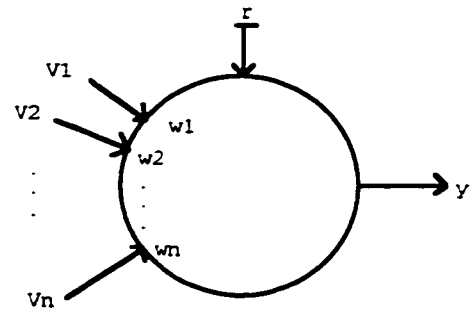


Figure 3-8 An ARP Processing Element.

Input pathways labeled V1 through Vn carry non-reinforcing input signals, each of which has an associated weight w. The input labeled r is the reinforcement pathway. Output is through path y. When the network is to update its weights the signal r is received by all elements simultaneously thus allowing a single reinforcement signal to be used [Hertz et al. 91].

An *ARP* network has many interesting features to recommend it for use in an inconsistent and noisy environment.

- Networks constructed in this way have been shown to achieve similar performance to a network trained with backward error propagation in learning a complex task[2].
- Such a network is capable of learning on-line and will continue to do so throughout its operation.
- The reinforcement signal is global, avoiding the problems associated with error assignment and propagation.

While these attributes are admirable, the networks typically require hundreds of performance trials to learn even simple tasks. Although ARP is a very promising network architecture it was abandoned due to its slow learning rate [Elgersma 94].

3.3.3.1.5 Rapid Reinforcement Network

Since the number of actions of a mobile robot is limited by its actuators, it is common practice to attempt to map what the sensors perceive to actions the actuators can actually perform.

[Fagg et al. 92] have suggested a means for performing this mapping employing a modified feed-forward, winner-take-all neural network to perform the selection of the next action and using a punishment/reward signal to act as a reinforcement generator. The is called a Rapid Reinforcement Neural Network (RRNN). We will concentrate on this work as it will be employed later in the construction of our model.

Temporal Credit/blame Assignment

The problem of assigning credit or blame to the action which was most responsible for an agent's situation can best be described by analogy. A person

[2] Stacey, D.A. (1994), The University of Waterloo, Personal Communication concerning her implementation of an ARP network at the University of Washington.

is driving their car and takes a wrong turn on a road. They continue to drive and miles down the road they come to a stop at a cliff. The salient question is what caused them to be in the situation they are in? Was it the action they took the split second before they came to a stop or was it the wrong turn they made? Obviously the wrong turn was significant but in order to assign blame appropriately, the action must be remembered and somehow avoided in the future.

[Klopf 82] and [Barto et al. 83] suggest a method for reducing the difficulty of determining credit by developing an evaluation function which employs a temporal component. They introduce the concept of eligibility expressed in the equation 3-8:

$$w_i(t+1) = w_i(t) + \alpha r(t) e_i(t) \qquad (3\text{-}8)$$

where a is a positive constant determining the rate of change of w_i, $r(t)$ is the reinforcement at time t, and $e_i(t)$ is the eligibility at time t of input pathway i.

The concept is quite elegant. Whenever certain eligibility conditions hold for input pathway i, then that pathway becomes "eligible" to have its weights modified, and it remains eligible for a period of time after the condition has occurred. If the raw reinforcement signal improves performance, then the weights of the eligible pathways are changed so as to make the element more likely to do whatever it was that caused it to do what it did. A negative signal makes it less likely.

Klopf proposed that a pathway should reach maximum eligibility a short time after the occurrence of the association of an input pathway and the firing of

an output element and decay towards zero after that. This implies a decay rate applied to $e(t)$ as shown in equation 3-9,

$$e_i(t+1) = \delta e_i(t) + (1-\delta)y(t)x_i(t) \tag{3-9}$$

where $0 <= \delta <= 1$ and determines the decay rate. Each pathway, of course, would have its own eligibility. Barto demonstrated a simple network which learned the pole balancing problem [Widrow et al. 64].

While this method adds temporal information, the action which should be most rewarded or blamed could fall outside the window provided by temporal decay. [Sutton 88] suggests a mechanism for predicting the outcome of a certain actions by using previous experience. Rather than predictions based on predicted and actual outcomes, this method assigns credit by means of temporally successive predictions.

In our case we are particularly interested in current and past reinforcements. This is accomplished by equation 3-10,

$$R'(t) = R(t) + \lambda P(x(t+1)) - P(x(t)) \tag{3-10}$$

Where $R(t)$ is the reinforcement received at time t. λ is the discount factor for future reinforcement. P represents the predictions both current $(t+1)$ and previous (t) and x is the state of the system at time t. $R'(t)$ is a modified reinforcement signal which measures the deviation of the actual reinforcement from that which was expected by a prediction mechanism. This means that if $R' > 0$ the system performed better than expected and $R' < 0$ it

performed worse. Sutton points out that this can be used as an internally generated reinforcement signal during every performance step.

The prediction mechanism itself can be any linear learning function, both Sutton and Fagg suggest a linear neural network such as an ADELINE [Widrow 60] as being adequate to the task.

3.3.3.1.6 Implementation with a RRNN as Learning Component

In the case of a RRNN the network produces the output φ_{DA} as shown in figure 3-9. The action selection units select from potential actions "L", "LF", "F", "RF", and "R". These stand for "left", "left forward", "forward", "right forward" and "right" respectively. As an action is selected, the signal is encoded and passed to the transfer function T_{DA} where its eventual transmission to an actuator is governed by equation 3-2.

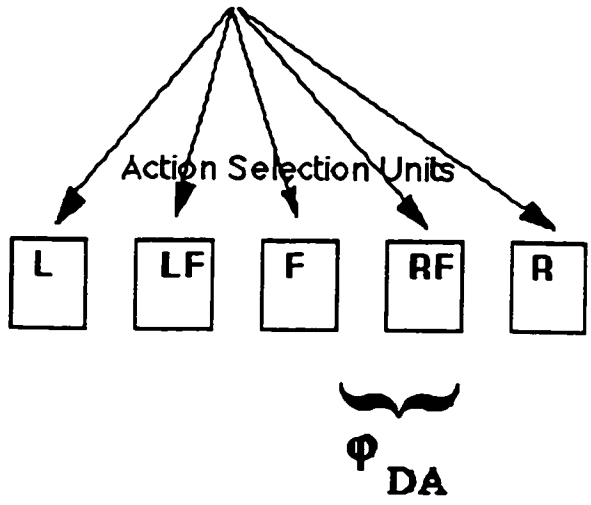


Figure 3-9 The Output of a RRNN as part of the DA.

The complete RRNN algorithm is provided in appendix C.

3.3.4 The Reflexive Instructor

The following section describes an arbitrary RI component of the hierarchy. While there is only a single DA component there can be many RI components. The kth RI component in a hierarchy is shown in figure 3-10.

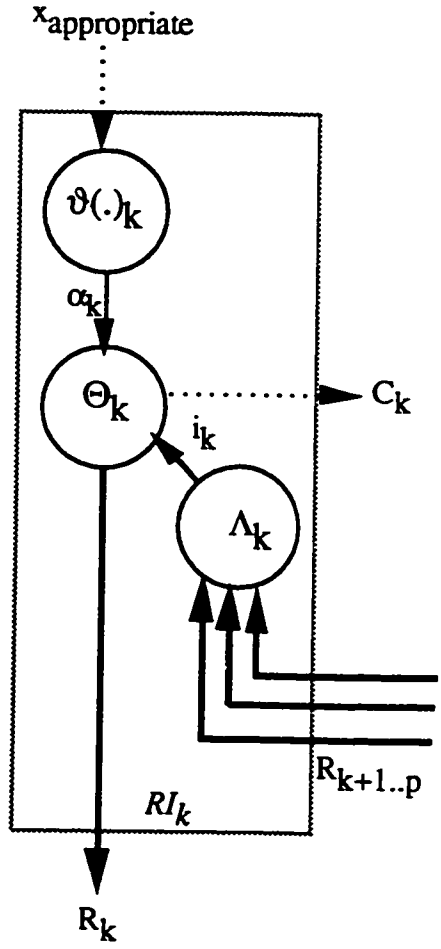


Figure 3-10 A RI component.

The signals $x_{appropriate}$ are a subset of the available sensors signals $x_{1..m}$ consisting of those signals which can be used by the activation function $\vartheta(.)_k$ of the reactive component of the RI.

The activation function is specific to the type of RI which is implemented and the activation policy associated with it. For example a single whisker sensor might provide activation if its contact is closed, or the function might become

active only if a certain number of sensor inputs is received. In any event, $\vartheta(.)_k$ produces the tentative control signal α_k. The signal is tentative because it is subject to being overridden by the inhibition signal i_k.

The inhibition signal is provided in order to prevent the RI_k component from issuing a control signal which would be inappropriate as determined by the activation of one or more RI components which are higher ranking than it is. The inhibition signal is produced by the inhibition function Λ_k which is governed by the reinforcement signals R_{k+1} through R_p coming from other activated higher ranking RI components, where p is the number of RI components. Inhibition is governed by equation 3-11,

(3-11)

$$i_k = \begin{cases} \text{active if } R_r \text{ active} \\ \text{inactive otherwise} \end{cases}$$

where R_r is an active reinforcement signal.

The tentative control signal α_k is passed through the RI_k transfer function Θ_k. This function, similarly to the DA transfer function, acts as a gate for the passage of the tentative RI_k control signal α_k. This signal will become the actual RI_k control signal to reach the actuators based on equation 3-12.

(3-12)

$$C_k = \begin{cases} \alpha_k \text{ if } i_k \text{ inactive} \\ \text{inhibited if } i_k \text{ active} \end{cases}$$

$\vartheta(.)_k$ is also responsible for producing a reinforcement signal R_k which will inhibit lower ranking RI components and the DA component if activated. The activation of R_k is governed by equation 3-13.

(3-13)

$$R_k = \begin{cases} \text{active if } \alpha_i \text{ active and } i_i \text{ inactive} \\ \text{inactive otherwise} \end{cases}$$

It should be noted that each RI component can be described using figure x-x except the highest ranking RI component RI_p which, by definition, has no higher ranking RI supporting it. In this case no reinforcement signal will inhibit its control signal and it architecture is a slight modification of figure 3-10 and is shown in figure 3-11.

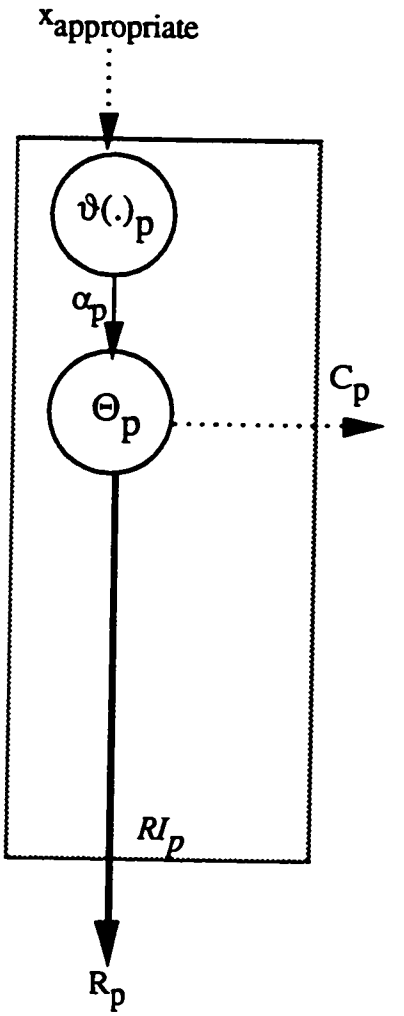


Figure 3-11 The Final RI Component.

3.3.5 Assembling the RI and DA Components

We can assemble the components described in previous sections to form a RIDA hierarchy as shown in figure 3-12. There are essentially two forms of data pathways in the RIDA architecture, reinforcement and control. All reinforcement paths are indicated by heavy lines, while control paths are represented by lighter lines.

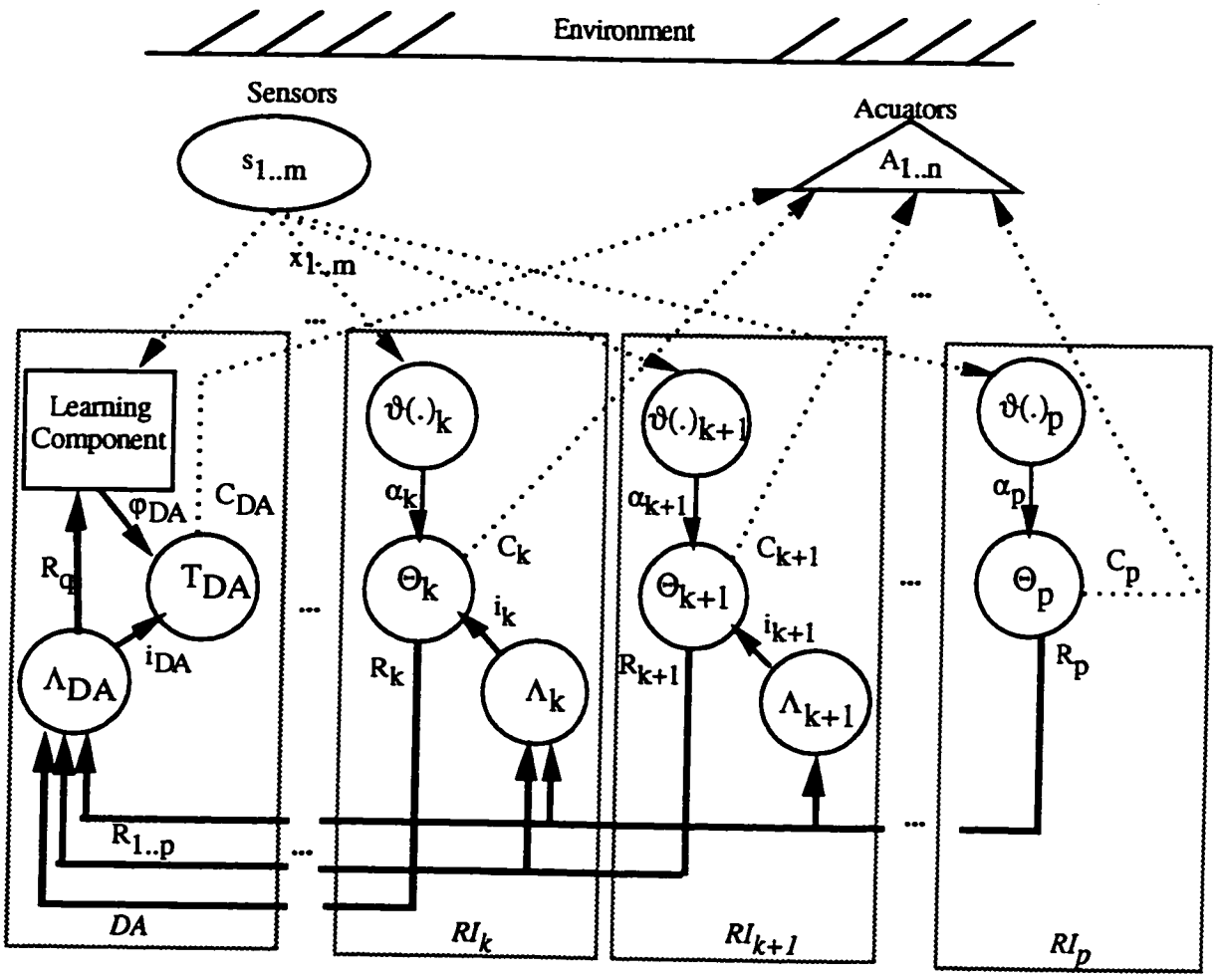


Figure 3-12 The Assembled RIDA Components.

The DA component can be seen on the far left of the diagram. It is the least trusted and lowest ranking component of the hierarchy. As one moves from

left to right components become more trusted and inhibit the control signals of all the components they out-rank. For example, the tentative control signal φ_{DA} of the DA can be inhibited by any of the RI components shown. RI components RI_k and RI_{k+1} are two typical RI components where RI_{k+1}'s reinforcement signal R_{k+1} will inhibit RI_k and the DA component. RI_p is shown on the far right of the figure. By necessity RI_p is the highest ranking component and receives no supporting inhibiting input.

3.3.6 RIDA Applicability Examples

The following sections provide several examples of how RIDA might be used in control problems. The first example describes how RIDA might be applied to a fictitious design problem and is used as an illustration of how RIDA components can be assembled. The second example uses the subsumption architecture as described by [Brooks 86] to develop a highly reliable RI component, in this way using subsumption as a building block for RIDA.

3.3.7 An Example RIDA Hierarchy

As an illustration of the RIDA concept, let us turn first to a specific design problem. Suppose a mobile robot was designed with three different sensors as shown in figure 3-13. A let us further assume that the mission of the robot is to navigate around an unknown environment where there are an unknown number of obstacles and potential dangers such as cliffs.

We could create a RIDA cascading control hierarchy to address this unknown environment assuming there exists an appropriate learning component which is capable of learning to navigate in an unspecified way using sonar as input.

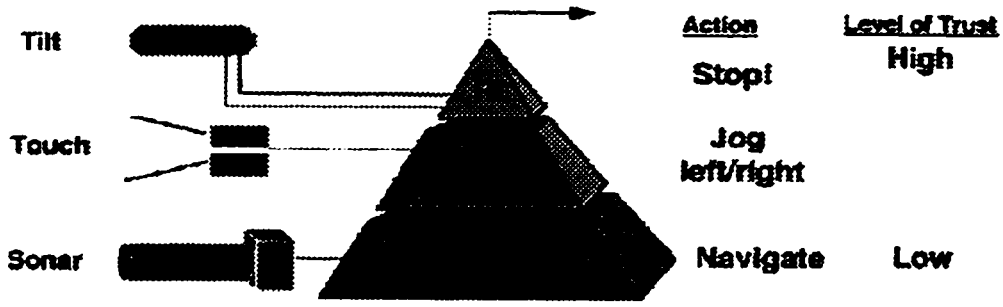


Figure 3-13 An example RIDA Cascading Hierarchy.

Figure 3-14 is the block diagram for the configuration of the RIDA components to implement this system. The Sonar passes signals to the Navigation learning component which makes decisions concerning where to go next. If the vehicle contacts an obstacle the touch sensor interacts with the first RI component which overrides the DA control signals to the actuators. If the vehicle is in real danger of falling off a cliff, the second RI components overrides both the DA and the first RI component.

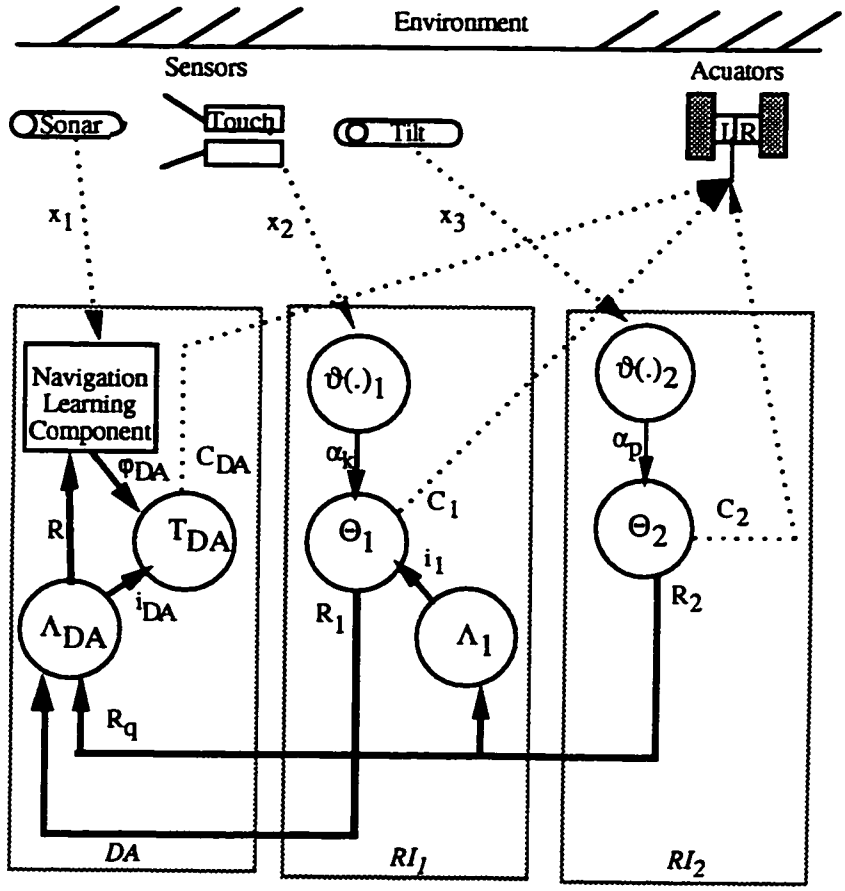


Figure 3-14 Specific RIDA Design

The cascading failures are shown in figure 3-15. Because of the gradual nature of the failures the system as a whole continues to be viable for an extended period.

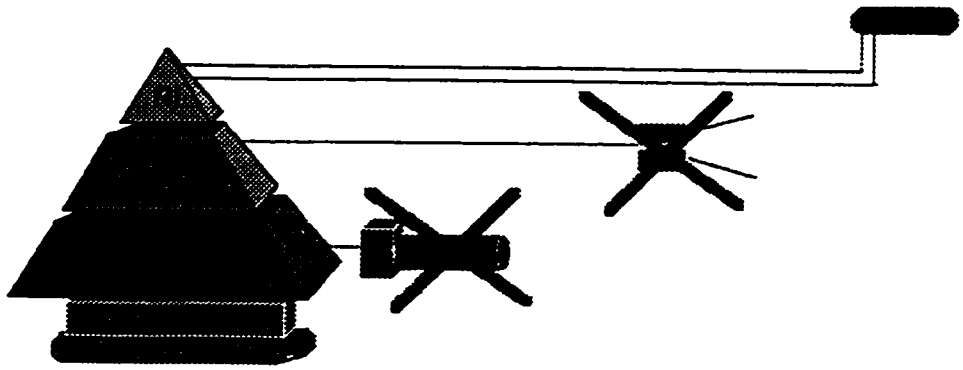


Figure 3-15 Graceful Degradation of Service

3.3.7.1 RI Implementation Using Subsumption

As an illustration of selecting and implementing an RI component we turn to the Augmented Finite State Machine (AFSM) based subsumption architecture as described in [Brooks 86]. Brooks defines of competence in autonomy tasks. Level 0 competence is defined as "Avoid contact with obstacles" (whether the objects move or are stationary).

An AFSM is depicted in figure 3-16.

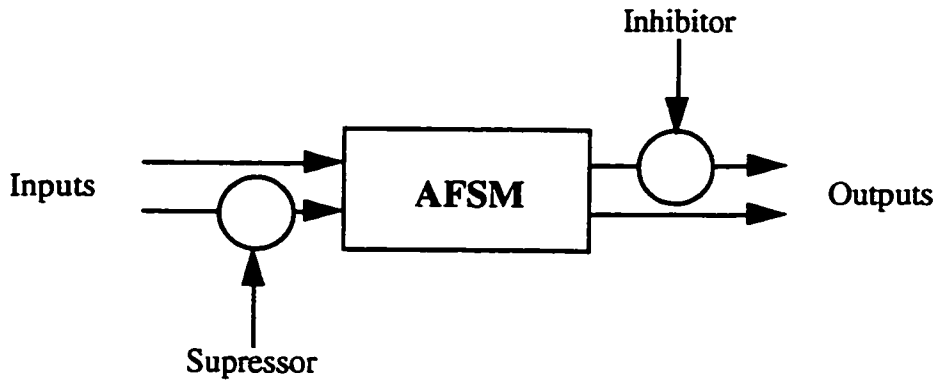


Figure 3-16 An AFSM.

The inputs could come from the environment in a similar manner to figure 3-4 or could come from other AFSM involved in a loosely coupled organization. The AFSM is augmented with timers which either allow or disallow activity of an AFSM if a required input is not received after a certain amount of time. Supressors and Inhibitors modify the behaviour of the individual AFSMs.

Brooks uses the AFSMs to Describe a level 0 competence module as shown in figure 3-17.

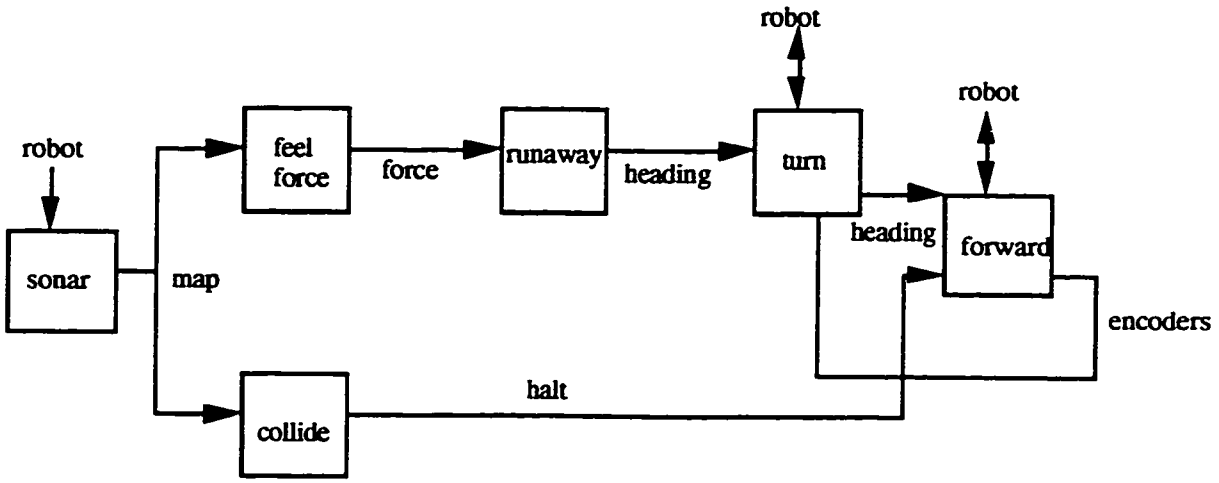


Figure 3-17 Level 0 Control System as Described by Brooks.

Each box in the diagram represents a single AFSM module. The primary sensors are the sonar which are attached to the forward area of the robot. Collisions are handled by the "collide" AFSM which sends an appropriate signal to the "forward" AFSM to stop movement. This signal is, in turn, passed on to the vehicle's actuators involved in forward motion. As a whole or in parts, this design has been used in various vehicle designs [Brooks 86][Brooks 87][Brooks 89] and has proven to be quite effective in collision avoidance tasks.

It is possible to employ this module as an RI component of a RIDA hierarchy as shown in figure 3-18.

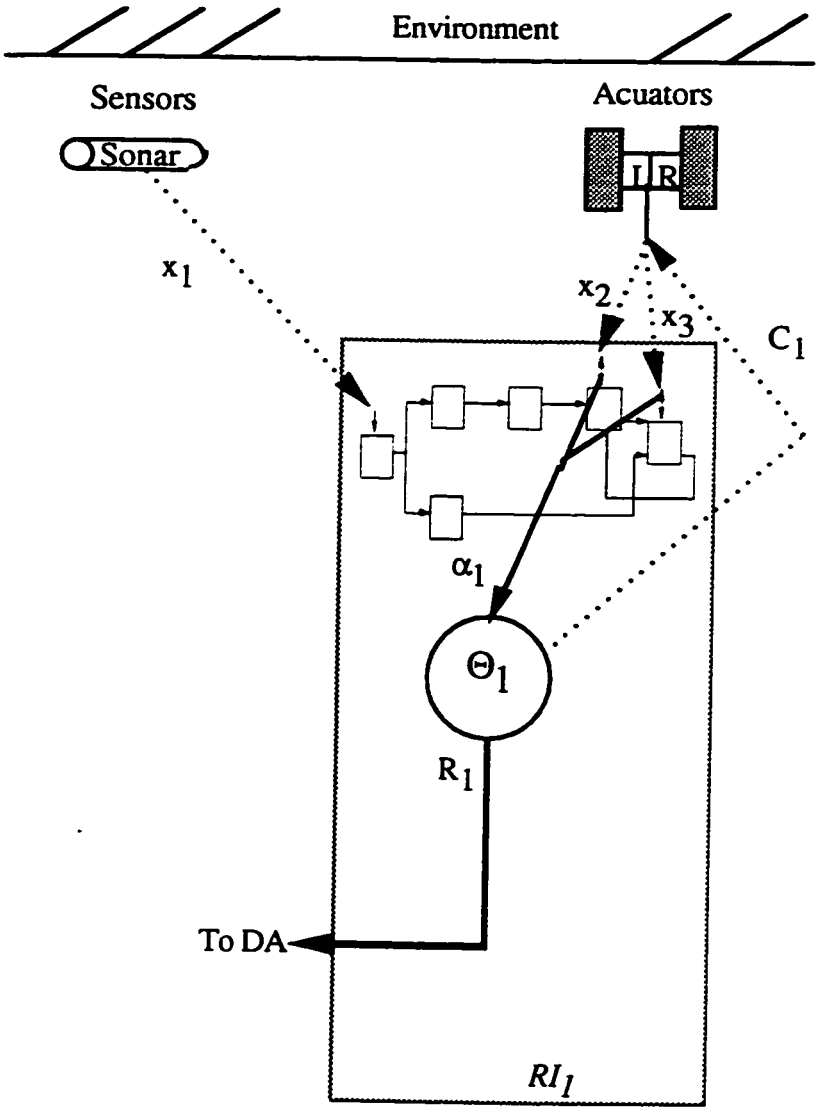


Figure 3-18 Using Level 0 Controller as RI Component.

It is assumed that the AFSMs responsible for "turning" and "forward" command signals are communicating with a differential steering actuators through the α_1 path. x_1 is the signal coming from the vehicle's sonar, x_2 and x_3 are the incoming telemetry signals from the disk encoders or similar devices on the actuators. The modules command signal C_1 will override any command signals sent from the DA and are passed through Θ_1 unmodified to the DA component on RI activation.

By incorporating subsumption in a RIDA hierarchy we can extend learning in a relatively straightforward way to subsumption as long as a sufficiently capable DA is employed. In addition a library of highly reliable low level subsumption modules are available for more capable RIDA vehicles.

3.4 RIDA Rationale

The RIDA architecture is able to address several important issues which have gone unresolved in the field of autonomous vehicle design. These are the problems of,

- reliability, and
- suitability.

3.4.1 Reliability

The fact that an architecture, such as the one being proposed is plausible, does not provide sufficient grounds for its deployment. There must be some level of confidence that the RIDA architecture will be at least as reliable as existing control mechanisms.

Fortunately the architecture is capable of incorporating existing systems within it. If an existing controller is capable of operating within an environment it should be possible to incorporate the controller as the RI side of a RIDA module. In this way we guarantee that the performance will be no worse than an existing system.

The RIDA architecture is reliable for several reasons,

- There is no single point of failure as the RI or the DA will always be active. If the DA fails completely, the RI is capable of controlling the system in a degraded state. This is also the case when the RI is

disabled. A trained DA will continue to issue control signals to the actuators it controls.

- Redundancy is achievable by provisioning more than a single RI. In this way, individual failures have minimal impact on the overall performance of the system.
- Loosely coupled reinforcement signals mean that the DA is not dependent on a single RI sending a timely signal. Even if no signal is sent at all, the system is in no worse shape than if it had only the DA.

Of course, there are no miracles. Sensor failure or the inappropriate selection of RI or DA elements could all lead to catastrophic system failure. However, the appropriate selection of control elements and the careful design of the RI and DA interaction should help mitigate this concern.

3.4.2 Suitability

In short, where can this architecture best be applied? Safety critical applications are certainly potential candidates. It is possible to envision such an architecture applied to vehicle mobility systems designed to enter extremely hazardous environments such as nuclear reactors or chemical fires. Planetary exploration is another potential area as a RIDA equipped vehicle would not be completely reliant on a human operator millions of miles away sending control signals delayed by minutes or hours.

Because an RIDA module provides a degree of fault tolerance in its gradual degradation in response and the fact that a system constructed in this way is able to learn and adapt the architecture is well suited to many types of mobility systems. For example, it should be possible to construct a RIDA for the control of robotic household appliances such as vacuum cleaners that learn to avoid furniture.

3.5 Conclusion

This chapter has introduced the RIDA architecture with respect to mobility control. The various features of the architecture have been addressed and comparisons made with other learning systems. We have shown how the architecture can be used to support the training of an arbitrary learning component, how The RI and DA elements interact, and how it supports the graceful degradation of the controller system as a whole.

4. Learning, adaptation and the Deliberate Apprentice

Go Soothingly in the grease mud, as there lurks the skid demon.

English translation of Japanese traffic sign

4.1 Introduction

In this chapter we discuss the value of incorporating learning as one of the characteristics of the RIDA architecture for the purposes of adaptation. In addition we revisit the theoretical contributions of this work.

4.2 Learning as an aid to adaptation

While great strides have been made in purely reactive systems by Brooks and others, their inadequacies become apparent when they are actually constructed. In order for such an agent to be effective in its intended environment, its designer must have anticipated everything in it that the vehicle is likely to encounter.

Inevitably, the designer must redesign after observation in an iterative process until the vehicle functions effectively. While this may be considered an advantage by some [Brooks 86], it is also a severe limitation. One cannot send an autonomous agent to a distant planet and continue to tweak it into correct behaviour.

Most autonomous systems researchers have, at least tacitly, seen the advantage of a system which is capable of adapting through learning. In some cases--subsumption for example--this has lead to an existing architecture undergoing considerable revision in order to accommodate learning [Mataric 92].

RIDA has not taken this approach. While a RIDA vehicle can function with merely its reflexive components, inherent learning allows much better performance and allows a path for graceful degradation of service.

Subsumption, and other strictly reactive systems, have had a tendency to design individual behaviours into their reactive subsystems and then have used learning to perform higher level tasks such as navigation. In effect, learning is treated simply as an adjunct to the primarily reactive systems.

As an illustrative example, several educational manufacturers will soon start marketing a version of the famous Genghis walking robot [Brooks 89] which is capable of learning navigational tasks. One can argue that even if the learning system were to fail it is unlikely the walking system will since they are independent. However, if the terrain changes in a way that makes walking very difficult, it is impossible for Genghis to adapt beyond its original design since it is fettered by its original assumptions. By applying a RIDA approach to the walking problem it should at least be feasible to adapt to change at this fundamental level.

4.3 Theoretical Contribution

Chapter 3 discussed several learning algorithms. Each has seen implementation in a mobile robot somewhere and has proven to learn a task. The problem is not learning but the speed of learning needed to adapt to an environment not conforming to the system designer's plan and a plan for failure because of it. This is where RIDA's theoretical contribution lies.

As we will see, RIDA contributes to the speed of learning of the algorithm selected--making it learn substantially faster than it would on its own. In addition, because the architecture addresses and depends on failure, a robot adhering to the architecture is better able to cope with an unpredictable environment --where failure is guaranteed to be the norm until successful learning occurs. In addition the architecture ensures that the vehicle plant continues to function by allowing for graceful degradation in performance.

Graceful degradation has not been well addressed in the literature. [Brooks 89] makes explicit reference to planning for failure however, there is no mention of how one plans for failure one did not predict due to poor prediction. For example, various investigators have made reference to fast reactive systems being able to cope with the random conditions associated with a lab environment--the constant rearrangement of furniture, the movement of people--all of these are very well addressed by a system which predicts them and has made allowance for sensing this environmental behaviour.

These systems, like many before them, have approached this problem from the perspective of omnipotence. The investigator has used as many a priori sources of information about the environment as possible to compensate for any possible anomaly. This is fine in a lab environment but has been shown to be highly dubious in an environment where a vehicle must fend for itself [Kaspar 94].

The problem of failure arises when the predicted and instantiated system cannot cope with a situation which has not been predicted. The result, more often then not, is catastrophic system failure. RIDA approaches failure from a local perspective. Because the learning system assumes it is learning, it can also learn from the reflexive system designed to save it. Failure is simply another part of learning, and is compensated for by an instructor which prevents unrecoverable damage to the vehicle.

Reflexive systems have been devised. Learning systems have been created. An architecture which addresses the realities of learning speed, the inevitability of insufficient planning by a robot's designer, the lack of global knowledge associated with an environment which cannot be predicted and the necessity to cope with all of them simultaneously would well describe the theoretical contribution of RIDA.

4.4 Conclusion

In this chapter we have discussed the significance of learning in adaptation and shown that speed of learning is essential for effective adaptation. We have reiterated that the improving learning speed and planning for inevitable failure are some of the significant contributions of this work.

5. Selecting A Reflexive Instructor

A noble heart embiggens the smallest man.

Jebediah Springfield

5.1 Introduction

This chapter describes the RI which was used to confirm the validity of the RIDA architecture. It explains how the RI functions, how it can be constructed and rational for its selection.

The RI was selected on the basis of qualitative reliability, which is appropriate for a controller which is the "final line of defence" against failure (note that in biological systems reflexes are normally the first line). Various trials were conducted to ensure this reliability in several RI strategies and to confirm that these strategies could co-exist on a single vehicle.

5.2 Motivation for selecting a Reflexive Instructor

Clearly the success of RIDA is based on reliable RIs. A reliable RI must have certain characteristics;

- It must be simple and effective. This is because its activation is critical to the successful learning of the DA,
- It must have a high probability of doing the right thing. Since the RI is the primary teaching vehicle of the DA, the RI must be "correct" as much as possible, and
- It must be simple to construct otherwise it becomes difficult to implement.

5.3 Prototype Reflexive Instructors. The SOLENODON trials

The SOLENODON trials were undertaken to select several appropriate reflexive instructor sub-systems for a RIDA implementation. Several reactive control strategies were examined and empirically evaluated in an unstable test-bed vehicle.

The approach taken in this study was to create a vehicle which was extremely difficult to control. Preferably any intelligent control system would find it impossible to master the fine control of such a vehicle. The point of this being that we would be able to compare how well different controllers deal with the inherent unmanageability of the task and how perspective RIs might help them.

A situation is cited by [Norman 86] in discussing the difficulties novice sailors have in learning to steer a compass course using the tiller of a boat. He proposed that "even a task that has but a single mechanism to control a single variable can be difficult to understand, to learn, and to do." Similar systems are evident in nature. For example the Haitian solenodon runs on its toes with a "stiff ungainly waddle, following an erratic almost zigzag course...Moreover, when a solenodon is alarmed and tries to put on speed it is as likely as not to trip over its own toes or even tumble head-over-heels." [Burton 69].

By designing such a vehicle, any mechanism controlling the vehicle--the intended DA, is forced to either fail to achieve successful control or rely on a lower level control mechanism--the RI, to take over control on impending failure. In this way, the RI can be examined on its own merits employing an arbitrary DA.

In the SOLENODON vehicle (discussed in the next section), the DA subsystem was replaced with a manual controller allowing remote operation by a human. This enabled the operator to manipulate the vehicle conveniently and provided a means for testing the RI subsystem independent of the DA.

5.3.1 The RI Tasks

Two tasks were selected as potentially useful candidates for aiding overall control;

- The first task was to avoid collisions and consequently keep the vehicle from risking damage. The RI was designed to keep the body of the vehicle from touching any vertical surfaces which would be detected by whisker sensors.
- The second task was to aid in finding and tracking a light source. The RI in this would shift the vehicle in the appropriate direction to "jog" the vehicle in the direction of the light detected by its photo resistor banks.

5.3.2 SOLENODON IV

A relatively simple functional and robust six legged walking vehicle was constructed to provide the test bed. Such a vehicle was selected because,

- it allowed for very tight turns--a capability seen as very important for escaping situations which might lead to getting stuck, and
- mechanical walking is far less stable--in the sense that "tipping over" and falling are more likely than in wheeled motion. The reduced "steadiness" of walking adds to the complexity of the control task, making it much more difficult to master for the human participants in the trial. This would ensure dependence on the RI to achieve good performance. Thus the reflexive systems would be tested and not the human drivers.

Several designs were prototyped, the one shown below was finally implemented in a working mobile robot. It provides sufficient steadiness to allow walking, sufficient unsteadiness to require the activation of the RI and flexible enough to make tight turns in place. The vehicle could be controlled remotely via a tether to twin joy sticks controlling the direction of motion of the left and right drive trains.

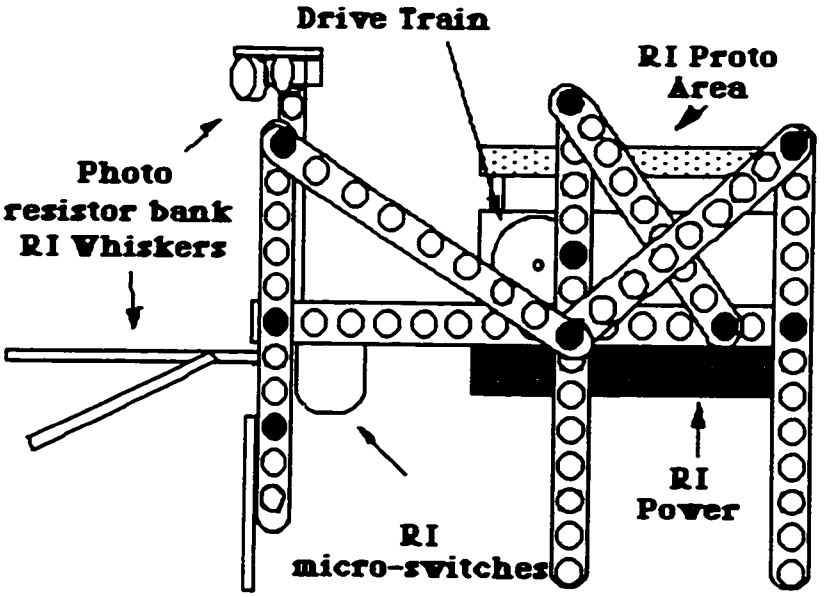


Figure 5-1 Solenodon IV (Schematic View)

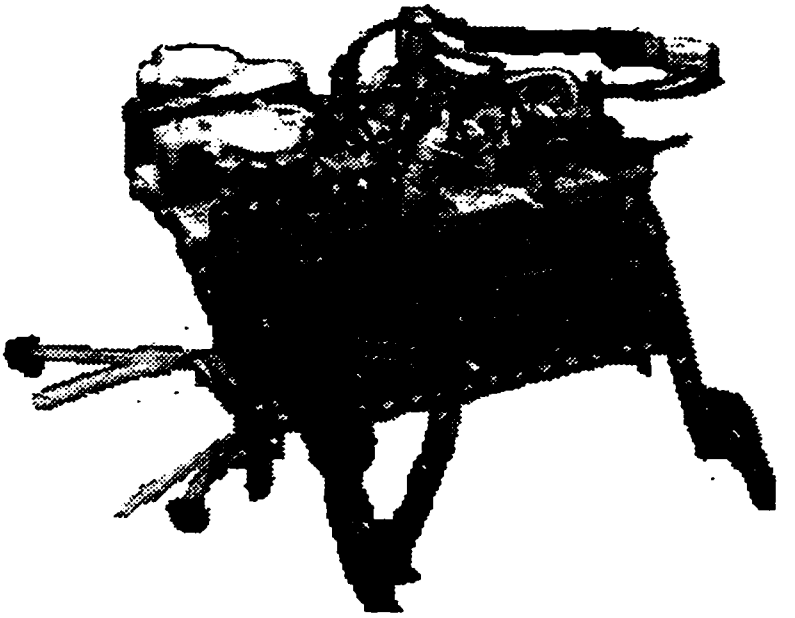


Figure 5-2 Solenodon IV (3/4 view)

Sensing for the vehicle was very simple, employing momentary contact micro-switches activated by a set of sensors inspired by a cat's whiskers. In addition banks of photo resistor were provided to enable the light seeking function of the vehicle.

The original design was employed in an earlier study called EMMA as described in the appendices.

5.3.3 The Collision Avoidance RI

In order for the vehicle to be reliable the reflexive instructor must be able to take over control of the vehicle the moment a control error has been detected. Also, the RI must behave as a "trusted" sub-system in the sense that it must have a high probability of reacting correctly.

With these goals in mind, several RIs were designed to take input from the vehicles whiskers and react appropriately when one or more was activated. Status LEDs were supplied to indicated which sensors were active and what circuit was engaged. These would eventually supply an analog signal to the DA.

The simple RI circuits consisted of Resistor/Capacitor (RC) elements designed to throw Double Pole/Double Throw (DPDT) relays when momentary contact micro-switches were closed. Because the power requirements of such a circuit is quite low, an extra battery of cells could be slung under the vehicle without affecting its performance and making the system completely independent of the tether. This controller is a modified version suggested by [Jones and Flynn 93].

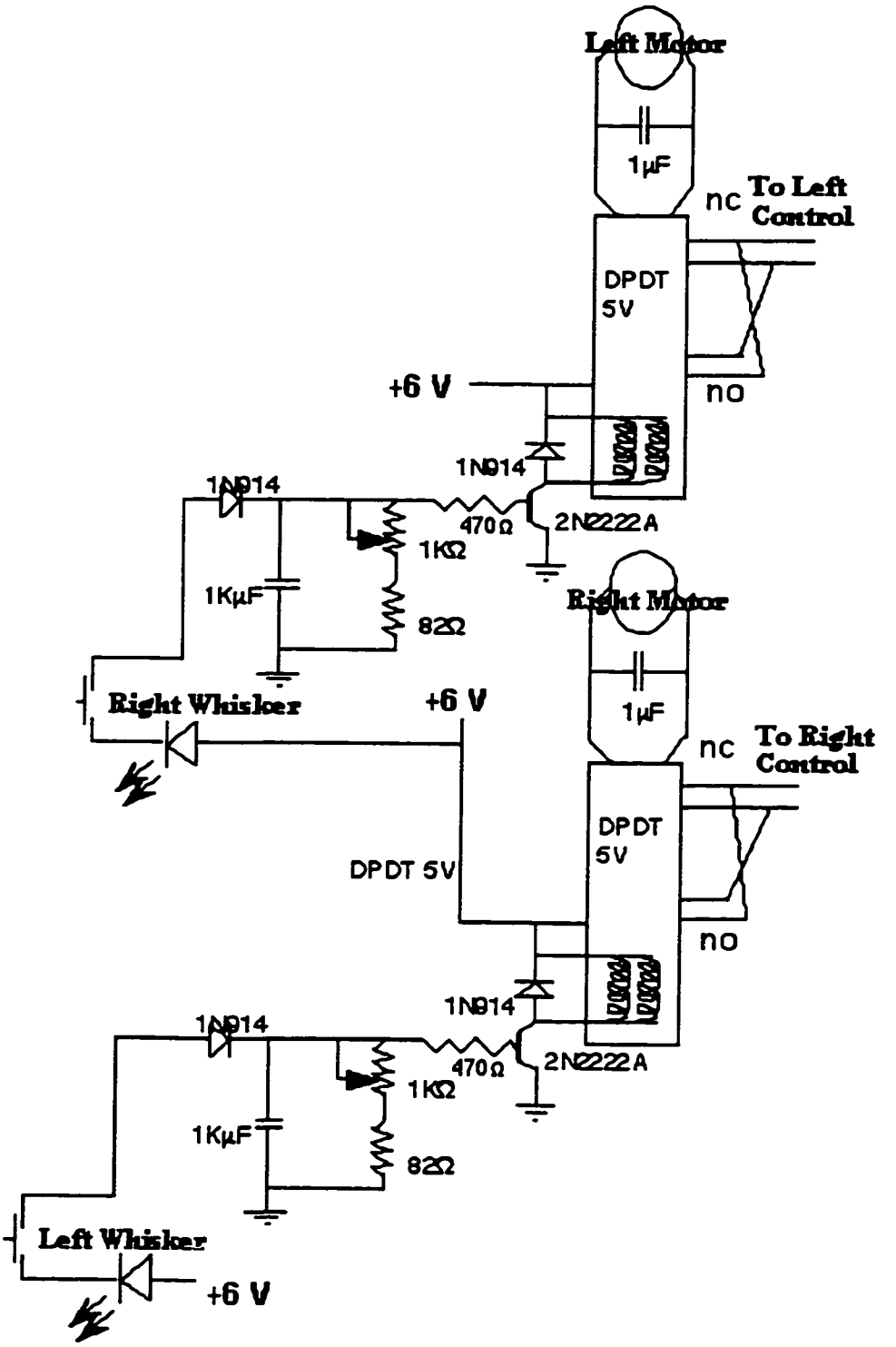


Figure 5-3 The Solenodon Collision Avoidance RI

5.3.4 Simple RI collision avoidance control strategies

All the active control strategies were designed with the following goals in mind, if the vehicle made contact with an obstacle,

- break contact with the surface the sensors contacted, effectively eliminating any danger, and
- veer the vehicle to one side and thus reduce the potential oscillations possible if the vehicle simply backed up (only to move forward and hit the obstacle again.)

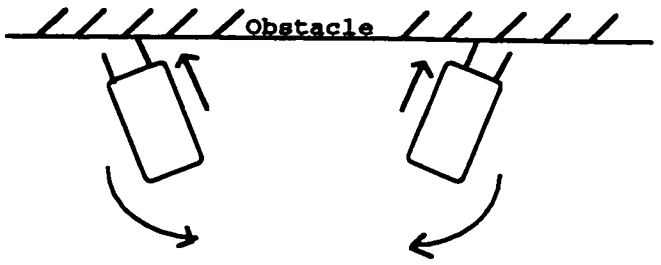


Figure 5-4 RI reacting to contact with one of its sensors

Five control strategies were devised to allow the RI circuits to aid in the control of the vehicle. These were based on observations from the EMMA study (see appendix B).

The first RI strategy was to do nothing. The human operator of the vehicle would be allowed to control it without the interference of an RI. This provided a base case. In order for a RI strategy to be considered useful it must perform better than this strategy.

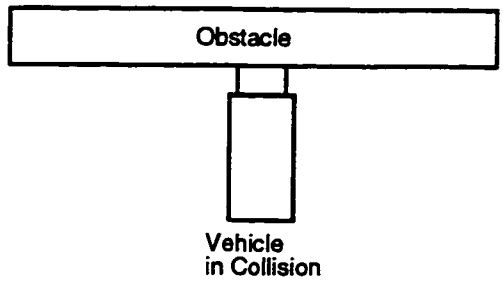


Figure 5-5 RI strategy One

The second strategy involved shunting power to the opposite legs of the vehicle and reversing their motion for a brief "sustain" period governed by the RC constant of the circuit (< 1 sec). This can be paraphrased as "what ever the legs were doing on this side of the vehicle, do the opposite on the other side for a short time". For example, if the vehicle came into contact with an obstacle with its left whisker and the vehicle's left legs were moving forward at the time than control strategy 2 would make the right legs move in reverse. In this situation the strategy would have a tendency to swing the vehicle away from the obstacle.

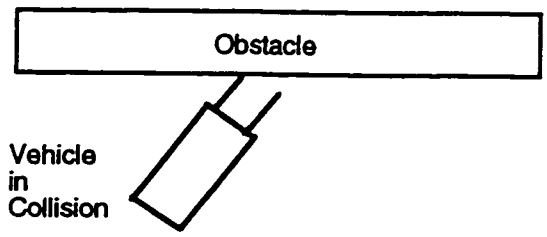


Figure 5-6 RI strategy two and three

In the third RI strategy the sustain time was increased to about 1.5 seconds.

The final two strategies involved the same cross connections as above except the power would no longer be shunted from the human controller but would come from the power supply slung under the vehicle. When a switch was closed a relay would fire and force the other side's legs to back-up. This can be paraphrased as "Whatever side of the vehicle comes in contact with an obstacle, have the other side's legs backup." Note that this is a subtly different strategy then II. Instead of shunting power from one side to the other, additional power is supplied

involuntarily to one side of the vehicle, thus powering its legs without human intervention.

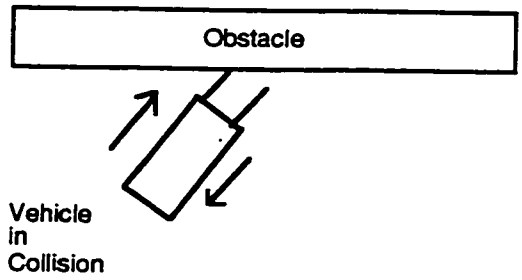


Figure 5-7 RI Strategy four and five

Strategy four provided a short sustain period and strategy five allowed a longer sustain.

5.3.5 Testing the Control Strategies

In order to test the various control strategies a course was constructed in an arena consisting of two straight paths connected by a 180 degree turn. The course was 920 cm long and was lined with 47 pylons and one vertical wall. The pylons were evenly spaced forming the path. The lane the vehicle traveled had a minimum width of 23 cm and a maximum width of 35 cm at the curve.

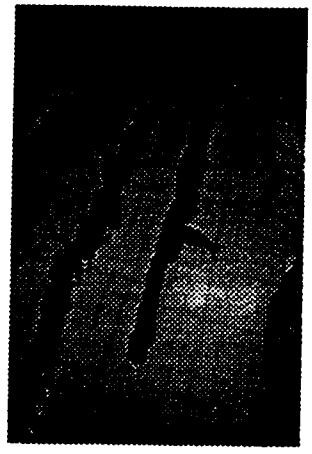


Figure 5-8 The track after a run

The SOLENODON is 16 cm wide (whisker to whisker) and 23 cm long (whisker to extended rear leg). Each pylon could be displaced by the vehicle should it come in contact with one.

Four volunteers were selected as operators. Each operator was given the opportunity to "drive" the test vehicle for at least 15 minutes prior to the actual trials. The test vehicle had no sensors active at that time, although the drivers were aware of the control strategy of the vehicle they were driving during the trials.

The goal given to each of the human operators was to move the vehicle through the course as quickly as possible while minimizing the number of collisions. For the purposes of these trials a collision was defined as "any non-sensor contact with a vertical surface". An observer was assigned to count the incidents of collisions with either walls or pylons. A vehicle traveling the path could make contact with individual pylons or the wall several times with each contact counted by the observer.

5.3.5.1 RI 1 Performance

The results of four trials without an RI strategy are shown below. Although each driver could control the vehicle fairly well, fine control was impossible due to the vehicle's inherent unsteadiness. Most collisions were caused by "unexpected" turns of the vehicle as it walked erratically.

Driver	Sensor Contacts	Obstacle Collisions	Time of Run (seconds)
1	N/A	7	47
2	N/A	11	45
3	N/A	15	47
4	N/A	12	50

Table 5-1 Results with no control strategy

5.3.5.2 RI 2 Performance

This RI seemed to work quite well as long as the vehicle stayed mostly in the centre of the path. Two of the drivers attempted to move very close to the pylons. This set up an oscillation condition in which the vehicle became very unstable and knocked two or three pylons out of the way before the operators could regain control and proceed.

Several of the drivers commented that the vehicle was difficult to control in "rogue" situations because the control signals they were supplying were being reversed by the vehicle's RI circuit and this confused them.

Driver	Sensor Contacts	Obstacle Collisions	Time of Run (seconds)
1	28	2	55
2	24	2	50
3	24	6*	67
4	27	5*	60

Table 5-2 Shunted Control with short sustain time (Asterisks indicate strong collision condition observed in trial)

5.3.5.3 RI 3 Performance

When the sustain time was increased, the rogue behaviour remained, as can be seen from the results of driver 2, and additional problems were observed. Most drivers learned quickly that they could stop the vehicle by releasing the controls however they did not like this technique as it tended to increase the amount of time it took to move around the course. Driver 4 had a very difficult time with this type of control.

Driver	Sensor Contacts	Obstacle Collisions	Time of Run (seconds)
1	31	5	67
2	25	4*	70
3	21	8	61
4	27	7	64

Table 5-3 Shunted Control with long sustain time

5.3.5.4 RI 4 Performance

All drivers found that this strategy served them best in the given task. In fact several of the drivers commented that keeping the vehicle moving forward was much easier with the assistance of the RI circuit.

Driver	Sensor Contacts	Obstacle Collisions	Time of Run (seconds)
1	29	2	41
2	26	2	47
3	24	3	43
4	23	2	48

Table 5-4 Cross Connected reactive control with short sustain time

5.3.5.5 RI 5 Performance

The longer sustain time proved to be disappointing once again. The uncontrolled backing had a tendency to hit obstacles with much greater force than a human operator would have. Even a casual observer would note that simply maintaining control of the vehicle was much more difficult for the operators.

Driver	Sensor Contacts	Obstacle Collisions	Time of Run (seconds)
1	27	5	63
2	27	6	58
3	25	9	65
4	31	8	60

Table 5-5 Cross Connected control with long sustain time

5.3.6 Observations

Cross connections between sensors and actuators proved to be quite useful. It is interesting to note, however, that this is not the control strategy employed in the reflex actions of animals. The SOLENODON withdrew when the opposite sensor was activated. In biological systems exhibiting a simple crossed extension reflex, the sensor that caused the reflex is normally the one withdrawn. The reason for this is that the sensor on a biological system is normally on the actuator that caused the reflex to be initiated in the first place. This is not true on the SOLENODON where the sensors are mounted on the vehicle body not on the actuator itself.

It also became apparent that, whatever reflex is chosen, it should not be maintained for long periods of time. A short sustain period tends to be quite helpful.

5.3.7 The Light Seeking RI

As with the collision avoidance RI, the light seeking RI must also perform reliably and accurately. The RI circuit was replaced with a simple analog comparator-based light seeking RI designed to "jog" the controlled vehicle towards a detected light source without actually taking over control of the vehicle for long periods of time. The circuit is very elegant in its operation. Once a certain threshold of light has been surpassed by the circuits photo-resistor, the op amp allows current to flow to the drive train farthest from the light source. Eventually the vehicle is jogged

toward the direction of the light. Once past a certain point the "blinders" (Appendix C EMMA II.5) mounted in front of the photo resistors block the light and the drive train loses power returning control of the vehicle.

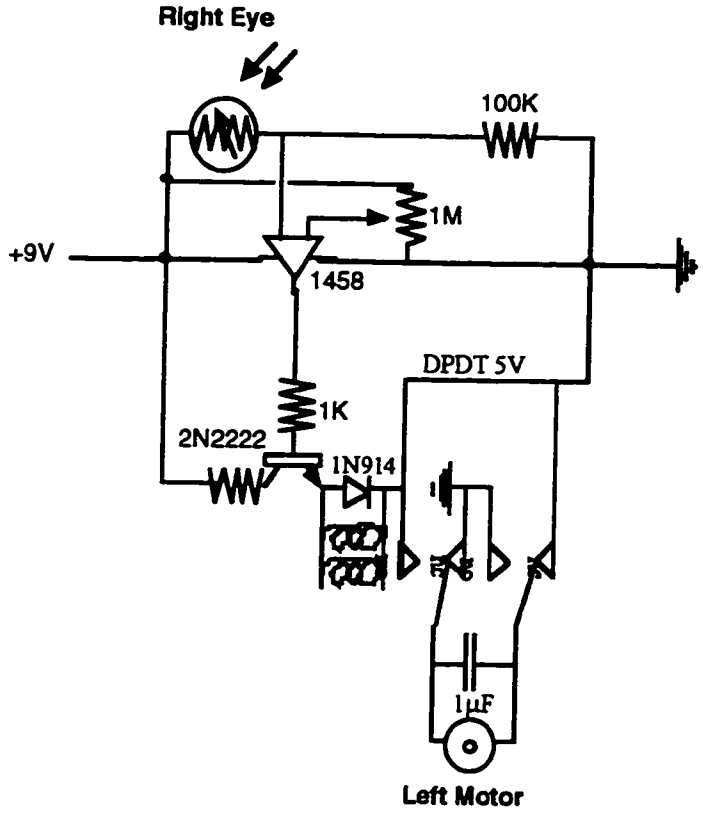


Figure 5-9 Half of Light Seeking RI circuit

5.3.8 Testing the Control Strategy

Several trials were made with the controller in a darkened room with single and multiple light sources. The goal given to the operators was to move toward the light using the SOLENODON vehicle equipped with a manual controller and the light seeking RI. In all cases the human operators were able to regain control from the RI even if the operators did not wish to move toward the light and the RI did.

5.3.9 Multiple RI Coexistence

Once it had been confirmed that the RIs could be constructed, an integration test was performed to ensure that both RIs could exist on the same vehicle at the same time without providing conflicting control information. This test was essential since any realistic control task would require the presence of more than one behaviour producing controller [Brooks 86].

The vehicle was released into a darkened room containing several light sources and various obstacles without its human controller. The vehicle traveled about the arena, moving towards various light sources it detected and avoiding collisions with obstacles it encountered with its whiskers. From this we concluded that the two RI subsystems were compatible and could coexist on a single vehicle

5.3.10 Observations

The second RI is fundamentally different from the first. The collision avoidance RI becomes active when a higher controller has failed to detect an impending collision. It moves away from potential impending damage. The light seeking RI, on the other hand, draws the vehicle toward light, rather than avoiding something it seeks something and becomes active only when it detects it and the higher level controller has failed in doing so. One RI prevents motion towards something, the other prevents motion away from something.

5.4 Conclusion

In this chapter a series of experiments were conducted which were conducted with the SOLENODON IV test-bed vehicle in order to select a reliable, robust and simple RI which could be used to teach a DA. The result of the work in this chapter adds weight to the argument that a reflexive control mechanism can be beneficial when interacting with a far more complex one (a human operator) assisting real time control. The strategy developed in this chapter will be seen

again in Chapter six during the autonomy tests when it is applied to the complete RIDA model.

6. Experimental Design

Ensure training is hard, realistic and of an intensity and duration expected in operations.

CFP 309(3) Chap 17 Pg 17.7 Para 4E.

6.1 Introduction

In this chapter we argue that it is possible to apply the concepts, characteristics and terminology associated with human autonomy to robotic systems. We do this in order to develop a framework for describing and measuring just how autonomous an autonomous agent is.

Within this framework we examine the question of autonomous mobility and develop a set of experimental tests which allow us to determine if a robotic system is, in fact, autonomous with respect to mobility. The framework and tests will then be applied to the RIDA architecture in the next chapter.

6.2 Linking Autonomy and Robotics

Assuming this notion has merit, there is relevant discourse in biology (Autopoiesis) defining and describing "living" beings in terms of their mechanical components, their interactions and relationships [Varela 79]. Machines and biology have been, and continue to be, closely associated. From the zoological figures present in astronomical formations to models of flight to present-day discussions concerning "thinking computers" and the computing brain, runs the compelling notion that the mechanism of living things is inextricably bound to what we know as life.

Often, the mechanical implementation of an entity is used to dismiss any notion of animal life being autonomous. The assertion is made that because of the mechanical nature of biological systems, there are no purposes in animal nature

[Bonner 80]. Its apparent purpose is similar to the purposes of machines--they do what they are designed to do. So, if machines are not the same as living creatures, why is it that the comparison is so easily made?

The mechanisms that define a machine like a mobile robot as an entity, and determine the nature of interactions and transformations it may undergo, we call the organization or architecture of the machine. The actual relationships that hold between the components that constitute a particular machine is its structure. The organization of a machine (or system) does not specify what components should be used to construct the machine as a system; it only specifies the processes and behaviours that these must generate to constitute the machine [Braitenberg 86]. Therefore, the organization of a machine is independent of the properties of its components, which are essentially arbitrary, and a given machine can be realized in many different ways by many different kinds of components. [Varela 79]. The validity of this argument can be seen in many places. Few people think of the individual components of an automobile when they think of the concept of "car" yet they know the functionality that they associate with being "car-like".

Biological systems are also defined by their mechanisms and are therefore machines of a particular class. The argument commonly made is that a living entity is defined by its organization independent of its structure or the material that physically embodies it. You are more than what you eat, or more accurately, you are independent of what you eat. In short, a biological life form is defined not by its components but by their interaction.

Any biological system can be treated in terms of the properties of its actual components as a physical system. There is no limitation whatsoever on doing so, except for the number of variables that one might have to consider. But this is only a complexity problem--all be it a profound one. Eventually, one should be able to have a physical description as accurate as required of any biological system. Of course, some biological systems are much easier to describe than others--an

amoebae versus a kitten for example. Weather this is, in fact, possible to fully realize is still a matter of some debate and is not necessarily relevant to this argument.

Since humans are biological systems, and by extension, physical systems this measure could be applied to humanity as well. This is a very important link as it at least allows us to argue that autonomy--as a philosophical concept--is applicable to at least one describable physical system and, by extension, should at least be examined in other physical systems.

Therefore, it should be possible to examine the autonomy of a mobile robot in terms of the autonomy of a human. While mobile robots may fail this examination--as animals might, they must still be examined using the same criteria.

Autopoiesis does not specifically address the concept of autonomy but it creates an essential link between mechanical and biological systems, at least at the descriptive level. Although machines are not commonly thought of as alive, it is possible to extend the notion of the autonomy of a human to that of the autonomy of a machine.

6.3 The Autonomy Framework

It is important to note at this point that there exists no framework for measuring autonomy in robotic systems. Typically, measurement is very task oriented, where tasks are selected based on a particular skill set rather than a set of characteristics. The vehicle either succeeds in the tasks or it fails. While tests may be duplicated between individual robots, there is no set of common terminology or means for making comparisons--even at the conceptual level. We propose to build some common understanding based on human autonomy concepts. From the preceding

discussion and our literature review of chapter 2 we can form the following framework for the classification of autonomous agents.

Autonomy can be examined on at least two levels--described as first and second order autonomy. When we refer to an agent exhibiting first order autonomous behaviour we refer to an agent's ability to make individual choices on its own. If these choices are made by some internal process of reasoning then we can refer to this as autonomy of judgment. If the reasoning is carried out with the assistance or interference of outside agents then there exists a dependence which is a restriction on an agent's autonomy from which one could argue this makes the agent less autonomous then one in which reasoning is carried out completely independently. This is somewhat problematic as the means for reasoning can often influence the results of the reasoning and the means were provided by an external agent. Efficacy, or Autonomy of action refers to an agents ability to actually carry out decisions it makes internally. Restrictions on autonomy may be affected by internal deficiencies or external limitations. Since a simulated agent cannot actually carry out their decisions they are less autonomous than instantiated agents because they lack efficacy.

If the nature of the decision making changes over time then the agent is said to "learn". This allows a more flexible agent and therefore one which is more autonomous than a non-learning agent. If the nature of the decisions made substantially change through internal deliberation then a first order autonomous agent is also second order autonomous agent. This implies that previous decisions are examined in totality and decision made based on their combined efficacy rather than on an individual basis.

While this framework is not complete it will serve to describe and illustrate the aspects of autonomy which the RIDA architecture can address. This framework in itself is a significant theoretical contribution as this type of linkage is not assumed in research today. One need only examine the controversy surrounding reactive

control vs. model building. There has been no means for promoting discourse in the area. One might not agree with the notion that most reactive systems address first order autonomy problems but at least there is some common ground for discussion.

6.4 Placing RIDA in the autonomy framework

We claim that the RIDA architecture supports the construction of first-order autonomous learning agents restricted in efficacy only by their ability to interact with their environments. While RIDA may support second order autonomy, no evidence is presented here to support this claim.

6.5 Tasks for measuring RIDA within the framework

Five separate tasks have been devised to provide confirmation of our claim with respect to the low level spatial mobility problem [Rashotte 85] in which an animal--in rashotte's case--is observed moving around in its environment and metrics applied to its ability to do so successfully. While Rashotte focused on optimal behaviour we will consider only the effectiveness of an agent's behaviour in low level tasks.

Each task is provided to demonstrate a different aspect of autonomous behaviour. Taken together, the tasks support our claim with respect to spatial mobility.

The tasks are;

- Task 0: Reflexive avoidance of objects
- Task 1: Learned avoidance of objects
- Task 2: Learned stimulus seeking
- Task 3: Learned behaviour change
- Task 4: Cascading RIDA control hierarchy within the first-order limitation

6.5.1 Task 0:

The simplest of all spatial tasks is the avoidance of objects. This is accomplished through a detection-action sequence where the object is perceived and an effort made to deliberately move away, stop or take other action.

The task has several characteristics associated with it:

- ***Single Behaviour:*** The majority of research which has been conducted with autonomous vehicles has concentrated on developing a robot which is capable of exhibiting only a single type of behaviour supporting a specific task: that of avoiding collisions. Clearly this is a standard which must be addressed.
- ***Reflex Only:*** A rigid relationship exists between the stimulus of perceiving a wall with a sensor and the reaction to it which is moving away from the collision. In biological systems, a reflex response provides an animal, including humans, with protective behaviour. Such responses have been shown to be present in animals which have been isolated from birth and are thus considered instinctive. Reflex responses are elicited independent of environmental factors.
- ***Instinctive Reactive Response:*** Instinctive reflexive behaviours have been shown to be quite useful in a number of species;

> A female digger-wasp emerges from her underground pupa in spring. Her parents died the previous summer. She has to mate with a male wasp and then perform a whole series of complex patterns connected with digging out a nest hole, constructing cells within it, hunting and killing prey such as caterpillars, provisioning the cells with the prey, laying eggs and finally sealing up the cells. All of this must be completed within a few weeks, after which the wasp dies. [Manning 79]

The diagram below illustrates a vehicle exhibiting a very simple reflex response to a collision stimulus.

Figure 6-1 Reflexive Collision Avoidance

6.5.2 Task 1

While reflex is useful, learning can be even more so. Take for example the development of lions;

> Born quite helpless, it is sheltered and fed by its mother until it can move around. It is gradually introduced to solid food and gains agility in playing with its litter mates. It has constant opportunities to watch and copy its parents and other members of the group as they stalk and capture prey. It may catch its first small live prey when 6 months old, but it is 2 years or more before it has grown sufficiently to feed itself. Its behaviors, and particularly the methods and stratagems it uses in hunting, may change according to circumstances throughout its life. [Manning 79]

Like task 0, Task 1 involves the avoidance of collision but in this instance the difficulty of the task is increased by forcing a learning component. While many creatures go through their entire lives being dominated by reflexes, once learning is introduced a wider of ranges of response is possible giving at least the promise of adaptation to unexpected circumstances.

Like the previous task, this one also has certain characteristics:

- ***Single Behaviour:*** While the behaviour is generated in a more complex way, it is still obstacle avoidance.
- ***Reflexive and Learned Response Coexist:*** Instead of having only a single mechanism for obstacle detection and avoidance, two mechanisms are available to the vehicle. In fact, the vehicle could "learn" to avoid obstacles through their interaction. If the vehicle can learn to avoid obstacles, it is arguably better suited to a changing environment.

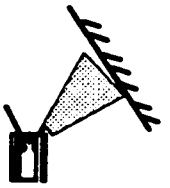


Figure 6-2 Learned Obstacle Avoidance

- ***Simple Decision Making supporting an argument for first order autonomy:*** When the vehicle has learned to avoid obstacles by acting on a stimulus, it can be said to have at least some low level form of enhanced first-order autonomy as it is making decisions to avoid obstacles rather than simply acting in a reflexive manner. While the response to an obstacle may be rather simple, the vehicle has "decided" to do this on its own.

6.5.3 Task 2

Having learned a single task, it is necessary to experiment with learning a second, unrelated task. Learning, one thing is rather limiting to say the least. In task 2 we simulate feeding. Feeding is considered the successful locating, tracking and contacting a source of sustenance. In the case of the arena this sustenance is supplied by the light source.

In the vehicle interaction with the light, it must somehow learn to associate it with "good". At the same time the vehicle cannot abandon its previously learned skill obstacle avoidance.

The characteristics of this task are:

- ***Multiple Tasks with consistent goal:*** Task 0 and 1 both concentrated on obstacle avoidance, this task adds light tracking/feeding. The vehicle is responsible for the selection of appropriate response independent of external influence.
- ***Diminishing Reflexive and Sustained Learned Response:*** As with task 2, this task requires learning as well as inherent reflexive behaviour. Unlike the previous task however, finding and moving towards the stimulus is considered good and would somehow be rewarded, where in task 0 and 1 moving towards a wall resulted in some form of "bad" resulting in either a punishing collision or negative reinforcement of some form.

We can observe this type of positive learning in infant children. They have reflex responses which causes them to suck on objects. This is very useful when feeding. Eventually feeding becomes its own reward and the reflex response disappears.

Selection of Choice of Actions: With more than one skill, the vehicle now must select between behaviours. This is because the intense light source is near a potential obstacle–a wall. Demonstrating this ability increases the autonomy of the vehicle since only secondary external influences were used to make the autonomous choice possible.

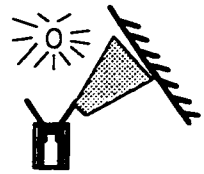


Figure 6-3 Learned collision avoidance and light seeking

6.5.4 Task 3

In task 2 the vehicle learned to follow light. Task 3 consists of modifying this response. The intent is to demonstrate the state of satiation. This might occur after the vehicle were near the light for some period of time and now other--more pressing needs--might take over. Again, some form of compelling negative stimulus could be used to attempt to force a change in what the vehicle considers "good". The vehicle must be encouraged to move away from the light even though it has presumably received positive reinforcement when it first found the light.

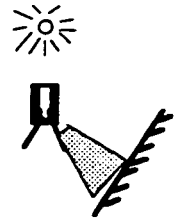


Figure 6-4 Learned collision and light avoidance

The characteristics of this task are:

- ***Multiple Tasks with changing goal:*** In this task a previously learned behaviour is overridden with the help of negative reinforcement. This is similar to the behaviour we see in most animals--humans for example. When we are hungry, the food we eat may taste very good--encouraging us to eat more. When we have eaten too much, we begin to feel uncomfortable, perhaps with nausea. At this point we have received negative reinforcement and stop eating. In either case we were not in direct control of the reinforcement mechanism yet it profoundly affected our behaviour.

The same must be true of the vehicle. Initially attracted by light it will be repelled by it in the end.

- ***Diminishing Reflexive with Sustained Learned Response:*** Similar to task 2
- ***Selection of Choice of Actions:*** Similar to task 2

6.5.5 Task 4

This task is rather different then the preceding four. There is a requirement that there be some mechanism to promote an architecture whose principles are reusable as the number of interacting control mechanisms function. This is essentially to address the cascading control aspects of RIDA.

As more and more mechanisms interact, their interactions can become quite complex. For example, consider the low-level autonomous robot "Herbert" developed at MIT [Brooks 89]. The vehicle could move around an office picking up coffee cups and returning them to a central room--however it could only do this successfully once, as the complex interactions of its various controllers made it very prone to failure.

Task four requires the application of the control principles in a specific architecture claiming to promote first order autonomous behaviour to be applied in more than a single instance. For example, if a controller learned to avoid obstacles, could it be used to teach another controller to do the same and possibly better?

6.6 Conclusion

In this chapter we have linked the notion of human autonomy to that of autonomous behaviour exhibited by mobile robots. A careful literature examination has shown that this approach has never been applied before, and has lead to significant disparities in how investigators conduct experiments, gather results and report them. By forming a bridge between the disciplines we are able to

utilize discipline and implementation independent terminology which, for the first time, provides a rudimentary set of common metrics.

Because of this linkage, we are free to explore the nature of the tasks which would demonstrate a level of autonomy in a robot based on associated psychological and philosophical thinking. We described a framework for measuring an autonomous vehicle's "autonomy" and from this framework we developed a set of tests which could be applied to a vehicle conforming to the RIDA architecture. In essence, we have described a method for validating claims made concerning autonomy in mobile agents.

7. Testing the RIDA Architecture against Autonomy Tasks

Profanity is the one language that all programmers understand.

Anon.

7.1 Introduction

In this chapter we present the experimental results of several tests run against the complete RIDA architecture in simulation. The chapter begins with a brief description of the simulation environment, followed by a section for each of the five tests conducted within the Autonomy framework described in chapter five. Within each section, the test vehicle and arena configuration, test results and observations will be discussed with reference to the RIDA architecture and how the architecture supports each autonomy task.

7.2 The Simulator and Playback Modules

The simulator and playback modules which instantiate the RIDA architecture and provide the arena environment, were written in the C programming language and support several different vehicle designs employing the RIDA control mechanism.

7.2.1 Experimental Environment

In order to make valid comparisons between individual performances of desperate vehicles it is essential to do so in a consistent environment. The environment selected was relatively simple yet provided certain characteristics which promoted the testing of autonomy characteristics,

- Multiple stimuli: A light source provides one of these and walls provide appropriate surfaces for tasks such as collision avoidance.

- Ease of Construction: The facility itself is rather straightforward to construct and can be easily replicated in many environments with limited material effort.
- Multiple Chambers: The central wall provides the ability to shield one side of the arena from the other thus promoting the testing of goal seeking.

Investigators involved in mobile robotics are often criticized for experimentation in unrealistic and simulated environments [Brooks 86] and, indeed, the arena is one such environment. However, the arena has been designed to allow useful measurement. Since its boundaries are fixed, comparisons can be made and conclusions drawn from comparisons of several different trials of different vehicles. Results are thus replicable and can be used to measure other vehicles in a meaningful way.

While an environment such as a lab might provide a slightly more "natural" setting, objective measurement becomes problematic as boundaries are ever changing.

7.2.2 The Simulator

The simulation environment consists of several logical sub-components;

- The *simulator engine* which interacts with the Deliberate Apprentice and arena generator components by responding as a vehicle would employing its own Reflexive Instructor.
- The *arena generator* is responsible for maintaining the arena image on the screen, and generating new views as time passes, commands are implemented and telemetry information is received.
- The *Deliberate Apprentice component* interacts with the arena generator in a "black box" mode. The DA accepts the sensor input from the arena, makes

decisions and issues commands. It receives reinforcement from the simulator engine.

The relationship between these components is shown in the diagram below;

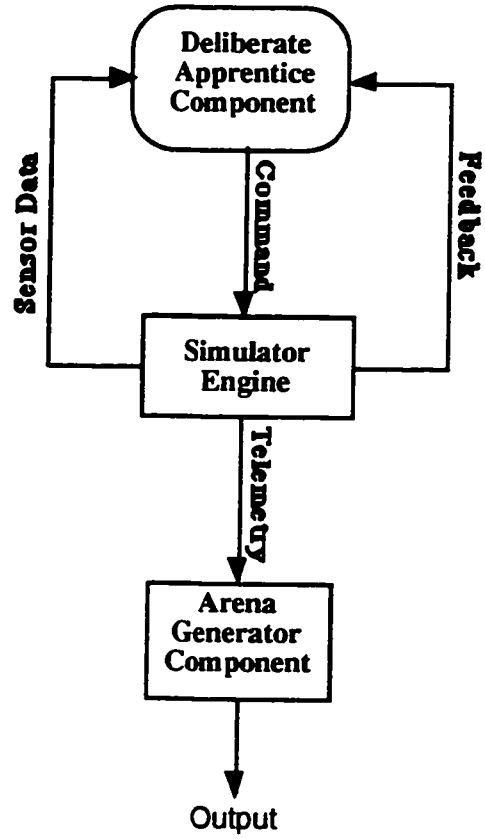


Figure 7-1 Simulator Architecture

While each vehicle was separately designed for the individual low-level autonomy tasks, they shared sensing and mobility characteristics. In any time step, the vehicle could move one "step" in any of the defined directions [Brooks 86]. The vehicle could not reverse, this restriction was placed on all vehicles as no sensors faced backwards.

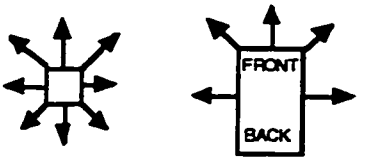


Figure 7-2 Permitted Movement

For the purposes of the simulation sensors were assumed to be perfect and no actuator dynamics were included.

7.2.3 The Sensor Array

Several sets of sensors were simulated based on the characteristics of data obtained from various manufacturers and empirically through the EMMA experiments (appendix C).

The basic set of sensors are the whiskers used to detect collisions. These were designed to attach to micro-switches whose signal would be filtered through a de-bounce circuit and sent to the controller. These sensors were used in pairs.

All light sensors were assumed to be shielded photo-resistors similar to those employed in the EMMA experiments. It was found that these devices provide a basic linear response to light when relatively close to a light source. This linear response can be extended to several feet depending upon ambient light conditions, the quality of the photo-electric sensor and the intensity of the light source. Their response is reduced exponentially as they are placed beyond a certain threshold.

The sonar employed shares the characteristics of shielded Polaroid sonar sensors--being able to detect and accurately measure distances of objects from between 6 inches to 30 feet. This range is broken into 3 discrete distances which are sent back to the controller by a simple quantizer circuit. These signals throughout these trials were near (from 0 feet to 1 step), middle (from 1 to 5 steps), and far (beyond 5 steps).

7.2.4 The Playback Module

The playback module was created to replay and demonstrate the motion of the vehicles without the need for the simulator itself. Essentially, the playback module reads vehicle telemetry information files and produces the appropriate vehicle positioning in the arena and telemetry read outs. This is shown in the diagram below. The L's refer to light sensors while the S's refer to sonar.

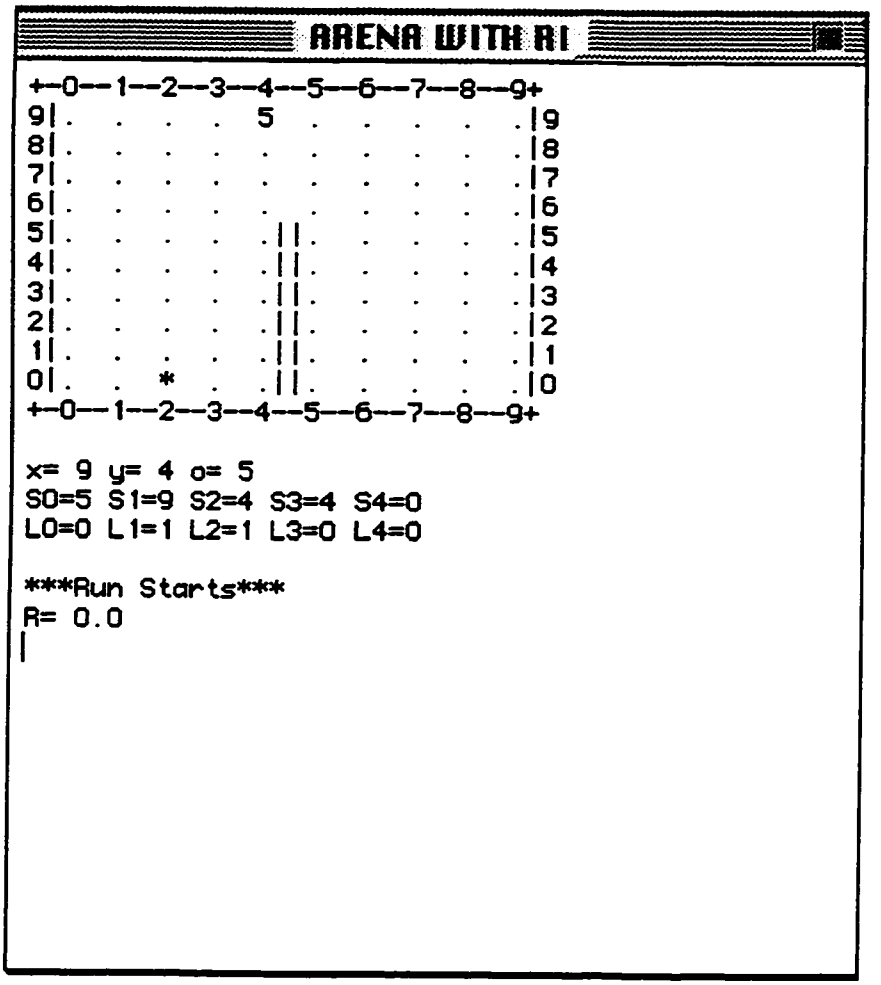


Figure 7-3 Screen shot of simulator running

7.3 Task 0: Reflexive avoidance of objects

7.3.1 Description of Task

The walls of the arena were employed as obstacles. A vehicle placed in the arena had the task of avoiding collision with any of its surfaces.

Task 0 is the implementation of reflexive avoidance of objects. This implies that the behaviour is specifically designed and built into the vehicle and becomes an inherent characteristic of it. Many examples of this behaviour mechanism can be found in nature ranging from the knee jerk reaction in humans to the escape mechanism of various insects. In addition, variations on this task are quite common in the literature [Brooks 86][Fagg et al. 94].

As no DA was required for this test a simple RI was implemented. The function of the RI can be described as "If a collision is detected, turn slightly away from where it happened." which is essentially the RI proposed in chapter 4.

7.3.2 Description of Vehicle

A vehicle was devised employing a single RI component shown in figure 7-4.

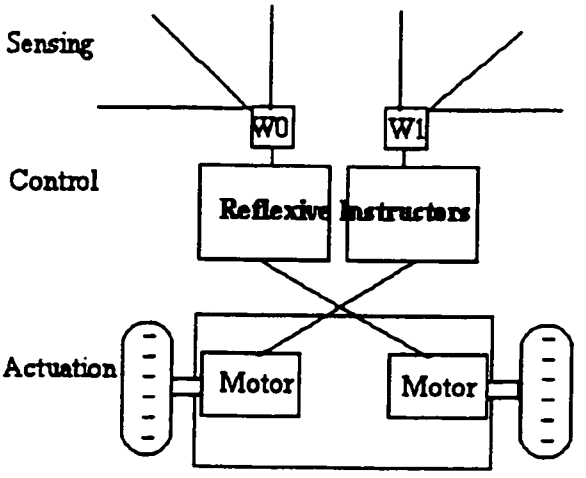


Figure 7-4 Specific configuration for test 0. The vehicle is equipped to react to the environment but not to learn from it.

Sensing was accomplished via twin whisker sensors and the RI controller was similar in design to that employed in EMMA II.

7.3.3 Discussion of Results

Four separate measured trials were made with this vehicle. The path of the vehicle for the second run is shown in figure 7-5 and is typical of the result. Note the vehicle reacts to collisions, in many cases, by continuing to collide. (Each dot on the perimeter of the diagram represents a collision)

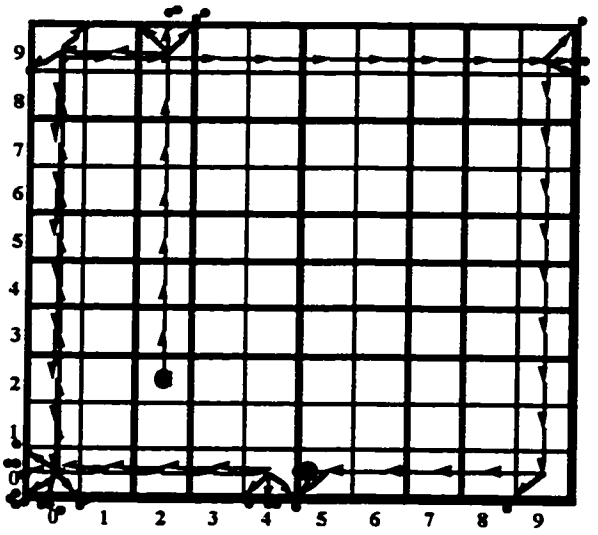


Figure 7-5 Trial run for test 0.

In this trial the vehicle is started at row 2 column 2 and orientation 0. At the end of the trial the vehicle had collided with walls 37 times out of 100 time steps allocated for the test, with collisions concentrated at the corners of the arena.

The vehicle simply continued in a straight line path until it collided. The RI caused it to deflect slightly right or left and then the vehicle attempted to move further. This rather naive approach to collisions was successful to the extent that the vehicle could find a new unobstructed path after every collision. However, the

vehicle spent considerable time attempting to clear its way past a collision. This is very similar to what occurred in the EMMA experiments and is quite common among other autonomous systems employing similar reflexive mechanisms [Zapata et al. 93] and is often cited as a weakness of purely reactive control systems [Green 93].

Figure 7-6 reflects this characteristic as collisions continue to occur as the simulation progressed. This is not surprising since the controller has no mechanism for learning from past events. To avoid collisions in the future. The graph measures the rate of collisions per 10 time steps. For a vehicle to improve its performance, its collision rate should drop over succeeding 10 step increments (The graph would slope downward). This is not the case with this controller as there is no inherent ability to improve its performance over time

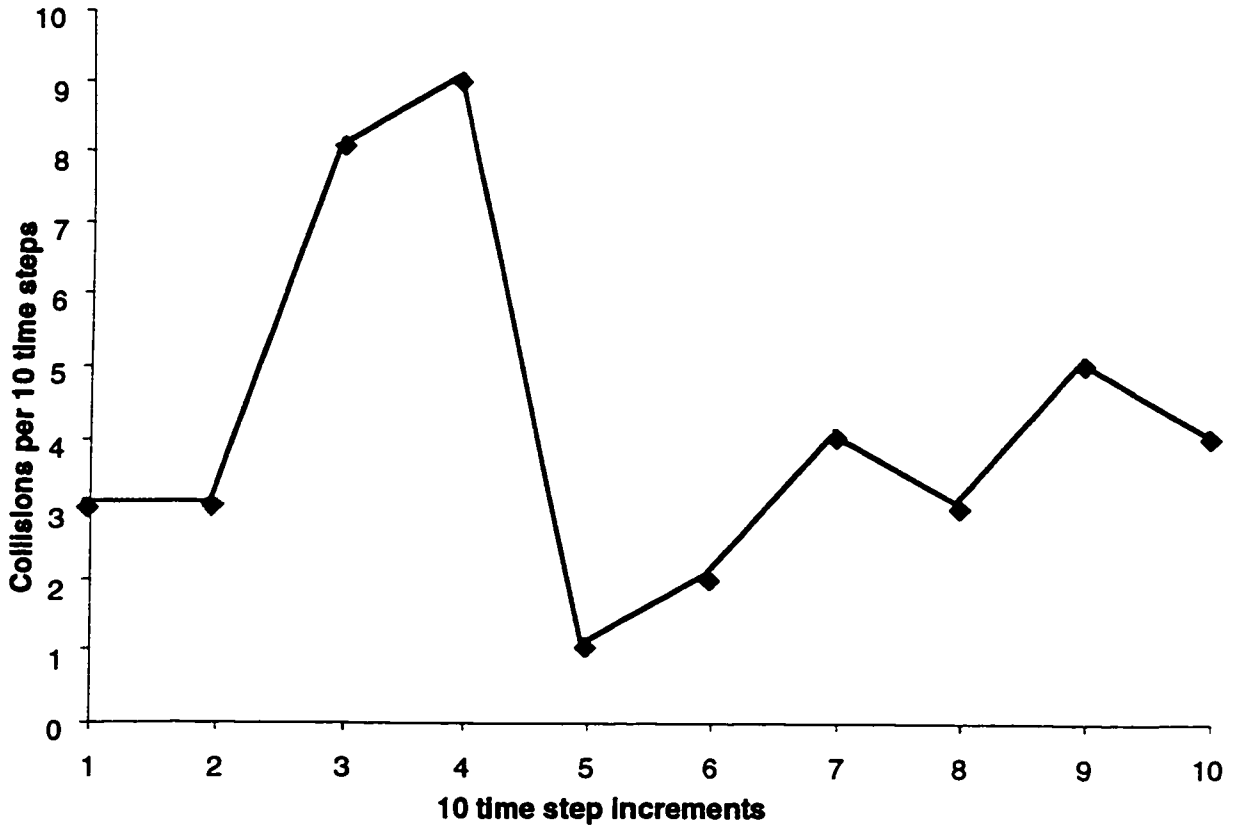


Figure 6 Task 0 showing collisions per 10 time steps.

7.4 Task 1: Learned avoidance of objects

7.4.1 Description of Task

This task is similar to task 0 but employs a much more capable vehicle equipped with the entire RIDA architecture. Again, the obstacles are supplied by the walls of the arena. A vehicle could be said to successfully complete this task if it produced substantially fewer collisions than that of a vehicle employed in task 0. This would confirm that learning had taken place. Learned response is the focus of much literature including [Fagg et al. 94][Najand et al. 92][Nehmzow et al. 93]. The focus of learning in this task was the transferal of the inherent skill provided by the RI to the much more capable DA.

7.4.2 Description of the Vehicle

The vehicle used for task 1 is shown in figure 7-7 below. The vehicle is equipped with touch sensors connected to the RI and sonar which is connected to the DA. At first the DA did not know how to utilize the sonar sensors as this ability was never provided to it. The RI again employs twin sets of whiskers to detect collisions. In addition, a set of 5 sonar detectors were added and connected to the DA. The DA in this case employed a modified rapid reinforcement neural network architecture [Fagg et al. 94] as described in chapter 2. The intent of the design was to promote learning of the neural network-based DA from the RI through negative reinforcement generated by the RI as it detects the obstacles.

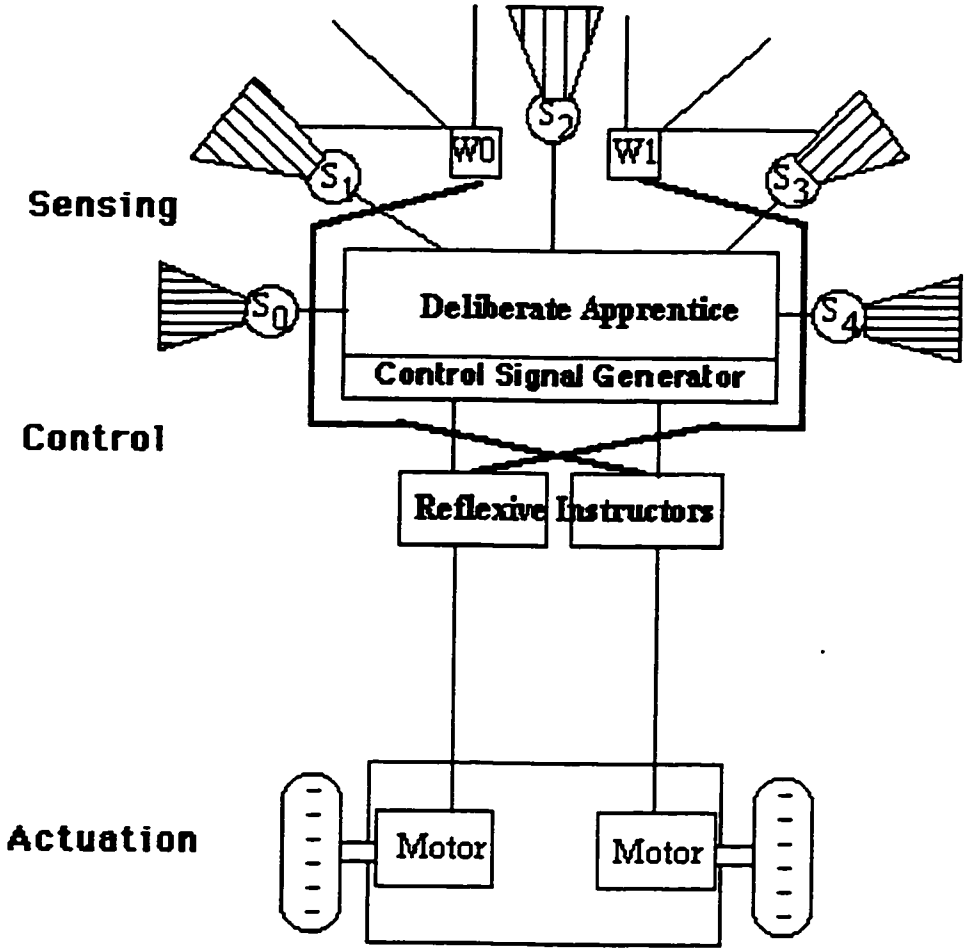


Figure 7-7 Specific configuration for test 1.

7.4.3 Discussion of Results

Figure 7-8 shows the results of the second of four parallel recorded trial runs conducted employing a RIDA equipped vehicle (on the left) and a vehicle equipped with only a DA (on the right). The diagram on the right shows the path of the non-RIDA vehicle. The path of the vehicle clearly exhibits the collision behaviour shown by vehicle 0 with no indication of improved performance over time. The diagram on the left illustrates the improved performance of the RIDA vehicle which initially collides with the walls of the arena but than quickly learns to avoid them by employing the sonar sensors.

At the end of the trial the RIDA equipped vehicle had collided with walls 11 times out of 100 time steps allocated for the trial showing a marked improvement over test 0. The test also confirms that the DA learns more quickly when supported by an effective RI. Note that no further collisions occurred during the last third of the trial. In contrast the simple reinforcement network equipped vehicle collided 64 times and continued to collide until virtually the end of the trial.

The RIDA vehicle incurred far fewer collisions and managed to traverse more of the actual arena in the same number of time steps than the vehicle on the right which was fettered by a series of collisions which lead to other collisions substantially hindering its progress. Of interest is the pattern that the RIDA vehicle exhibits. One can clearly see that the vehicle has actually discovered a form of wall following as it moves about the arena. The circular motion evident in the lower right quadrant of the left diagram occurred at the end of the trial and is a condition which is inherent in reinforcement learning methods and was also observed in [Fagg et al. 94].

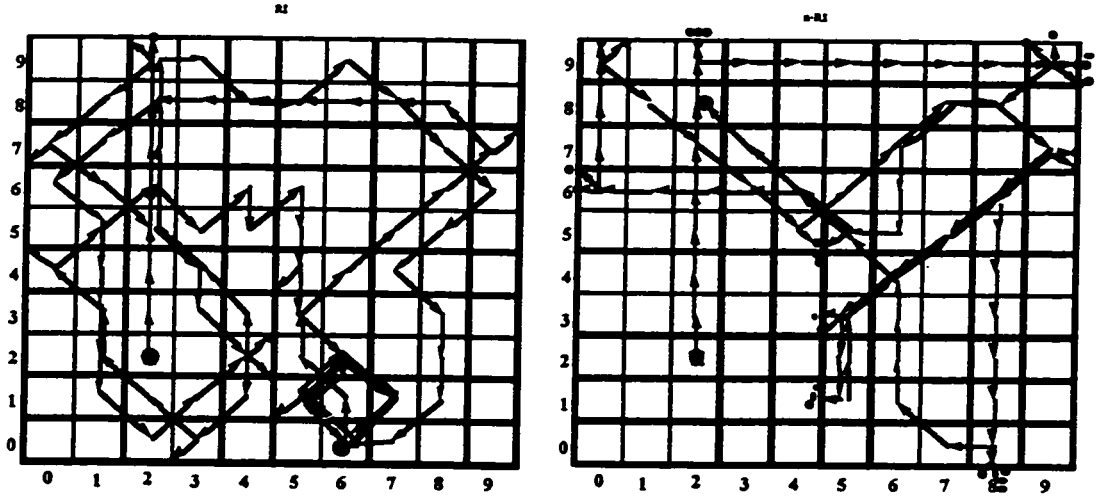


Figure 7-8 Test 1 trial run showing both results (with RI and without).

Figure 7-9 illustrates that after an initial "training set" of collisions, the RIDA controller stopped employing the RI and become wholly reliant on the DA for

guidance. Note that after an initial collision-free run the collision rate increases as the DA learns. After the peak the collision rate quickly falls and finally no further collisions occur. Since the non-RI vehicle continued to collide throughout the trial, clearly the RIDA performance is better.

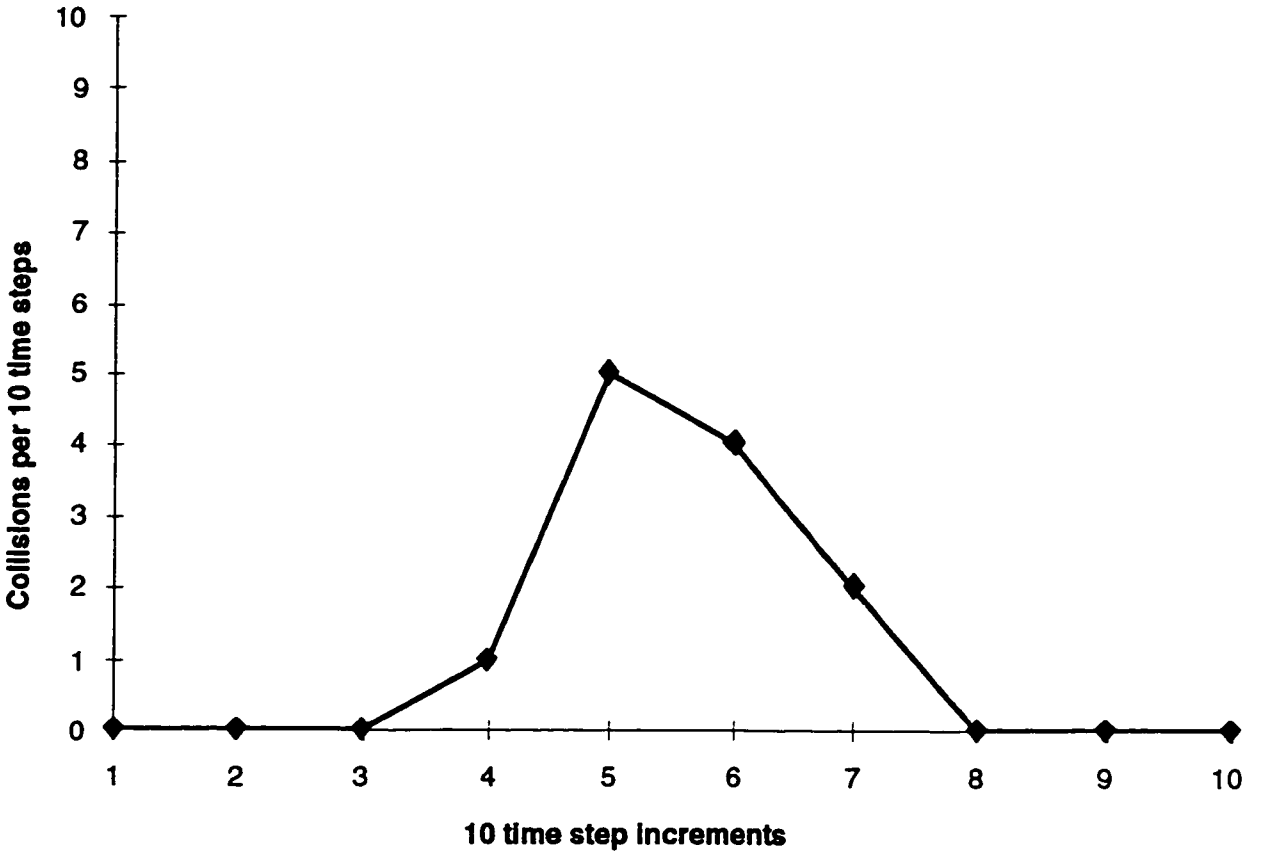


Figure 7-9 Test 1 with RI and DA active.

Of particular interest is the next graph (figure 7-10). The curve labeled with diamonds (RI00) is the same curve plotted in task 0. Since it is incapable of improving its performance, the curve continues to meander up and down as collisions continue. The curve labeled with triangles (NRI10) was produced by the vehicle equipped with only a reinforcement network controller (a DA with no RI). What is interesting to note is that the RIDA vehicle (labeled with squares, RI10) performed better then both of the other vehicles--meaning it incurred fewer collisions. Also of interest, the vehicle employing only reflexes still outperformed

the reinforcement network vehicle, although this performance would eventually be reversed as the reinforcement network learned the task.

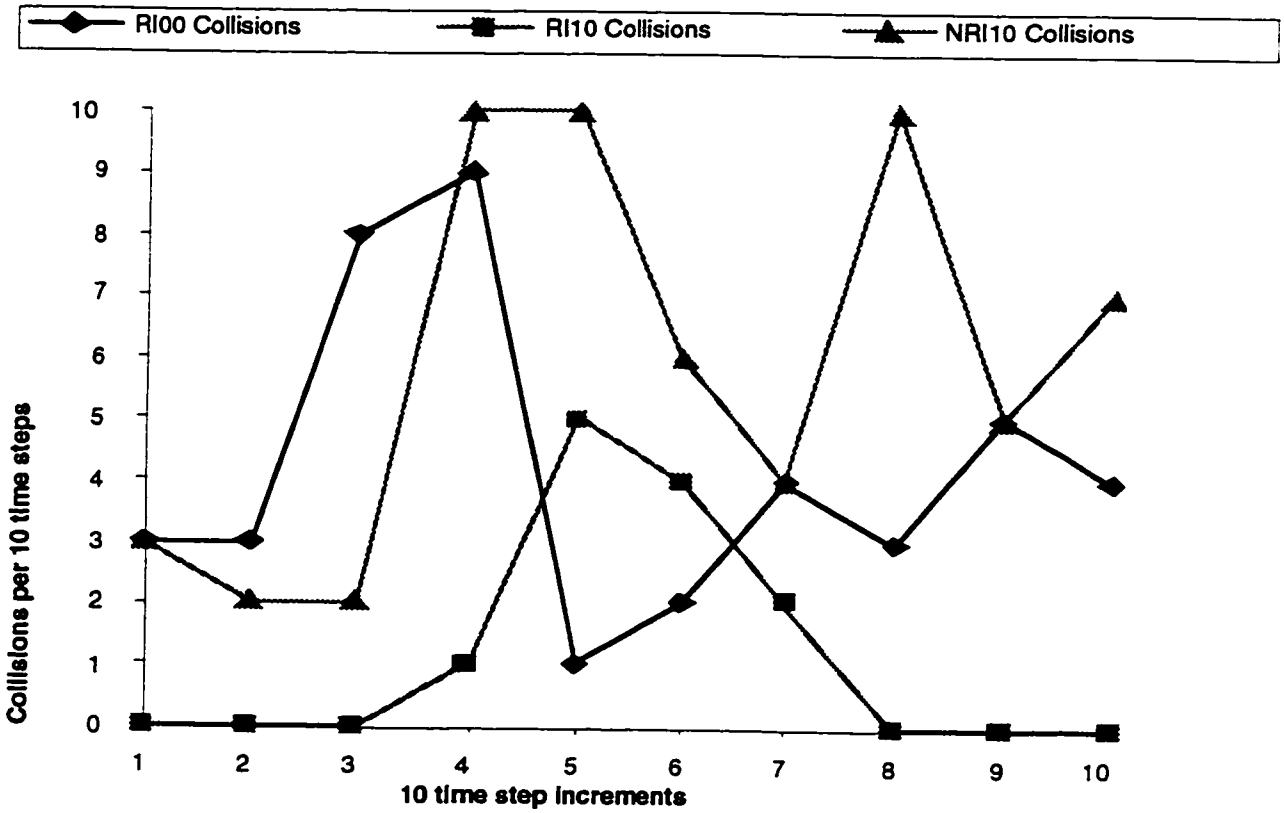


Figure 7-10 Test 0 vs. Test 1 with and without RI.

One can see that the RIDA equipped vehicle was able to

- learn the task quickly, and
- outperformed both of the vehicles employing more traditional designs.

7.5 Task 2: mixed goals--avoid collisions and find the light

7.5.1 Description of Task

Task 2 requires the adoption of two unrelated skills. Having learned a single task (collision avoidance), this task attempts to ensure a degree of scaleability. It forces a control architecture to deal with learning a second, unrelated task.

In this case the unrelated task is the ability to "feed". Feeding is considered the successful locating, tracking and contacting of a light source positioned within the arena (row 2 column 2). The same DA as test 1 is employed to learn this task except that the reinforcement policy was altered to employ positive reinforcement if the vehicle moved toward the light of its own accord .

The light-seeking RI receives its input from two out-lying light sensors. When one of these two sensors detects a light source it "jogs" the vehicle in that direction. If the DA had already sent control information to accomplish this then the RI's signal is irrelevant since it matches that of the DA and it is effectively ignored. If the DA attempts to move away from the light the RI takes over, jogs the vehicle, and sends a negative reinforcement signal to the DA.

Again, the vehicle incorporates a collision RI employing negative reinforcement.

7.5.2 Description of the Vehicle

Figure 7-11 shows the next vehicle configured for task 2. Being very similar to the previous vehicle, this one adds light sensing to its list of capabilities--using and array of photo diodes. The five light sensors were mounted in the same positions as the sonar giving the same angular coverage. The DA remained the same as task 1 but an additional RI was added as described above. In this case the RI reflexively draws the vehicle to the light. It is important to note that if the DA persists in attempting to move away from the light it can since the RI only jogs the vehicle controls. This ensures that the vehicle is not locked into a simple reactive control policy.

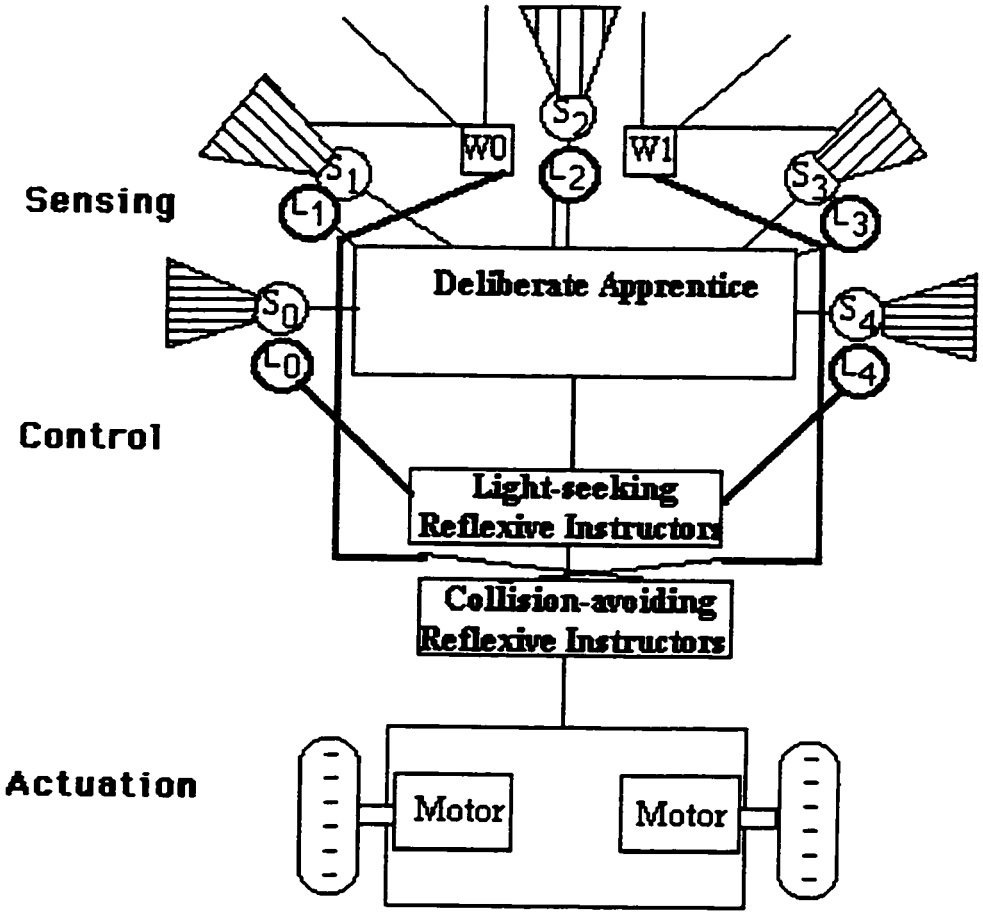


Figure 7-11 Vehicle Specific configuration for test 2.

7.5.3 Discussion of Results

Five runs were completed from different starting locations in order to facilitate the DA's learning. In the trial results shown in the table 7-1, the vehicle was started first at row 3 column 2 and orientation 7. Once the light was found the vehicle was placed at (2,7,2), (8,6,4), (3,8,3), and (0,9,7). In parallel, a second vehicle without benefit of an RI was run at the same time as the RIDA vehicle. The "#Time Steps" columns indicate the length of time it took the vehicle to find the goal while the "collisions" columns indicate the absolute number of collisions experienced during that time. The rates of collision are provided as well. One can see that the RIDA vehicle found the goal faster in all but the fourth case and in all cases its collision rate was lower.

	non-RIDA Vehicle			*RIDA Vehicle*		
Start	#Time Steps	Collisions	Collisions/time step	#Time Steps	Collisions	Collisions/time step
(3,2,7)	*22*	*7*	0.32	10	0	0.00
(2,7,2)	*198*	*60*	0.33	212	57	0.26
(8,6,4)	*76*	*8*	0.11	34	1	0.03
(3,8,3)	*379*	*15*	0.04	90	3	0.03
(0,9,7)	*97*	*5*	0.05	65	2	0.03

Table 7-1 Performance of different vehicle configurations

From this one can see that both vehicles experienced a reduction in the number of collisions as the simulation progressed. As in task 1 the RIDA vehicle accomplished the learning of the light-seeking task much faster than the other vehicle.

The Light RI also had a role to play in successfully finding the light source. This is evident if we graph the relationship between the number of times the light RI is activated and the number of times the DA is given positive Reinforcement per 10 time step increments, shown in figure 7-12. As the simulation continued, the number of times that the Light RI (curve labeled with diamonds) became active approached zero while the DA continued to receive positive reinforcement (labeled with squares). This means that the DA had discovered how to find and follow light without the use of the RI.

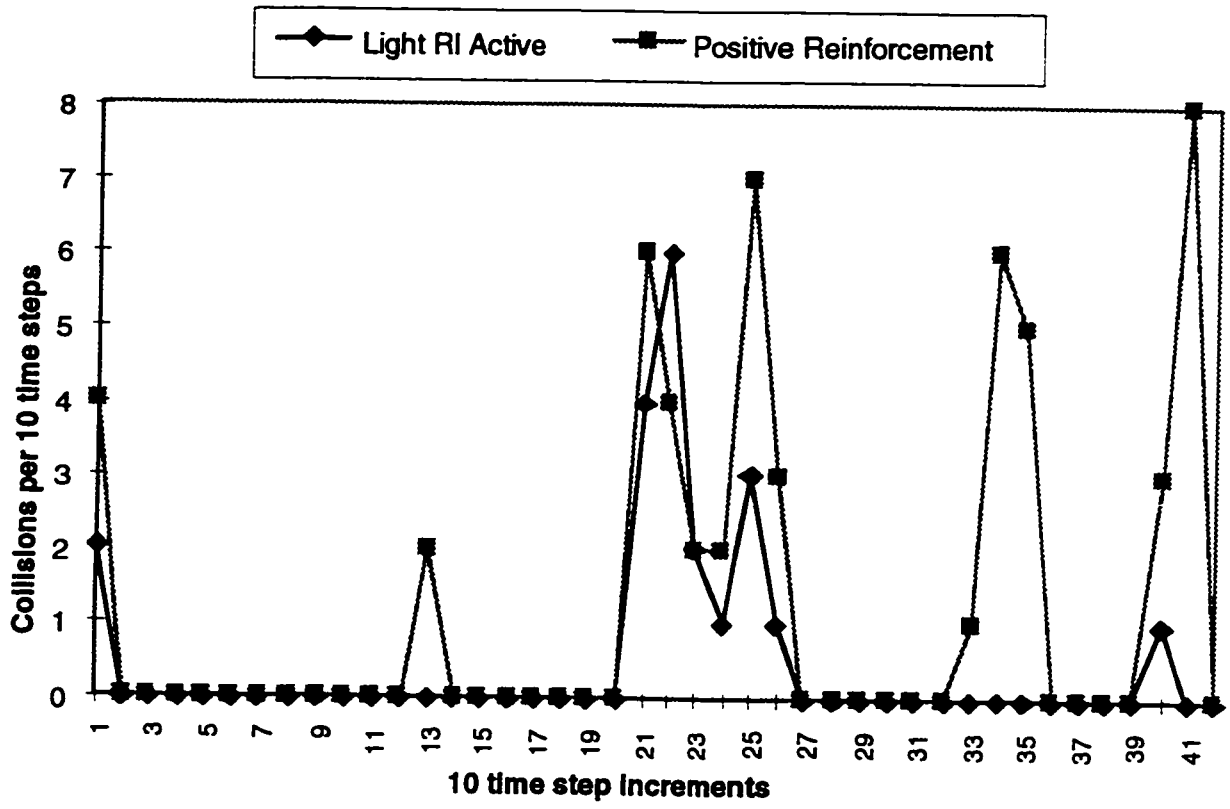


Figure 7-12 Light seeking RI's relationship to positive reinforcement.

The Light RI is initially activated often when the vehicle detects the light source, indicating that a lot of jogging is going on. However, the DA quickly learns to move towards the light and the light RI remains essentially dormant from that point on. The reflex is thus effectively suppressed.

7.6 Task 3: mixed goals--avoid the light and avoid collision

7.6.1 Description of Task

This task is almost the reverse of task 2. Once a vehicle's controller has learned the locating of light and the avoidance of collisions this task provides a slight twist in that one of the tasks is reversed. In this case that task is light seeking which becomes light avoiding. This reversal of behaviour is common and is evident in many systems we use every day. We are attracted to food when we are hungry and

this attraction fades as we become satiated [Sheperd 88]. We actually change our goals from the location of food to other more pressing matters.

7.6.2 Description of the Vehicle

The identical vehicle as task 2 was employed except the RI function was modified to jog the vehicle away from the light source and to punish a DA which attempts to move towards it of its own accord. Positive reinforcement is employed if the vehicle moves away from detected light on its own.

A DA having been trained over 200 time steps learning task 2 was employed in this task.

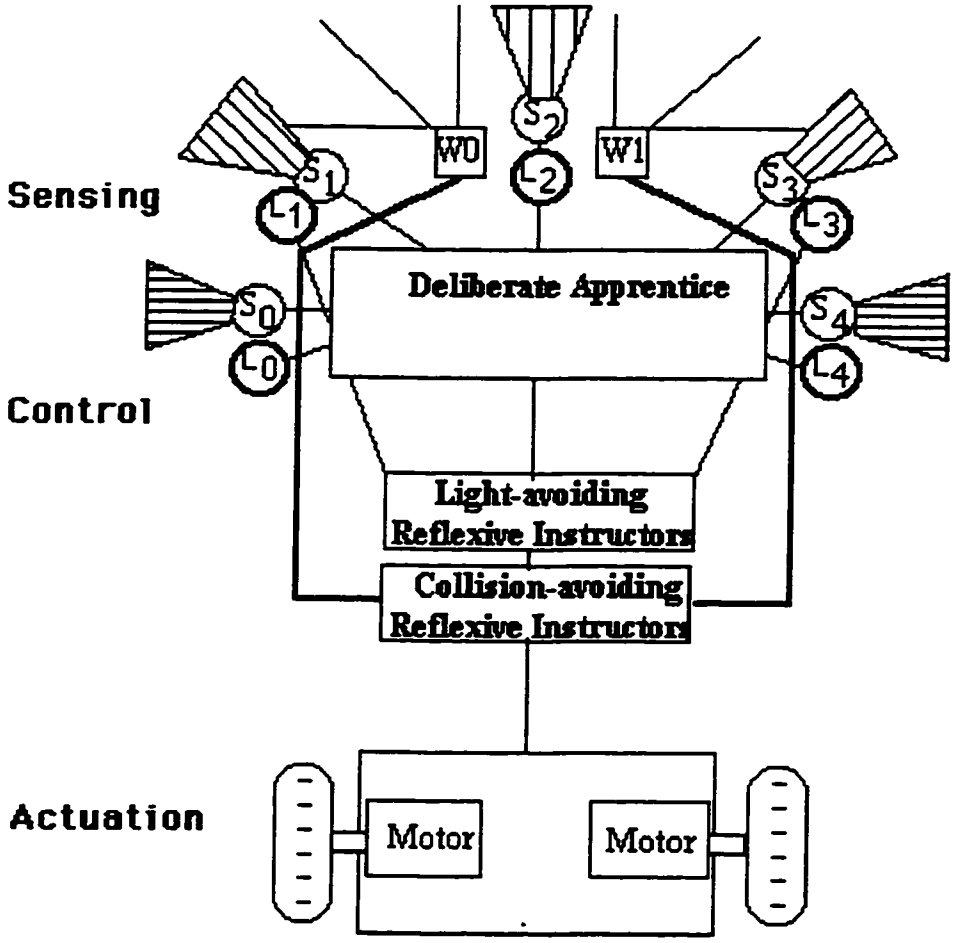


Figure 7-13 Specific vehicle configuration for test 3.

7.6.3 Discussion of Results

Figure 7-14 indicates the performance of a RIDA equipped vehicle deployed in the third trial run of four measured tests. One can see that the rate of positive reinforcement obtained by the RIDA vehicle is approximately twice that of a vehicle learning through simple reinforcement. In addition the rate of activation of the light-avoiding RI dwindles as the trial progresses. As the vehicle moves through the arena it continues to receive positive reinforcement (labeled with triangles) as the vehicle in test 2 did. Note that the collision-RI curve (labeled with diamonds) is quite shallow indicating minimal activation.

It should be stated that this is not what happens in a biological system. We do not like food when we are hungry and then immediately detest food when we are satiated (which is what is being modeled in this test), but the test indicates that the response of seeking and avoiding can be quickly assimilated even if they do not reside in the learning mechanism at the same time.

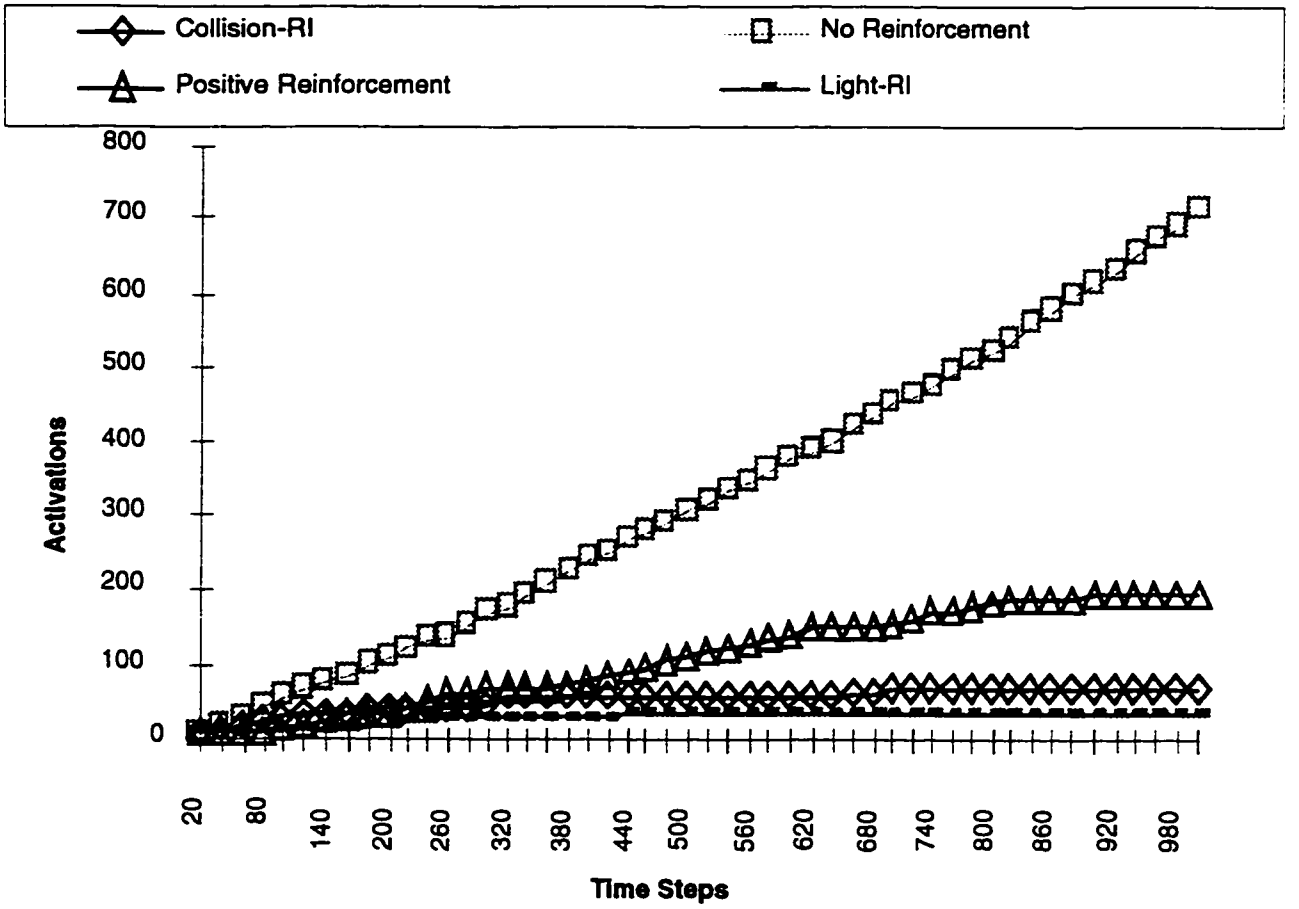


Figure 7-14 Test 3 vehicle performance characteristics with RI active.

7.7 Task 4: Cascading RIDA control: Reliable DA becomes RI to teach new DA

7.7.1 Description of Task

Test 4 was designed to show that an architecture is scaleable within its problem domain. In this case the test is designed to show that once a DA is trained it can form the basis of a very reliable RI and, in turn train another DA.

The results shown are for trial zero of four. The trial was conducted in 2 phases. The first phase consisted of a 200 time step training cycle in which a modified perceptron ANS learned from a reinforcement collision avoidance RI based on the

DA created in Test 2 Trial 1. In addition there was a Low level collision avoidance RI activated on failure of the Upper level RI.

7.7.2 Description of the Vehicle

The vehicle used in this task employed a trained rapid reinforcement network created during test two as higher-level RIs. This RI received sensory input from all the sonar sensors. Its task was to send negative reinforcement signals to the DA when it encountered a collision. In addition, since this RI was able to make use of a richer set of sensor inputs and make decisions based on learning, the RI was able to send a richer signal to the DA consisting of a set of control information which it would have sent to the actuators had it been the DA. This is significant because the new DA can now make use of both the reinforcement signal and a correct response and what was reinforcement learning becomes supervised learning allowing us to select from a wider range of learning methods.

A simple linear perceptron neural network was selected as the new DA for this task. An additional low level RI was employed similar to vehicles 1, 2, and 3. Its task was support the more advanced RI in case of its failure. Whiskers and sonar were the only sensors employed in the test.

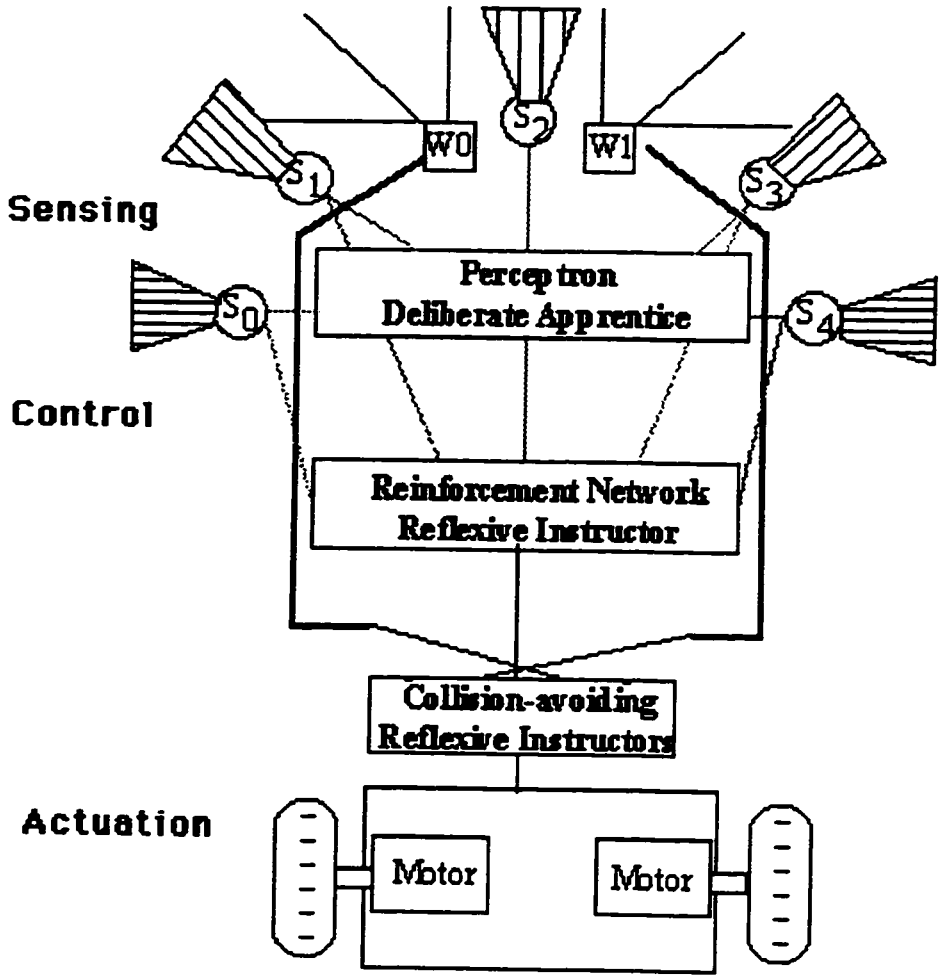


Figure 7-15 Specific vehicle configuration for test 4.

7.7.3 Discussion of Results

Three trials were conducted with very similar results. During the DA training phase, the Reinforcement higher level RI fails in 1 case in the 200 time step allowed for training the new DA. This was successfully compensated for by the lower level RI. The training phase for the first trial is shown in figure 7-16. As the training progressed the perceptron became better at the avoidance task through the training received via the RI components. One can see that after the 120th time step the new DA had learned to avoid collisions. Appendix D describes the implementation of this further.

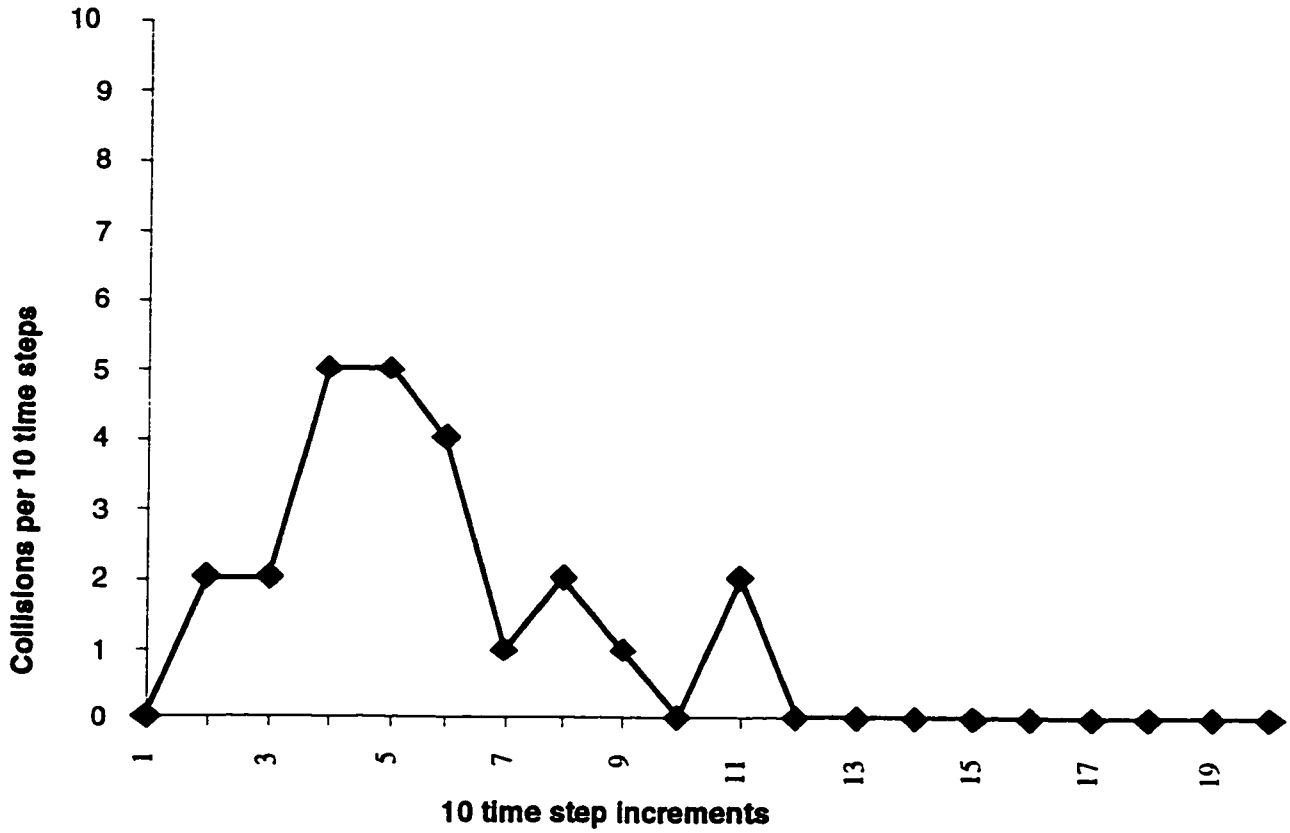


Figure 7-16 Training of Perceptron DA by Reinforcement RI.

During the performance phase of the trial, the Perceptron was allowed to control the vehicle without benefit of any RI. It quickly discovered a collision-free path and repeated it until the end of the 100 step performance phase. Only one collision occurred in this particular trial--when the vehicle was released. This is shown in figure 7-17.

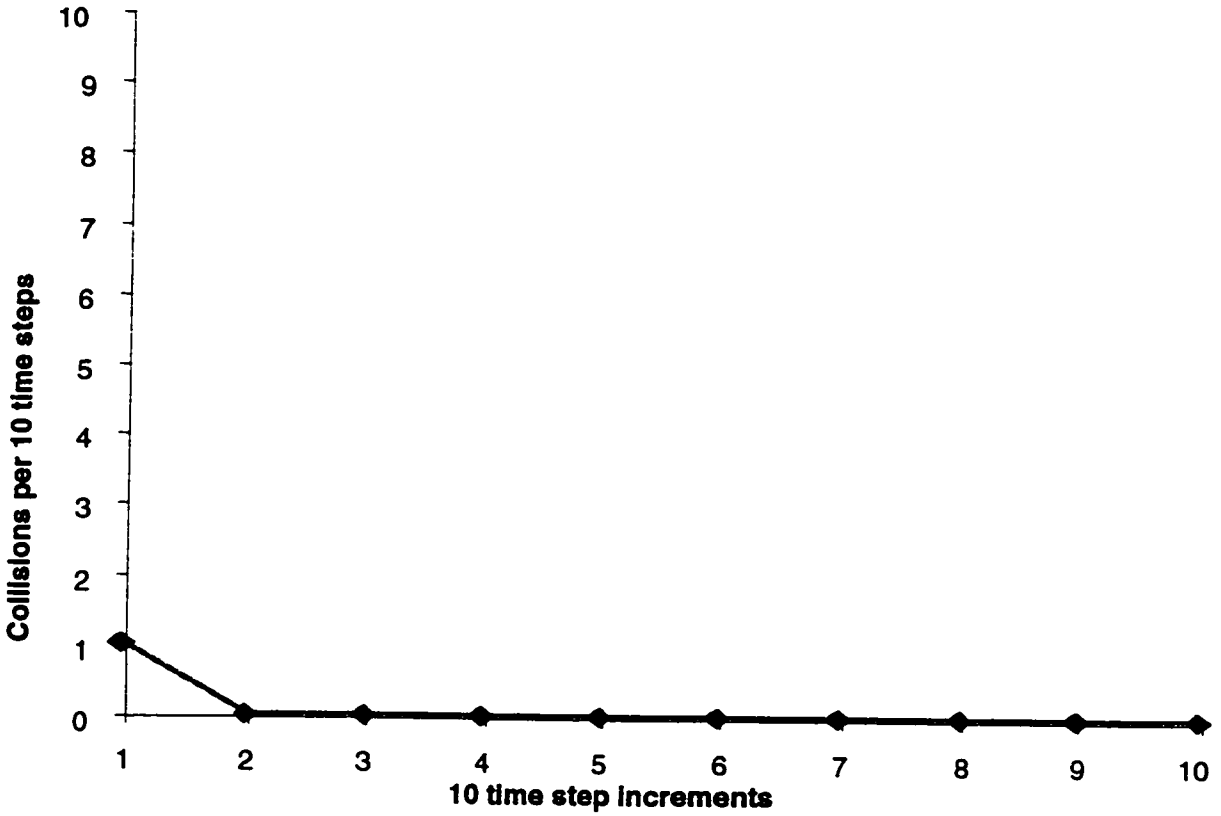


Figure 7-17 Performance of Perceptron DA after training.

7.8 Conclusion

In this chapter we have examined the RIDA architecture in the autonomy task framework developed in the previous chapter. We confirmed that the architecture supports all the tasks, requiring only relatively simple changes to the vehicles involved. From this result we can draw the conclusion that the RIDA architecture support first order autonomy when dealing with the mobility problem. This result is independent of the learning algorithm employed, as can be seen in the different capabilities of the rapid reinforcement network and the perceptron. The architecture can be scaled by training the DA which suits the task at hand. This might mean the training of a more capable DA by a lesser RI (as seen in tests one through three) or the training of a less capable DA by a very capable RI (as demonstrated in test four).

8. Relevance

Daddy gone?

Charlotte Ferworn

8.1 Introduction

In this chapter we focus on the five areas outlined in chapter one. We recap what has been discussed, reiterate how it has been demonstrated and discuss the significance of each item. Within this context, we go on to suggest areas where the RIDA architecture could be employed to enhance existing systems.

8.2 Revisiting the goals of this work

- We developed a framework for autonomous mobile agents based on the human concept of autonomy.

We argued in chapter five that common ground between psychology, philosophy and robotics could be found when discussing autonomy and that the terminology and concepts introduced in chapter two could be applied to robotics. The linkage was forged when biological systems were examined as machines, taking into account humanity's membership in this class. We demonstrated that with this linkage, a new way of examining autonomous vehicles and tasks is possible and necessary. Necessary because there is currently no common ground to examine "autonomous" systems in any kind of objective framework. Individual investigators end up comparing the merits of individual vehicles rather than the merits of the design methodology and principles the vehicle follows.

We have introduced such a framework to the literature based on notions of first order and second order autonomy. First order, or autonomy of action, has been displayed by various mobile robots including those employing reactive

mechanism as pioneered by [Braitenberg 84] and successfully implemented by [Brooks 86]. However, it is an unresolved problem to apply reactive techniques to achieve the reflective nature of second order autonomy. When a mechanism cannot learn it cannot miraculously improve to change its fundamental nature.

While second order autonomy has not been demonstrated by RIDA, this architecture among others [Nehmzow 94][Fagg 95], has the potential for reflection.

We went on to place the RIDA architecture within this framework. The placement of RIDA allowed us to suggest tests which either would confirm or deny RIDA's place in the framework. This, in turn, allowed us to test RIDA in meaningful ways with test situation which support the framework.

It must be emphasized that the literature does not use this methodology for examining vehicle architectures. In many cases authors rely on the capabilities of their vehicles to suggest tests which they can pass. We are suggesting that it is more appropriate to pick tests to confirm the placement of an over all architecture within our framework for autonomy.

- We demonstrated how the RIDA architecture supported the training of a learning system while at the same time allowing real time interaction with the environment while learning occurred.

Chapter six discussed the implementation of two learning systems selected from the literature. One very simple--the perceptron requiring supervised training, and another complex system implementing a reinforcement learning scheme. We demonstrated that both disparate learning algorithms could be incorporated into the RIDA model and could, in fact, coexist.

We demonstrated that the presence of a RI component not only protects the vehicle from the fall-out of erroneous decisions made by a high level controller--

the DA, but it also can contribute to improving the speed of learning in the DA. This became amply clear in the startling reduction in training time experienced by the rapid reinforcement DA when a RI was present.

It was also shown that the RI component plays a vital role as a safety system as the vehicle learns. The trials conducted in chapter 4 illustrate this as human operators were significantly aided in a difficult control task by employing an appropriate reliable RI.

It was also demonstrated that by employing both a RI to provide reinforcement to a DA a richer set of information is possible. While reinforcement signals can be provided by the RI it is also capable of providing a correct response when an error has been made by the DA. In this way both reinforcement and supervised learning were demonstrated.

- We showed that the architecture supports the graceful degradation of the vehicle's performance as control subsystems are removed or fail due to damage or through other unforeseen circumstances.

We began in chapter three with the understanding that a high level controller is likely to fail. In chapter five we presented alternative low level controllers for collisions avoidance and light seeking which could be, for the most part, depended upon to "rescue" the high level controller. We tied these notions together in chapter four when they were described as features of RIDA.

In addition, this work contributes to classic reinforcement learning by suggesting a workable method for extending the available time for learning through action of the RI. In this way, slower learning is not as much of a problem as in systems which employ reinforcement learning alone. Because the RI also provides an indication of what the correct action might be, it also has been shown to provide a richer reinforcement signal.

- We demonstrated how the architecture can be expanded allowing cascading RIs to coexist without substantially changing the existing modules.

In chapter six we demonstrated in autonomy test four, three levels of RIDA interaction when we employed a Perceptron DA backed by a Reinforcement network RI which itself was backed by a collision avoiding reactive system RI. In this way we demonstrated layers of controllers in a strict but effective hierarchy interacting to protect the vehicle from damage and support the DA.

- We developed a simple yet effective scheme for tying the RI and DA modules together

We presented the basic RIDA architecture in chapter three. We argued that the DA must be allowed to make decisions and the RIs must be allowed to intervene and correct those decision. This was confirmed in chapter six.

8.3 Continuing Work

Since the completion of this work two promising areas have been pursued. These will be introduced in the following sections.

8.3.1 N-CART, The Natural Selection Research Group and Autonomous Vehicles

Independent work has started to implement a full version of the RIDA architecture within the Natural Selection Research Group at the University of Guelph and at Ryerson Polytechnic University in the Network-Centric Applied Research Team (N-CART).

At Ryerson it is currently the focus of an undergraduate student research project. The initial tasks will be those described in chapter five. It is planned that the

vehicle will be able to successfully negotiate the classrooms, lounges and students on the computer science floor at Ryerson.

At Guelph the autonomy architecture is being used in a masters thesis project to simulate certain biological reflexes accompanied by learning. This work is nearing completion with a successful implementation of RIDA.

8.4 The Real Time Problem

The RIDA presents an architecture which supports the graceful degradation of a controller's performance. In addition, it has been shown that The RI portion of the architecture can be used to speed the learning of the DA portion by providing a richer signal to the learning algorithm. However, the selection of the DA is still problematic. While the Rapid Reinforcement network provided a workable solution to the problem of slow learning in a spatial task, it is obviously limited in its applicability to other, more complex tasks.

Work must continue in the area of rapid learning, as clearly the need for speed is applicable to all areas of robot control. There have been fruitful efforts demonstrated by [Dorigo 93] and others in applying genetic-based approaches, but the development of rapid learning techniques applicable to a wide range of control tasks is still an unresolved issue.

8.5 After word

We have introduced several unique theoretical contributions including a framework for autonomy and a workable architecture for the implementation of reliable learning first level autonomous agents called the RIDA architecture. We have demonstrated the feasibility of constructing agents employing this architecture and have argued the benefits of relying on a complete architecture

supporting graceful degradation, the ability to attach additional RI controllers when desirable and provide reliability.

Appendices

Appendix A: Sample Simulator Telemetry Information

Appendix B: The EMMA Experiments

Appendix C: The Rapid Reinforcement Network Architecture and Algorithm

Appendix D: Implementation Details

Appendix A: Sample Simulator Telemetry Information

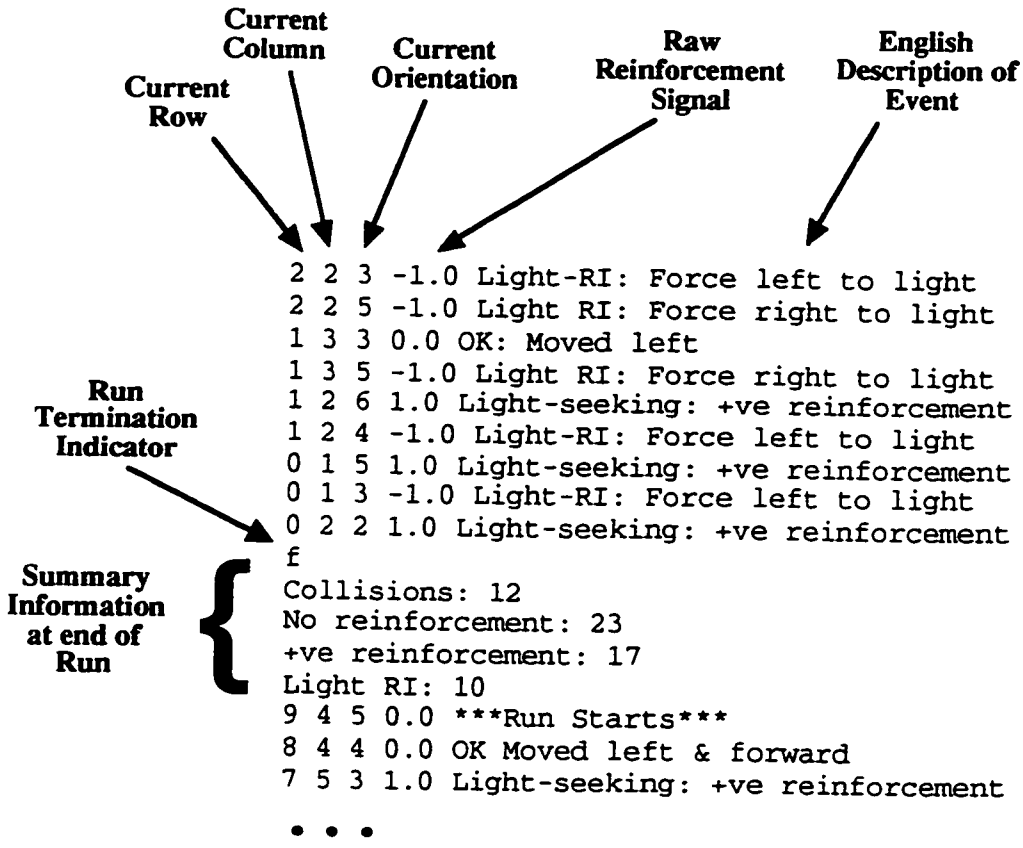


Figure A-1 Example telemetry information

Appendix B : The EMMA experiments

Introduction

These experiments extended our notion of continuous diverse mobility in an autonomous vehicle. By utilizing commonly available construction material and employing techniques which promote certain behaviors, a series of Electro-Mechanical Mobile Animats (EMMAs) were constructed.

The goal of this work was to push the realm of what is possible using only simple techniques. Perhaps surprisingly, these devices share many characteristics with more elaborate mobile robots. Even more importantly, these EMMAs not only function in their target environment but are equipped to survive in them and in some cases can adapt to changing environments.

The Environment

A standard "Arena" was selected as the target environment for each EMMA. The arena is essentially a bathroom, but for the vehicles it is a flat surface surrounded by walls with a number of discrete obstacles which is in a constant state of change as people and cats enter and exit the room, obstacles are placed in the room and then removed again. The arena was modified by adding a high intensity light source attached to one of the walls. This light source acted as a power source for various EMMAs.

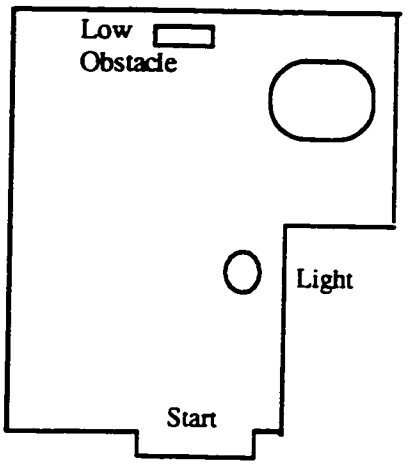


Figure B-1 The Arena

EMMA I: Foraging

When we think, work or play we are constantly using energy and must eventually refuel. If we fail to refuel in a timely manner we stop functioning. This is also true of any artificial autonomous vehicle. Various Animats have been constructed which are designed to seek a "docking station" when their source of energy is spent. This solution was considered in the design of EMMA I but was quickly rejected. For a docking station to be viable, any vehicle relying on it must be equipped with a means of finding the station when required and successfully orienting itself for refueling.

To avoid these problems a relatively inefficient but highly effective means was selected for refueling. The power source for EMMA I was a battery of 4 1.2 volt Nickel-Cadmium cells connected in series. These cells can be recharged by a regular battery charger. Instead, a solar panel was employed to produce a recharging current of 7.2 volts. This simple circuit is illustrated below.

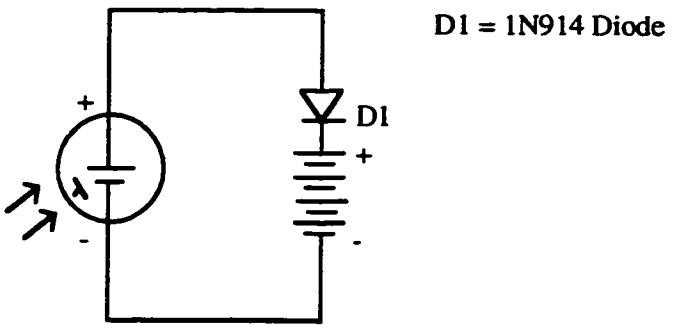


Figure B-2 Solar recharging circuit

Making it possible to recharge the cells did not mitigate the necessity of finding a location which could produce enough light energy to recharge them. Ideally EMMA I would be able to find the light source in the arena to recharge. This implied the ability to search --or forage for energy-- would be the prime behavior of EMMA I.

A simple crossed connection mechanism for moving towards a light source was suggested by [Braitenberg 86]. This design is illustrated below.

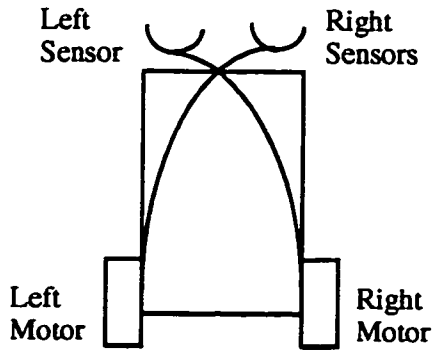
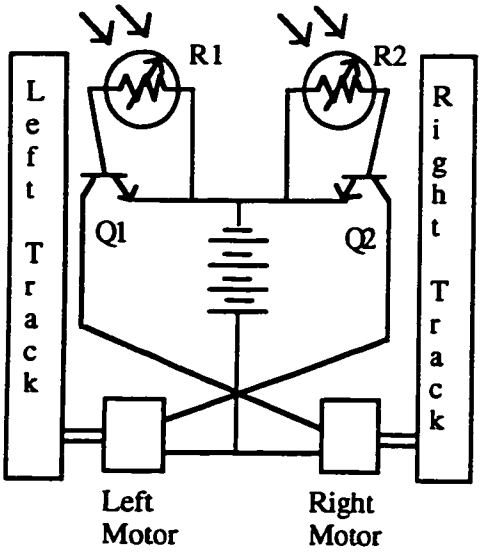


Figure B-3 Braitenberg's crossed connections

As the a sensor comes in contact with whatever it is sensing, it allows the opposite motor to turn more quickly thus turning the whole vehicle towards the source.

Photo resistors were selected as the left and right sensors. A modified Braitenberg circuit is illustrated below.



R1, R2 = 50Ω 1/4 watt resistors in bright light and MegaΩ resistors in darkness.

Q1,Q2 = 2N2222A transistors.

Figure B-4 EMMA I search circuit

When a photoresistor is in contact with a bright light source, its resistance is reduced which allows more current to reach the transistor's base which, in turn, causes the motor on the opposite side of the photoresistor to turn faster. In this way EMMA I has a tendency to turn toward the light.

If this circuit were left as the only mechanism, it would fail to bring EMMA to an appropriate recharging point as the tracks would move EMMA past the maximum light source. This occurs because as long as R1 and R2 in the diagram receive light from the source their resistance will be relatively low and the vehicle will not stop optimally.

To avoid this situation a simple solution was adopted. The solar panel was placed to partially obscure the photoresistors. This has the benefit of allowing the sensors to pick up a bright light source and move towards it when the source is in the distance. Once the tracks have moved EMMA I under the light source a shadow is

cast on the sensors which raises their resistance and stops the vehicle directly under the light source. This is illustrated below.

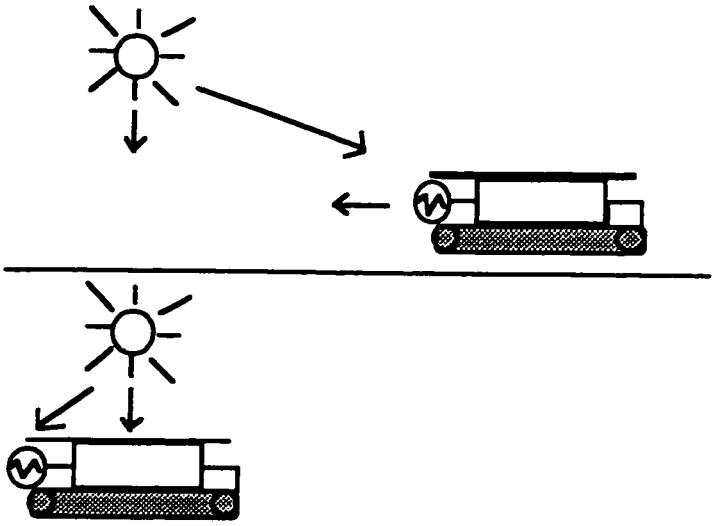


Figure B-5 EMMA I moving, stopping and recharging

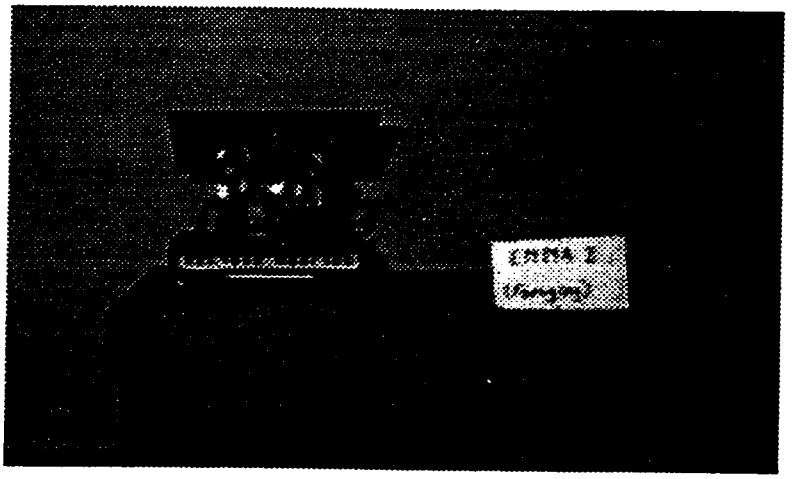


Figure B-6 EMMA I

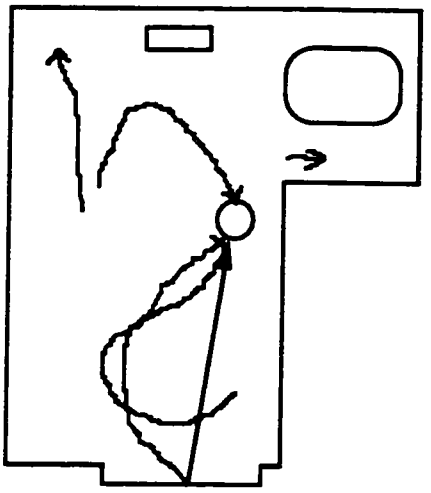


Figure B-7 EMMA I typical trajectory in the arena

Lessons Learned from EMMA I

Several important lessons were learned,

1) The solar panel is capable of recharging EMMA's power pack in about 12 hours under a 100 watt light source.
2) Differential steering using tracks allowed EMMA to negotiate small obstacles and maneuver very quickly.
3) There is no need for a special docking station with its associated docking problems using this technique. As long as the vehicle finds the source it will stop in the vicinity of it.

EMMA II: Following and Wandering

While the ability to refuel is essential, it leads to quite boring behavior. EMMA I would find a light source sufficiently strong to attract it, and stop directly under it. EMMA I would only move if the light source moved or another light source attracted its sensors.

EMMA II was designed to move around its environment. It is interesting to note how successful an insect like a cockroach is at moving around its environment [Dalcomyn 93]. One of the goals of EMMA II was to develop movement which is similar to that of a cockroach [Pearson 76] of course EMMA's movement would be on tracks and not legs.

Of particular interest was a cockroach's ability to follow the edge of a wall. This behavior may not be intelligent but it does tend to keep the insect out of harm's way. When not following a wall, a cockroach will wander until it finds one. This behavior can sometimes be observed when the lights are turned on in a room full of roaches. It also seemed like a reasonable goal for EMMA.

Several solutions to wall following and wandering have been proposed. [Beer and Chiel 90] simulated this behavior using neural networks consisting of dozens of model neurons heavily connected. [Jones and Flynn 93] suggest an analog Resistor-Capacitor circuit designed to back the vehicle out of trouble when necessary. Both solutions were considered to be a bit complex for the nature of the problem. Instead, the circuit illustrated below was employed in EMMA II.

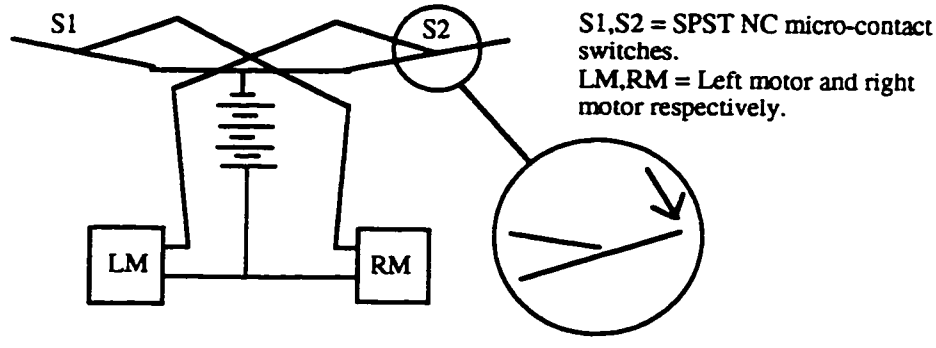


Figure B-8 Contact switches used to promote wall following in EMMA II

In order for EMMA II to actually follow a wall it must find one. Since there is no world model within EMMA, a bit of entropy was used. The differential steering mechanism used in all the EMMAs employ two motors. Although the motors are

very similar, one has a tendency to spin faster than the other. This, in conjunction with differing component friction, causes EMMA to tend towards one side. In our case EMMA II liked to move slightly left. This left movement causes EMMA II to seem to prefer walls on the left, this turned out to be rather important.

When one of EMMA's sensors comes in contact with a wall surface, the power is cut to the motor on the opposite side. This power loss allows the other motor to move EMMA away from the wall until the sensor loses contact with the wall and the stalled motor begins to spin again. Because the vehicle has a tendency to move left, EMMA is soon in contact with the wall once more and the process is repeated. In this simple way wall following is achieved. With this arrangement, wandering behavior is a simple matter. When there is no wall present, EMMA II will wander!

This leaves the problem of tight corners. The previous set of sensor "whiskers" are inadequate to prevent getting stuck in a corner. As the vehicle follows a wall and strikes an orthogonal wall it will surely become stuck. The solution to this is quite simple. A second set of whiskers were added to the first and directed forward. Additionally, the right-hand switch was replaced with one which required more force to activate than the left-hand switch. This arrangement is shown below.

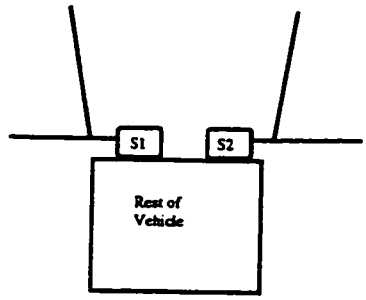


Figure B-9 Modified Whiskers on EMMA II

The forward whiskers come in contact with the intersecting wall. As the left whisker touches the wall it disconnects the right motor and EMMA swings around

the corner without getting stuck before the right switch is depressed. This situation is illustrated below.

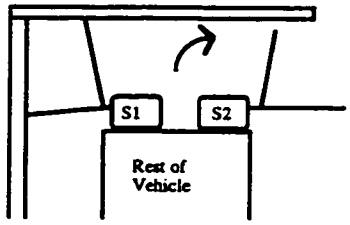


Figure B-10 EMMA II negotiating a corner

There is another situation which could conceivably trap EMMA. This situation can be thought of as a narrowing blind ally. Initially the entrance is wide enough to allow the whisker sensors to pass without activation but then becomes so narrow that both whiskers are activated at once and the vehicle halts. For this situation the SPST momentary contact switches were replaced with DPDT switches. When both DPDT switches are activated a separate RC circuit, similar to the roach circuit of GARBOT III, is used to reverse the right-hand track for a specified period, effectively turning EMMA around to a clear path.

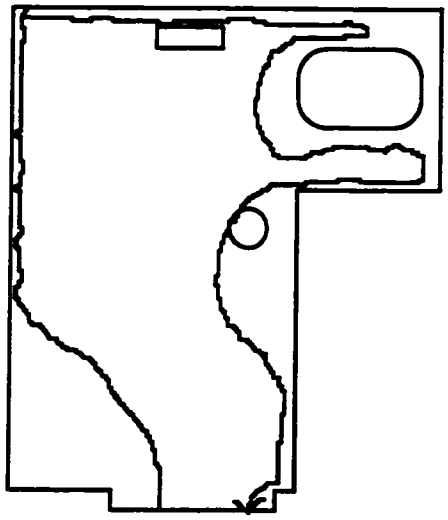


Figure B-11 Typical EMMA II trajectory in arena

Figure B-12 EMMA II

Lessons Learned from EMMA II

1) Simple switches are, for the most part, capable of performing fairly sophisticated control tasks.
2) By modifying the whisker extensions on the contact switches, the repertoire of behaviors can be extended without adding additional circuitry.
3) Many approaches used in the past to solve this simple control problem have been overly complex and costly.
4) This solution to the control problem has the benefit of simplicity and robustness. EMMA II eventually achieved a peak performance of about 40 minutes of continuous movement before the power cells ran low.

EMMA II.5: Switching Behaviour

EMMA I was capable of feeding, EMMA II was capable of exploring. It was now necessary to fuse the behavior of both these vehicles into one. The required behavior is one of general exploration until fuel is low (hunger) then look for a way of recharging (eating) and then continue wandering. Clearly, one did not want the creature to constantly feed or have the creature completely drain its cells before

refueling. In fact, there is some evidence that this type of behavior might be governed by simple hysteresis loops governing the desire to feed or drink in some animals [Booth 78][Toates and Oatley 70].

An analog circuit was devised which could accomplish this. The creature would function mostly as EMMA II would--searching the arena. While searching, all current to the drive motors would be coming from the primary cells. The primary cells would also be holding closed 3 low-power DPDT relays 2 of these relays would serve as shunts passing electricity to the drive motors. The other relay would connect the EMMA I solar cells to the secondary power cells for recharging from ambient light.

When the primary cells are almost exhausted, the relays will flip open. This will direct power from the secondary cells to the motors and cause sensing to come from the photocells. In turn the third relay will connect the solar cells to the primary power cells. When the vehicle arrives at a good feeding area, it will stop until the primary cells are recharged. When this occurs the relays again flip closed and EMMA II.5 can go on its way. In this way the behavior is modified to suit the circumstances. A simplified circuit is shown below.

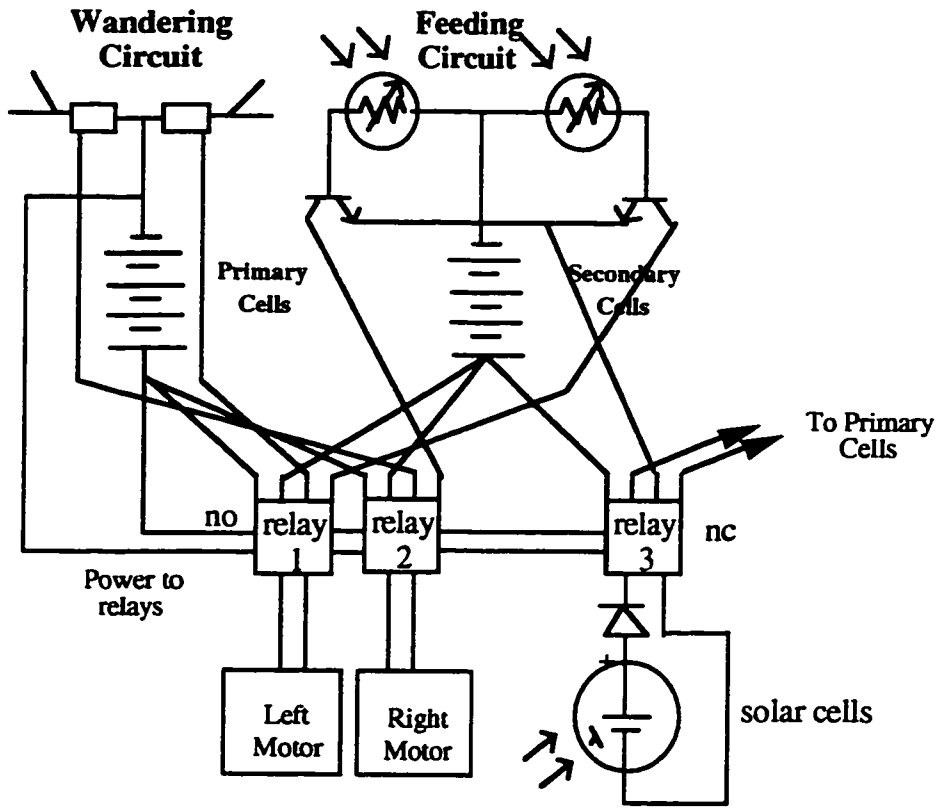


Figure B-13 EMMA II.5 behaviour modification circuit

When EMMA II.5 was constructed it moved basically as indicated. Two successful runs were made starting with fully charged primary and secondary power cells. The vehicle wandered around the arena for about 35 minutes to 40 minutes until the relays opened.

In the first run EMMA successfully found the feeding station and fed for about four hours and then began to wander again. This EMMA wandered for about 10 minutes until its left- hand contact whisker became jammed in a crack and damaged the contact switch at which point the trial was stopped. As previously mentioned, the cycle time for recharging from solar cells is about 12 hours. It was surmised that the primary cells were not being fully charged before the relays fired. To improve charging a Zener diode was placed in series with the primary cells. Since the diode had a 6 volt conduction voltage, the cells charged for a longer period of time before the relays fired.

In the second run EMMA found a light source it liked better in a reflection coming from a wall outside the arena. Manual intervention (turning it around) was needed to get EMMA back to the correct feeding station. The trial was stopped soon after this.

EMMA II.5 is capable of extremely complex behavior yet the creature is limited by its inability to adapt to a changing environment. This is generally true of all subsumption systems. The behavior is "hard coded" into the AFSMs which makes them quite inflexible. While it is possible to add new AFSMs or change their connections, this must be done through manual intervention which may not be feasible in a changing environment.

A changing environment is not that hard to envision. In the case of the arena suppose the bathtub overflowed to partially flood the arena--how would the tracks work? A predator might be introduced or considerably more obstacles added. These situations must be anticipated by the system designer. It would be useful if the Animat could adapt to these new situations without intervention.

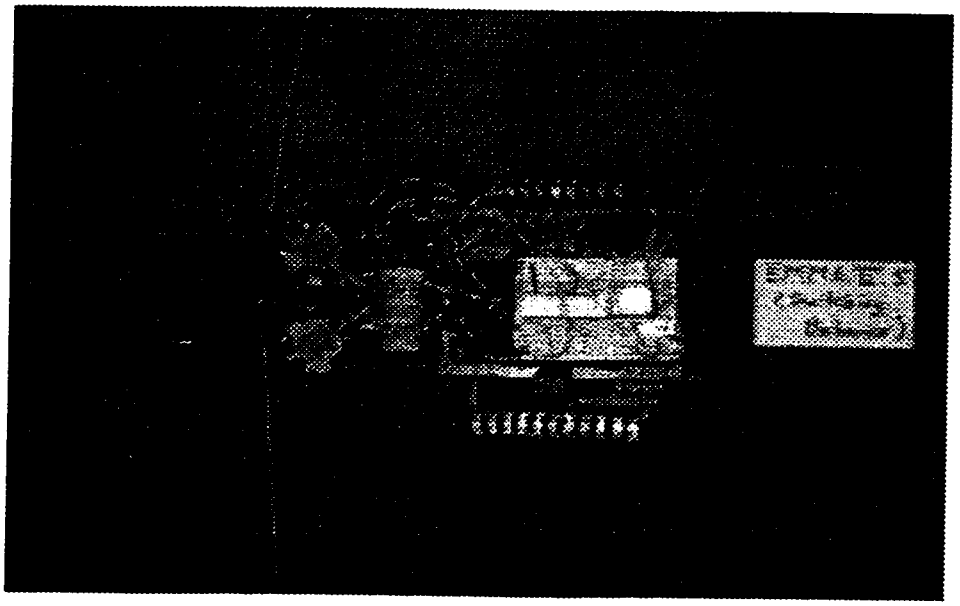


Figure B-14 EMMA II.5

Lessons Learned

1) Subsumption can lead to complex behavior which is suitable for a relatively stable environment.
2) Subsumption architecture creatures are incapable of learning from their environment or adapting to a new one.
3) There are situations in which the ability to adapt and to learn from the environment are essential.

EMMA III: Defence

One of the problems experienced by EMMA I and II was not unique to these vehicles. As part of the arena environment objects can come and go--including Dave the cat. The EMMAs are generally good at avoiding obstacles but they are incapable of deterring a potential predator from attacking them. The goal of EMMA III was to create a creature capable of defending itself from external attack (in this case a tabby pulling at EMMA's wires). A number of different non-lethal options were available.

Initially a pump was connected to a series of contact switches. The intention of this arrangement was to shoot a stream of water straight up in hopes of hitting the predator. This behavior is similar to that of a skunk except the skunk is capable of more accuracy. The technique was abandoned when it became apparent that the stream of water simply annoyed the cat and made him even more determined to attack.

Another approach was successfully implemented using a series of camera photo-flashes. The capacitors of the flash circuits were charged by two of the four nickel-cadmium cells. The flashes had the characteristics of a fast recharge cycle, brilliant flash and minimal current requirements. 5 flashes were distributed around the body of a test bed vehicle, made from a stripped down EMMA II.

Each flash had an SPST momentary contact switch protruding from the body of the vehicle. All flashes were capable of operating and recharging independently. This defensive technique is very similar to the way a porcupine defends itself using its quills. The vehicle could move around the arena and not activate any of the flashes but if one of the contacts were touched a temporarily blinding flash would result. This technique proved very effective.

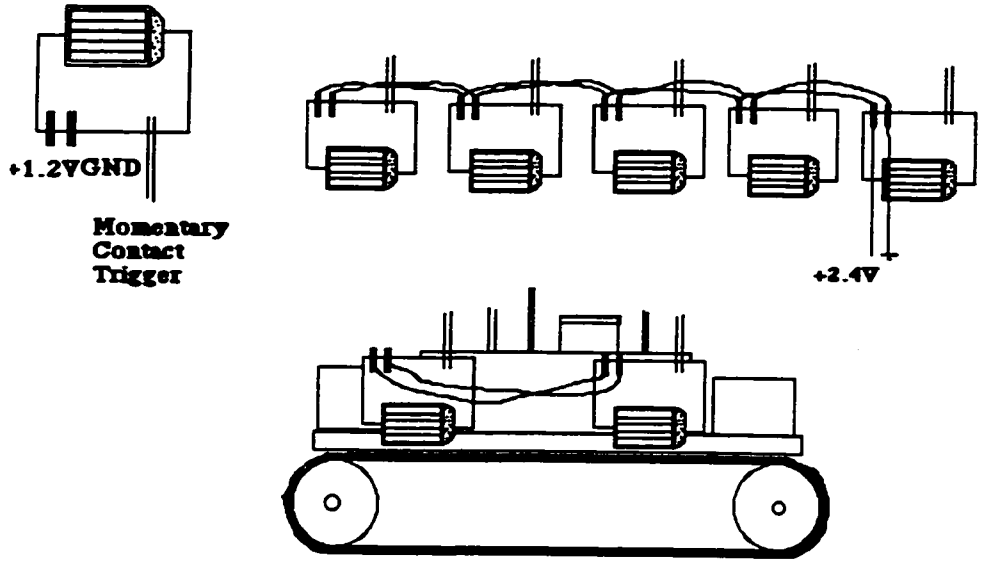


Figure B-15 EMMA III (the photo-pine)

The trial was conducted in the dark except for the shielded arena light source illuminating a small section of floor. EMMA III wandered for about 5 minutes and was then assaulted by the cat. Two flashes were activated. Dave lay down and closed his eyes. The vehicle wandered for about 10 more minutes before the trial was stopped. Dave the cat had lost all interest in trying his luck and never came near the arena again that day.

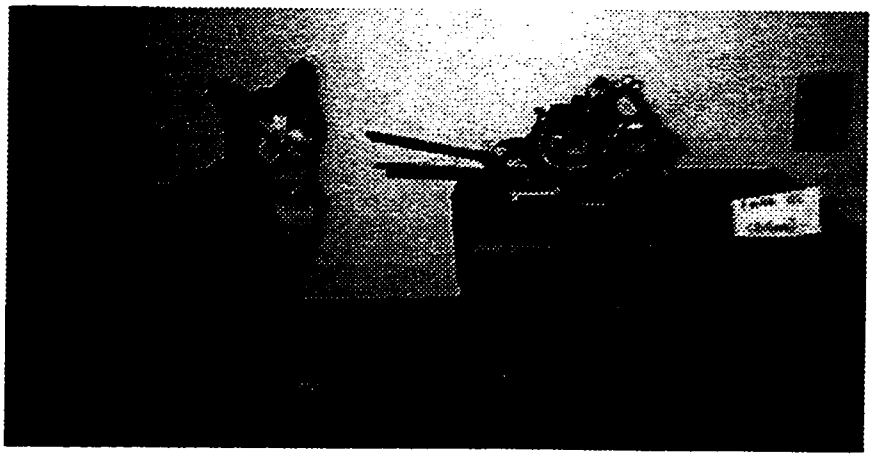


Figure B-16 EMMA III

Appendix C :The Rapid Reinforcement Network Architecture and Algorithm

The following is a brief description of the algorithm derived from the work of [Fagg et al 94], [Barto et al 83], [Widrow and Hoff 60], [Klopf 82], and [Sutton, 88]. The operation of each part refers to the diagram above it.

Reinforcement Network

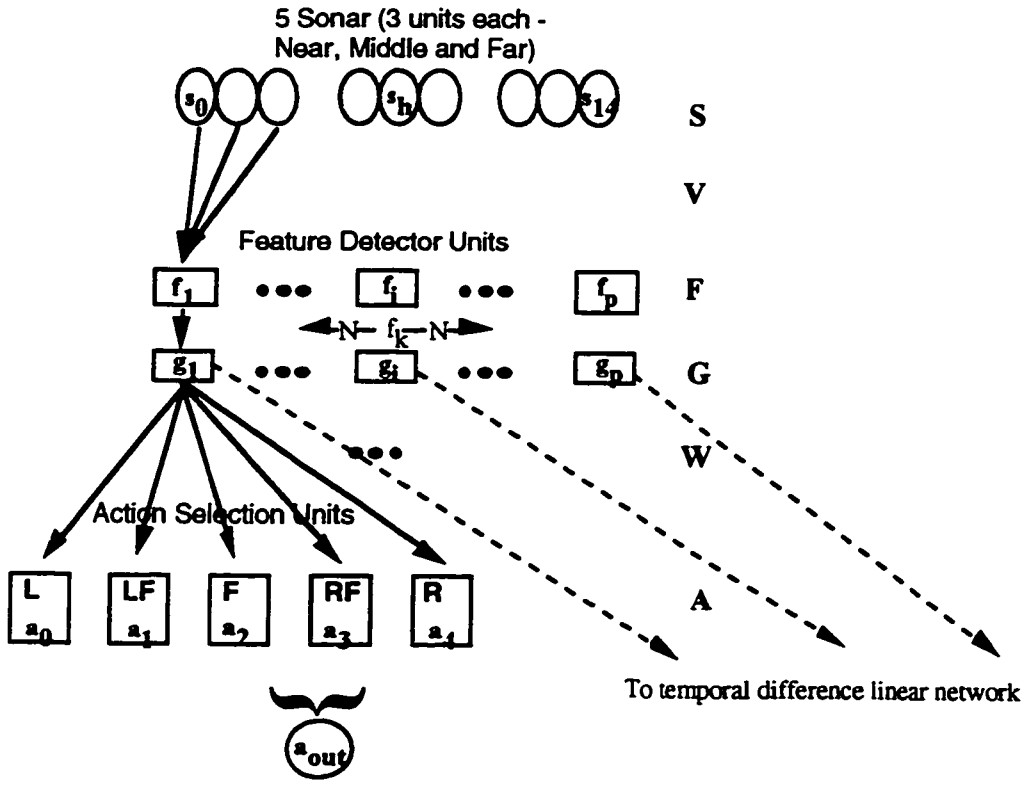


Figure C-1 Rapid Reinforcement Network Architecture with Input and Output shown

Initialization

Assign random values between -1 and 1 to all v_{hi} and w_{ij}.

Feed Forward Operation

For all i=1..p $fi = \sum_{h=1}^{n}(s_h v_{hi}) + \xi$ where ξ represents random noise.

For all i=1..p

$$\text{Winner}_i = \begin{cases} 1 \text{ if } f_i = \text{MAX}_{i-N \le k \le i+N}\{f_k\} \\ 0 \text{ otherwise} \end{cases}$$

Where N defines a local neighborhood of feature detectors,

$g_i = Winner_i * f_i$

For all j=1..q

$a_j = \sum_{i=1}^{p}(g_i w_{ij}) + \zeta$

$a_{out} = MAX\{A\}$

Feed Backward Operation

$\Delta v_{hi} = \alpha R' e_{hi}$

where $e_{hi} = \delta e_{hi} + (1-\delta)(s_h g_i v_{hi})$

for all h = 1..n, for all i = 1..p, δ determines the trace decay rate where $0 <= \delta < 1$. α is the learning rate where $0 <= \alpha < 1$.

$\Delta w_{ij} = \alpha' R' e'_{ij}$

where $e_{ij}' = \delta' e'_{ij} + (1-\delta')(g_i \hat{A}_j w_{ij})$

for all j = 1..q, for all i = 1..p, $\hat{A}$ is a vector where all elements are zero except for the jth where j is the winning action, δ' determines the trace decay rate where $0 <= \delta' < 1$. α' is the learning rate where $0 <= \alpha' < 1$.

Temporal Difference Linear Network

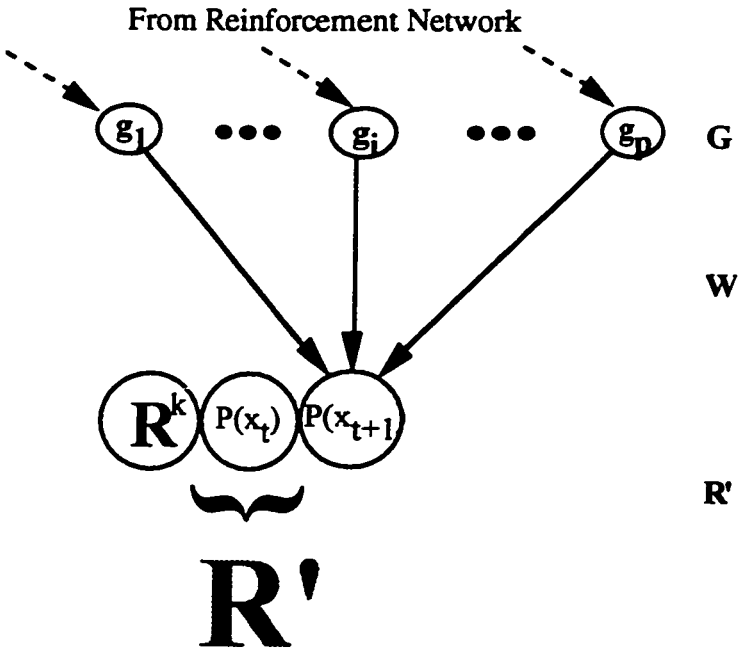


Figure C-2 Temporal Difference Network Architecture

Initialization

Assign random values between -1 and 1 to wi for i = 1..p and Q. l is the discount factor for future reinforcement, $0 < l < 1$.

Feed Forward Operation

$$P(x_{t+1}) = f(\sum_{i=1}^{p}(w_i g_i) + \Theta)$$

$$f(x) = \begin{cases} 1 \text{ if } x>0 \\ -1 \text{ otherwise} \end{cases}$$

$$R' = R^k + \lambda P(x_{t+1}) - P(x_t)$$

$$d = R^k - R'$$

Feed Backward Operation

$$\Delta\Theta = \alpha d$$

$$\Delta w_i = \alpha g_i d$$

Appendix D : Implementation Details

The following describes the implementation/simulation details of the vehicles employed in autonomy tests 1 through 4 of this work.

Vehicle 1

Recall that the vehicle is equipped with a sonar semi-ring of five sensors attached to the DA component and a left and right set of contact sensors attached to the RI component. Figure D-1 shows the interconnections between components of the simulation of test 1.

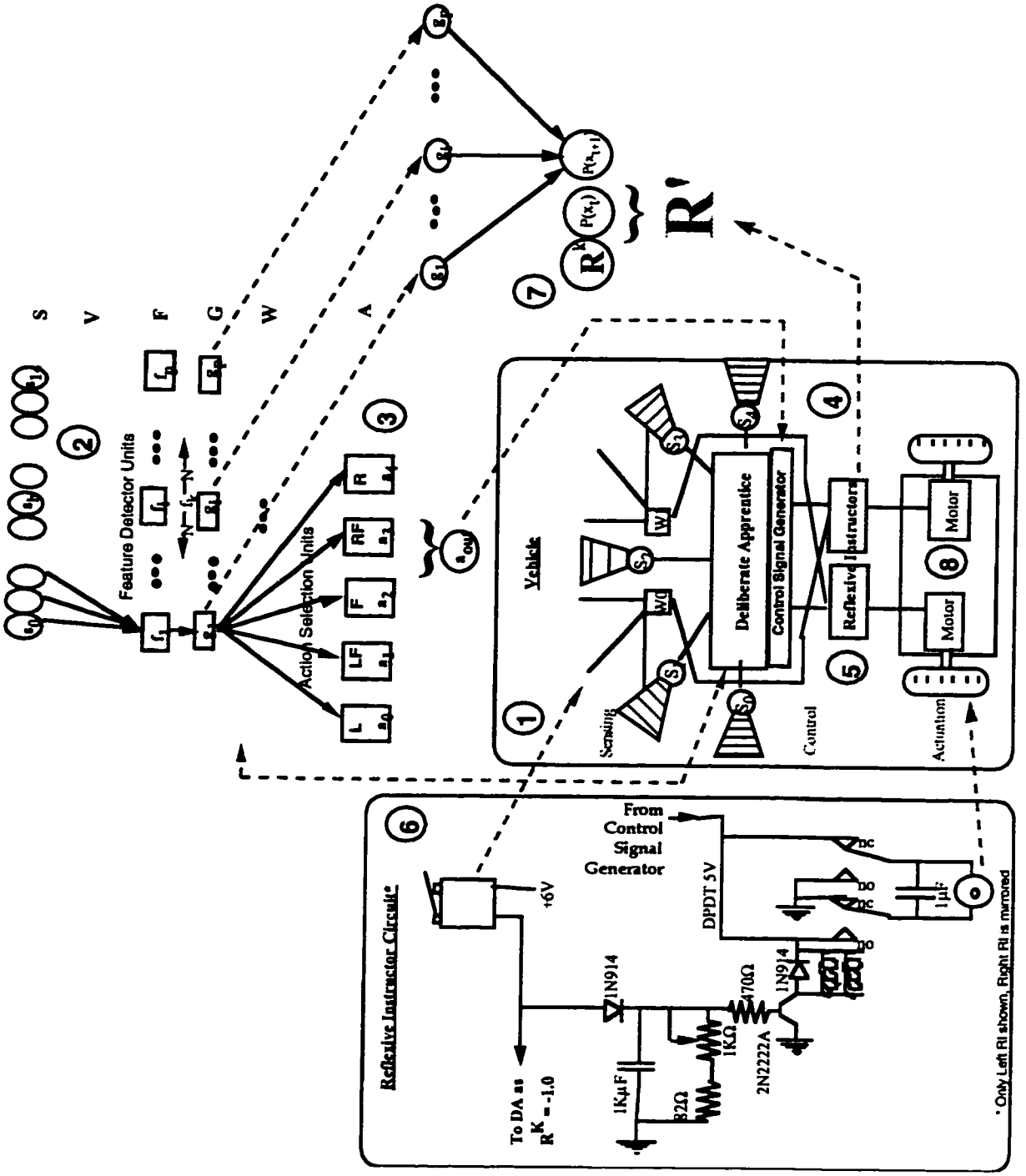


Figure D-1 RIDA Vehicle 1 Implementation Detail.

The portion of the diagram labeled with a circled 1 is the sonar semi-ring consisting of 5 sonar sensors which are described in section 7.5.3. Each is able to send an encoded signal back to the DA input units (2 in the diagram). The signals

are either "near", "middle" or "far" and activate the appropriate input unit in the rapid reinforcement neural network (RRNN) input layer. In each case, if that particular input unit is on it receives a value of 1.0 with the other receiving 0.0 inputs. The RRNN filters the signal through its feature detector units and eventually selects an action to perform in its action selection units (3 in the diagram).

The selected action is encoded (4 in diagram) and the appropriate control signals sent to the vehicles differential drive motors (8 in diagram). If one of the vehicles whisker sensors detects an object the RI component (labeled 5) is activated and it activates the reflex circuit as indicated in 6. In simulation this amounts to a quarter turn in the direction away from the obstacle. Head on contacts are treated stochastically with an even probability of moving left or right. The RI component sends a reinforcement signal of -1.0 to the RRNN linear network (labeled 7) which allows the RRNN to update its weights with this information.

The RIDA architectural diagram is shown in figure D-2.

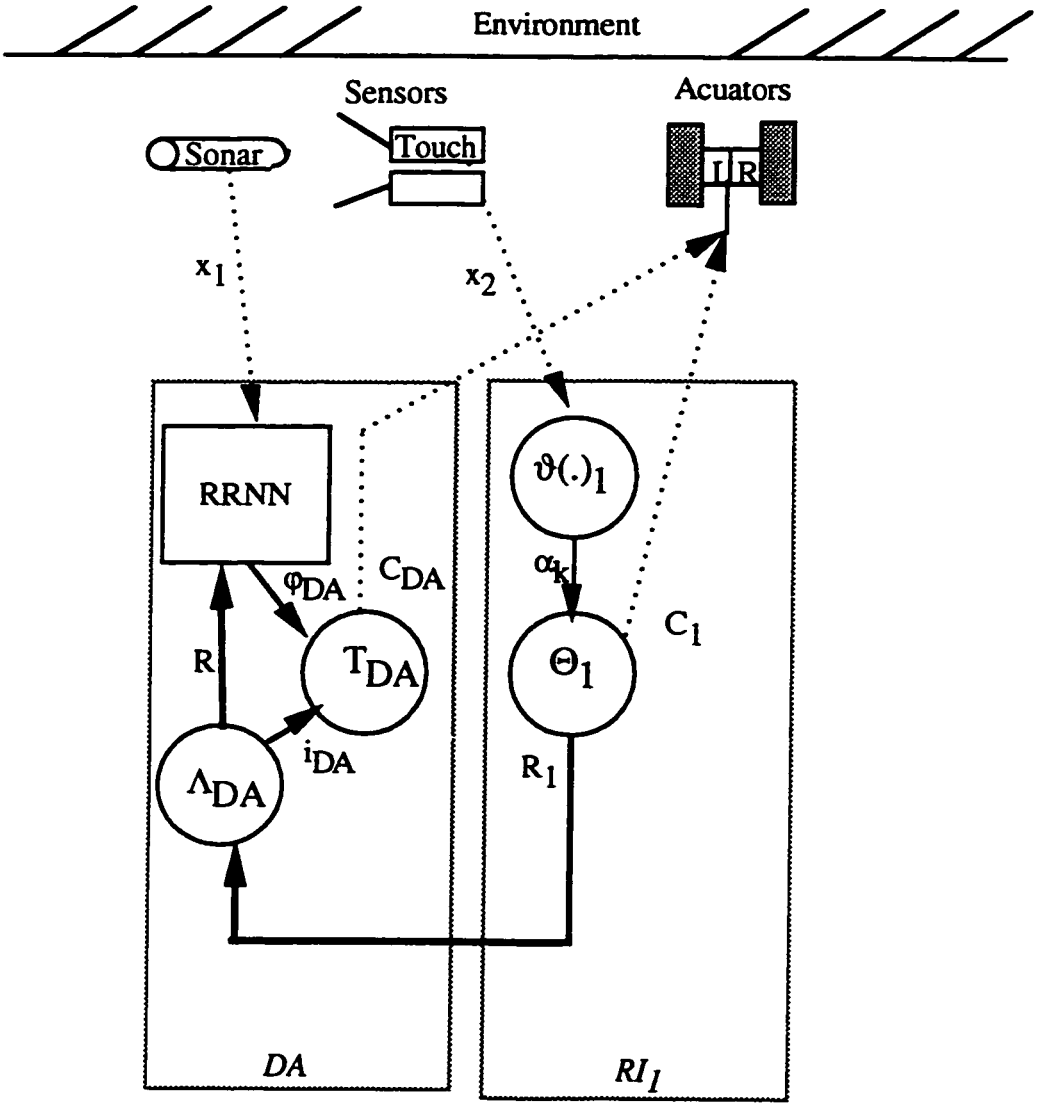


Figure D-2 Vehicle 1 RIDA Architecture

The sonar provides input to the DA while the touch sensors provide input the RI. Both components attempt to gain control of the actuators.

Vehicle 2

Recall that vehicle 2, like vehicle 1, is equipped with a sonar semi-ring of five sensors attached to the DA component and a left and right set of contact sensors attached to the RI component. In addition vehicle 2 is equipped with a semi-ring of 5 light sensors which are described in section 7.2.3. Figure D-2 shows the

interconnections between components of the simulation of test 2. For clarity, the Contact RI is not shown but is identical in location and function as in vehicle 1.

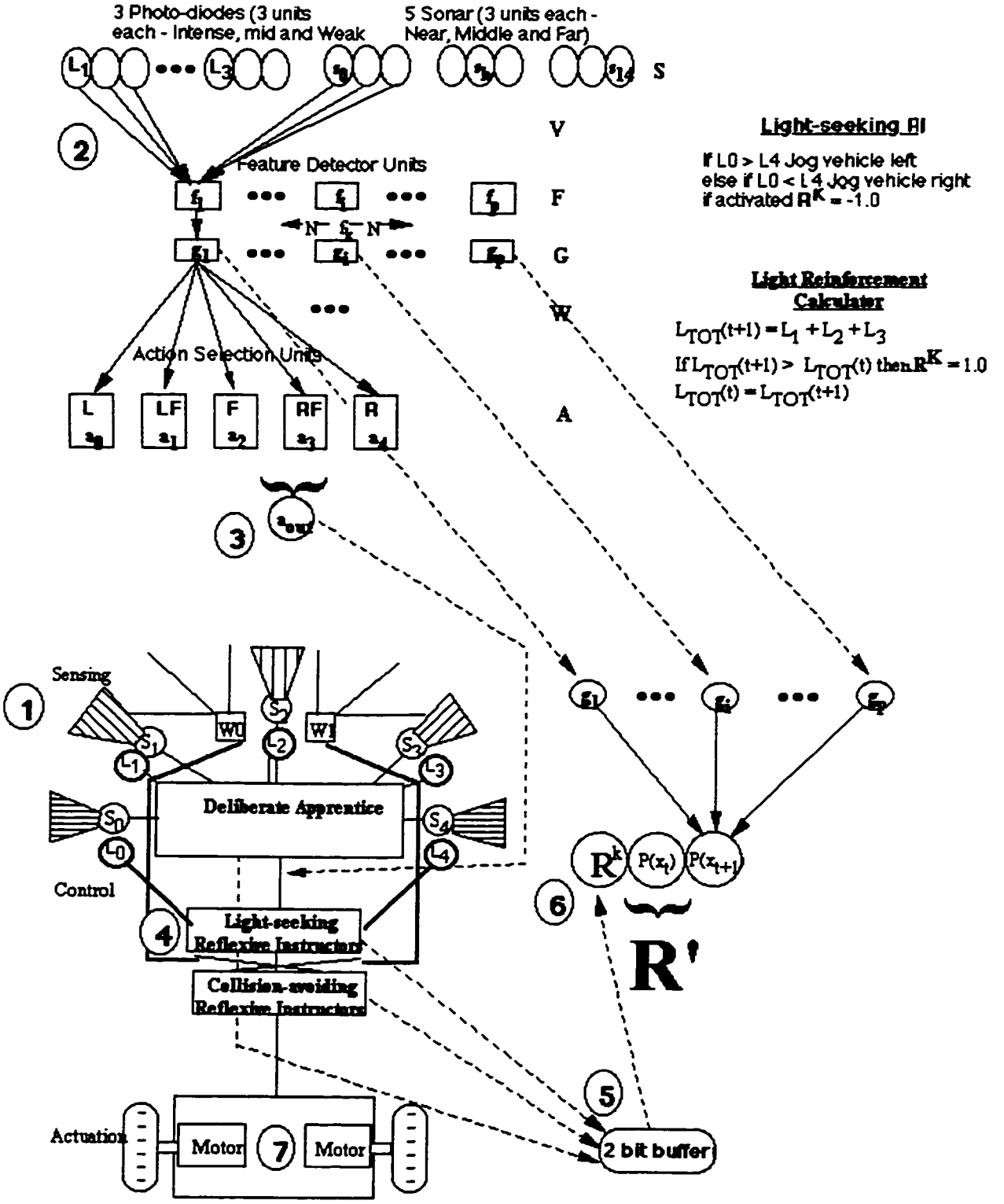


Figure D-3 RIDA Vehicle 2 Implementation Details.

The portion of the diagram labeled 1 are the sonar semi-rings consisting of 5 sonar and 5 light detecting sensors assumed to have the characteristic response of photo-diodes (linear over short distances). The sonar is identical to vehicle 1. Each light sensor is able to send an encoded signal back to the DA input units (2 in the diagram). The signals are either "near", "middle" or "far" and activate the appropriate input unit in the RRNN input layer. The RRNN filters the signal through its feature detector units and eventually selects an action to perform in its action selection units (3 in the diagram). The selected action is encoded and the appropriate control signals sent to the vehicles differential drive motors (7 in diagram). The collision detection RI functions in the same manner as in vehicle 1 and is the lowest level RI component in the hierarchy.

If one of the vehicles peripheral light sensors (L0 and L4) detects light, the vehicle is jogged toward the light by the light-seeking RI (shown at 4) unless there was a collision (see figure 0-4) in which case the collision RI inhibits this response. In addition the light-seeking RI generates a reinforcement signal according to the reinforcement policy shown at the bottom of the diagram. The signal is 1.0 if the vehicle had already moved in that direction (meaning the DA had initiated the action) or -1.0 if the DA had failed to move to the light and the RI moved for it. The 2 bit buffer (shown at 5) stores this signal until the linear network (6) in the RRNN can make use of it.

The RIDA architectural diagram is shown in figure D-4.

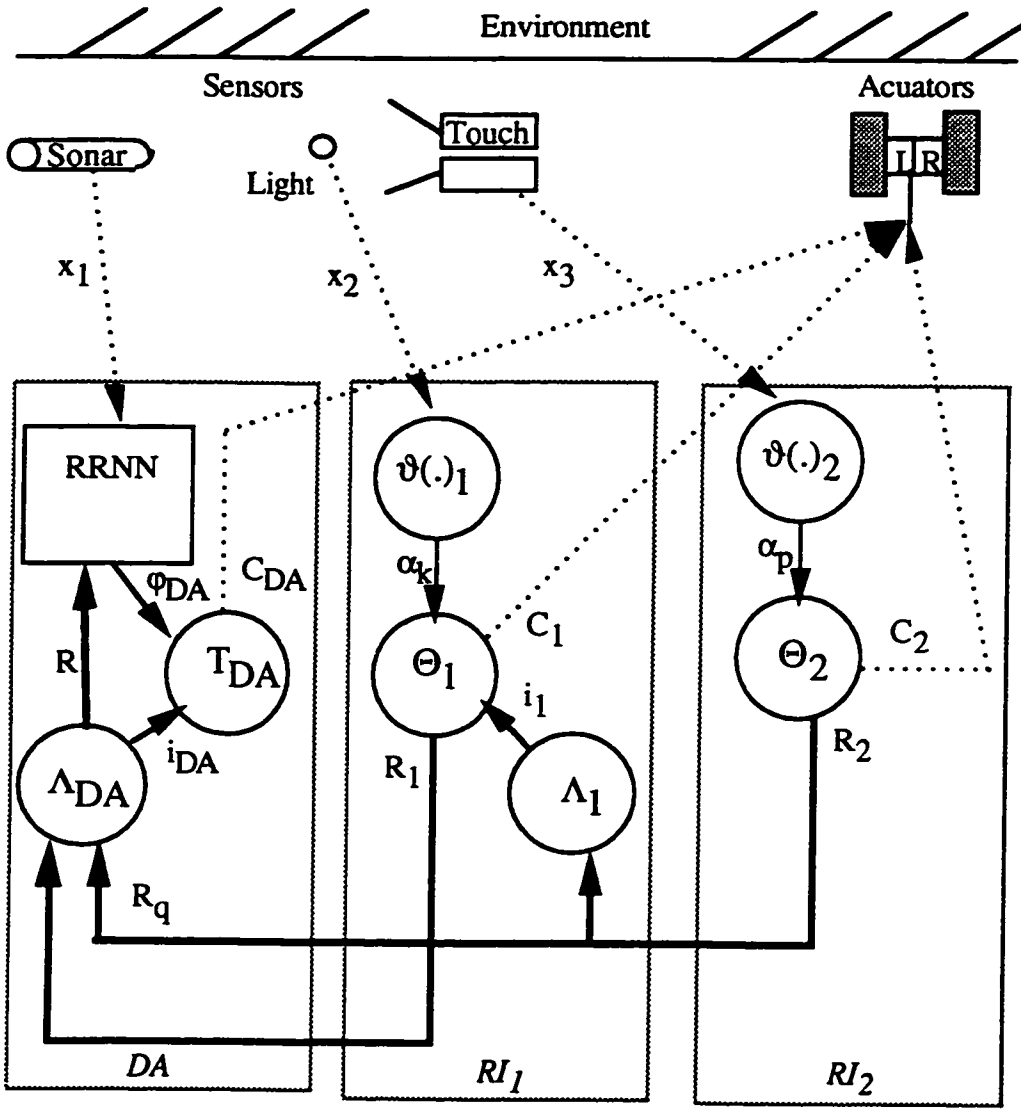


Figure D-4 Vehicle 2 RIDA Architecture

In the case of vehicle 2, the collision avoidance RI is the lowest level RI in the hierarchy. It is capable of inhibiting both the DA and the light-seeking RI. Note that in this case a richer reinforcement signal can be sent allowing both positive and negative reinforcement.

Vehicle 3

Essentially vehicle 3 is implemented in the same manner as vehicle 2 with the following exceptions;

1. The reinforcement policy is reversed for the light-avoiding RI. This means that if the DA moved towards a light source it received a -1.0 reinforcement with a 1.0 reinforcement if it moved away as measured over 2 time steps. No reinforcement was provided if no change in light was detected.
2. The control policy for the light-avoiding RI was reversed in that the vehicle was jogged away from detected light based on the peripheral light sensor readings.

Vehicle 4

Vehicle 4 was intended to demonstrate that a trained DA can take the place of an RI. In this case one of the RIs became a RRNN trained in one of the previous tests. The RRNN was replaced as DA by a simple linear perceptron. Figure D-5 shows the implementation details of vehicle 4.

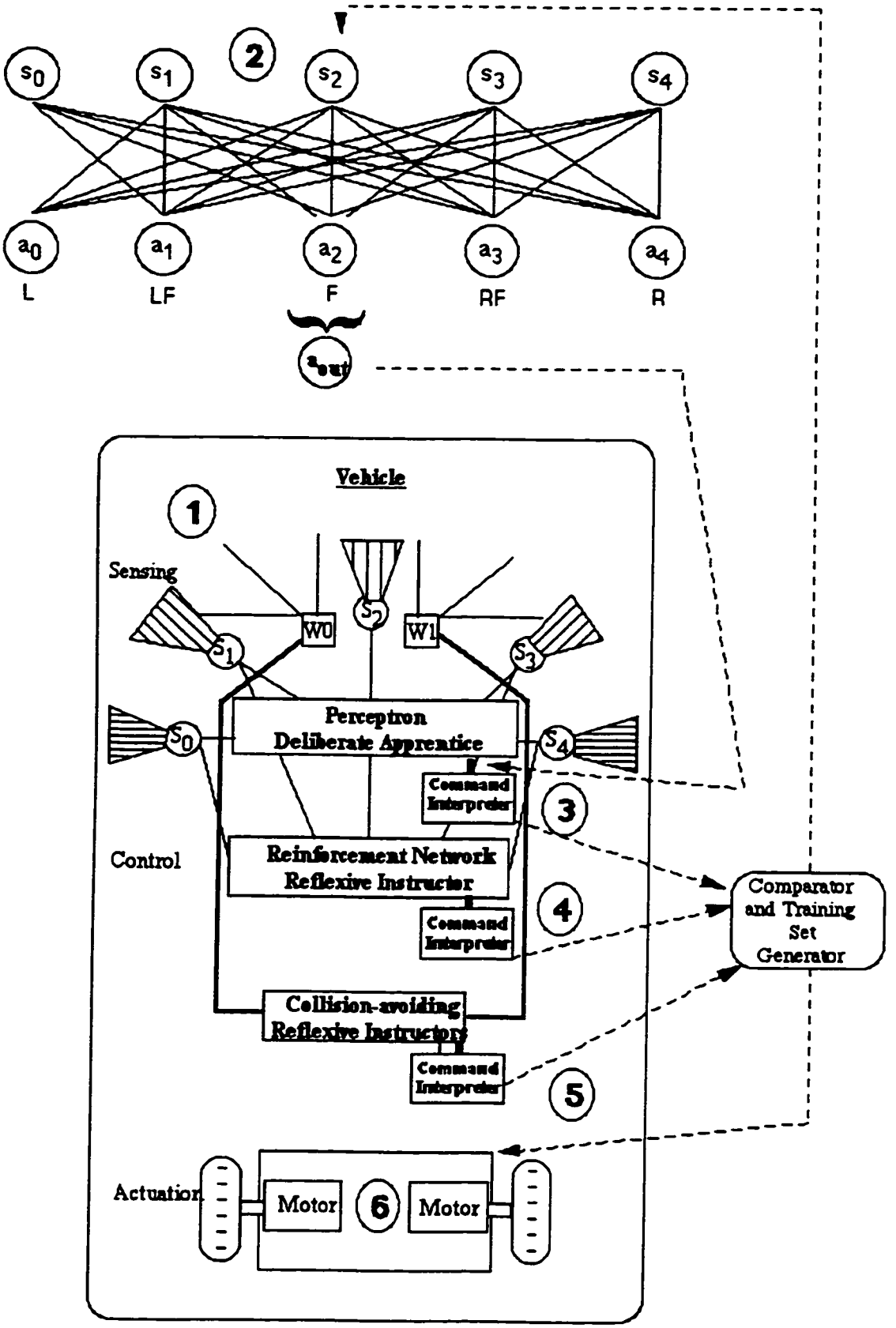


Figure D-5 RIDA Vehicle 4 Implementation Details

Vehicle 4 was equipped with a sonar semi-ring and contact sensors only (shown at 1). Sonar input was provided to the perceptron DA input layer in a slightly modified manner. Instead of encoding near, middle and far distances, only a single distance measure was encoded--called near. An object was near if it was within three steps of the sensor. In this case the input unit was provided a signal of 1.0, otherwise the signal was 0.0.

The perceptron filtered the input through its weights and produced an output which was interpreted (at 3) and appropriate control signals to a comparator. The sonar sensors were also connected to the RRNN RI which also interpreted their output and sent its control signals to the comparator as well (shown at 4). This also occurred with the collision avoidance RI (at 5).

The comparator actually implemented the RIDA cascading hierarchy by comparing the control signals of the DA with those of the RRNN RI and the collision avoidance RI. If the DA control signals were different it was assumed to be wrong and the comparator would generate an appropriate training set based on either the RRNN RI or the other RI. If these two disagreed the collision avoidance RI was assumed to be correct. In this way the hierarchy was maintained. Although somewhat unrealistic, the training set was used in an off-line training session for the DA. In addition the comparator would pass the appropriate control signals to the vehicle's motors (shown at 6).

The RIDA architectural diagram is shown in figure D-6.

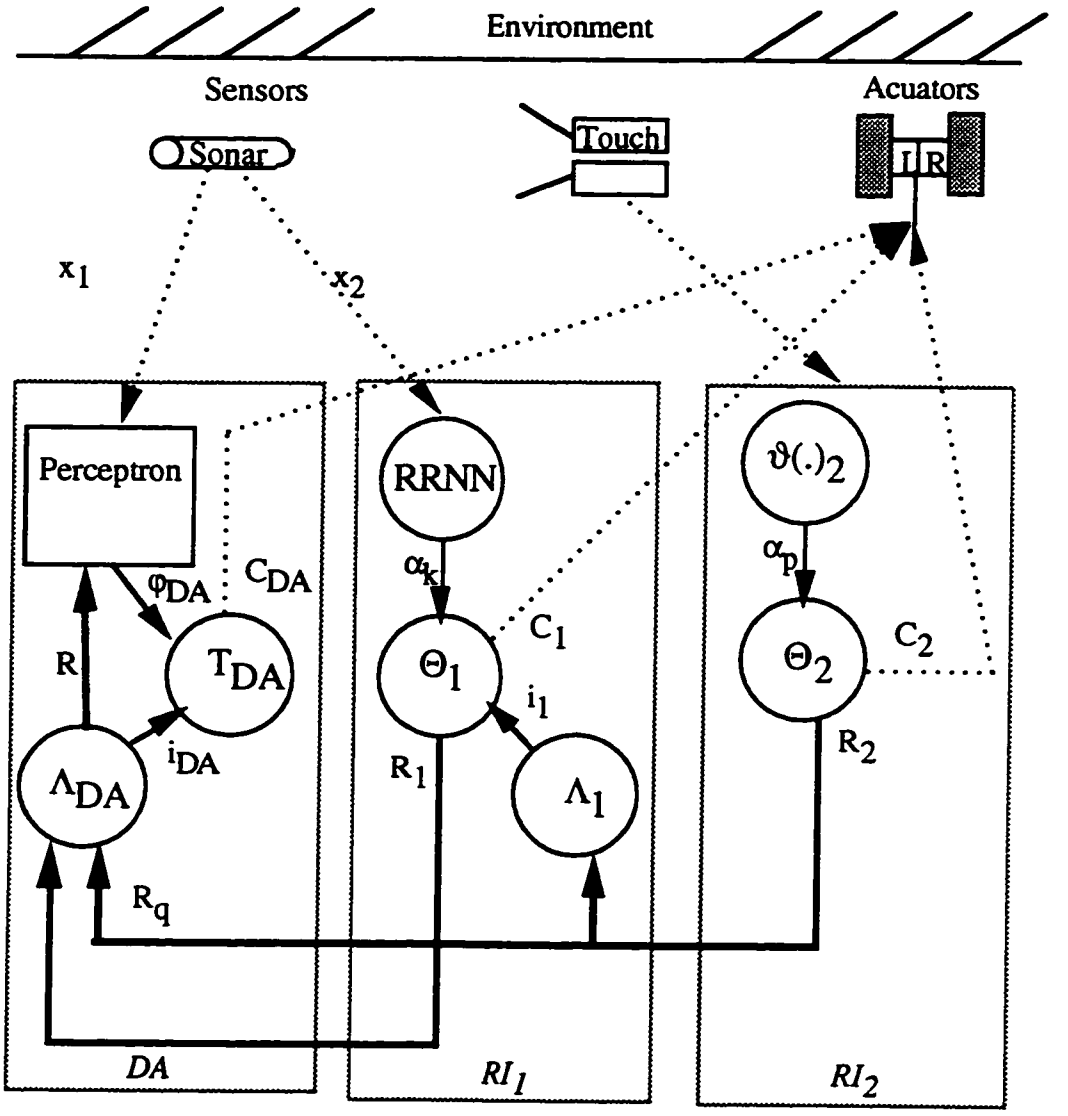


Figure D-6 Vehicle 4 RIDA Architecture

Note that the hierarchy maintained by the comparator is illustrated here.

Bibliography

Akaike, H. (1974), "A new look at the statistical model identification", Annals of the Institute of Statistical Mathematics 22, 202-217.

Ando, Y., & Yuta, S. (1995), "Following a Wall by an Autonomous Mobile Robot with a Sonar - Ring", Proc of the IEEE International Conf on Robotics and Automation, 2599-2606.

Antsaklis, P.J., & Passino, K.M. (1989), "Towards Intelligent Autonomous Control Systems: Architecture and Fundamental Issues", Journal of Intelligent and Robotic Systems 1, 315-342.

Barto, A.G. (1992), "Reinforcement learning and adaptive critic methods", In Handbook of Intelligent Control, D.A. White and D.A. Sofge (eds), 469-491.

Barto, A.G., & Anandan, P. (1985), "Pattern Recognizing Stochastic Learning Automata", IEEE Transactions on Systems, Man and Cybernetics 15, 360-375.

Barto, A.G. (1985), "Learning by Statistical Cooperation of self-interested neuron-like computing elements", Human Neurobiology 4, 229-256.

Barto, A.G., Sutton, S., & Anderson, C.W. (1983), "Neuronlike adaptive elements that can solve difficult learning control problems", IEEE Transactions on Systems, Man and Cybernetics SMC, 5834-46.

Beer, R.D., & Chiel, H.J. (1990), "The Neural Basis of Behavioral Choice in an Artificial Insect", Proc of the 1st intl conf on simulation of adaptive behavior, 247-254.

Benn, S. (1976), "Freedom, Autonomy and the Concept of a Person", Proc of the Aristotelian Society Vol. LXXVI, 124.

Bonner, J.T. (1980), The Evolution of Culture in Animals, Princeton University Press.

Booth, D.A. (1978), "Prediction of feeding behavior from energy flows in the rat". In Hunger Models: Computable Theory of Feeding Control, D.A. Booth (ed), Academic Press.

Braitenberg, V. (1986), Vehicles, Experiments in Synthetic Psychology, MIT press.

Brooks, R. A. (1986), "A robust layered control system for a mobile robot", IEEE Journal of Robotics and Automation RA-2, 14-23.

Brooks, R. A. (1987), "Intelligence without representation", Artificial Intelligence vol 47, 139-159.

Brooks, R.A. (1989), "A Robot that Walks: Emergent Behaviors from a carefully Evolved
Network", Neural Computation vol 1:2, 355-362.

Brooks, R.A. (1990), "Elephants Don't Play Chess", Robotics and Autonomous Systems 6, 3-15.

Brooks, R.A. (1991), "New Approaches to Robotics", Science, vol 253, 13 Sept, 1227-1232.

Brooks, R. A. (1991), "How to build complete creatures rather than isolated cognitive simulators", in Architectures for Intelligence, K. VanLehn (ed), Lawrence Erlbaum Associates, Hillsdale, NJ, 225-239.

Brooks, R. A. (1991), "Integrated Systems Based on Behaviors" SIGART Bulletin 2, 46-50.

Bryson, A.E. & Ho, Y.C. (1969), Applied Optimal Control, Blaisdell, New York.

Delcomyn, F. (1993), "The Walking of Cockroaches--Deceptive Simplicity", in Biological Neural Networks in Invertebrate Neuroethology and Robotics, R.D.Beer, R.E. Ritzmann , & McKenna, (ed), Academic Press.

Dearden, R.F. (1972), "Autonomy and Education" in Education and the Development of Reason, R.F. Dearden, P.H. Hirst & R.S. Peters (ed), Routledge and Kegan Paul, London, 453.

Descartes, R. (c1989), The Passions of the Soul, translated and annotated by Stephen Voss, Hackett Pub. Co., Indianapolis.

Dewey, J. (1963), Freedom and Culture, Capricorn, New York.

Dorf, R.C. (1992), Modern Control Systems, Addison-Wesley, New York.

Dorigo, M. (1993), "Genetic-Based Machine Learning and Behavior-Based Robotics: A New Synthesis", IEEE Transactions on Systems, Man and Cybernetics vol 23:1 (January), 141-154.

Downie, R.S., & Telfer, E. (1971), "Autonomy", Philosophy vol 15, 301.

Dworkin, G. (1988), The theory and Practice of Autonomy, Cambridge studies in philosophy.

Elgersma, S. (1994), 27-642 Artificial Neural Networks Project Documentation, Natural Selection Group Project Documentation, University of Guelph.

Elsley, R. (1988), "A learning architecture for control based on back-propagation neural networks", IEEE Conf. on Neural Networks vol. 2, 584-587.

Fagg, A.H., Lotspeich, D. & Bekey, G.A. (1994), "A reinforcement Learning Approach to Reactive Control Policy Design for Autonomous Robots", in Proceedings of the IEEE Conf. on Robotics and Automation, 39-44.

Fagg, A.H., Lotspeich, D., Hoff, J. & Bekey, G.A. (1994), "Rapid Reinforcement Learning for Reactive Control Policy Design in Autonomous Robots", submitted draft to WCNN.

Fagg, A.H., & Arbib, M.A. (1992), "A model of primate visual-motor conditional learning", Journal of Adaptive Behaviour vol 1(1), 3-37.

Fearing, F. (1930), Reflex Action: A Study in the History of Physiological Psychology, Williams & Wilkins, Baltimore.

Feinberg, J. (1971), "The Idea of a Free Man," in Education and Development of Reason, R.F. Dearden, P.H. Hirst & R.S. Peters (ed), Routledge and Kegan Paul, London, 161.

Fischer, H. P. (1988), Reflex and Reflexes.

Glauber, R.J. (1963), "Time-Dependent Statistics of the Ising Model", Journal of Mathematical Physics 4, 294-307.

Goldstein, J. (1978), "On being Adult and Being an Adult in Secular Law," in Adulthood, E. H. Erikson (ed), W.W. Norton and Co., New York, 252.

Goodridge, S.G. & Luo, R.C. (1994), "Fuzzy Behavior Fusion for Reactive Control of an Autonomous Mobile Robot: MARGE", IEEE International Conf on Robotics and Automation, 1622-1627.

Green, P. (1993), "How to Watch Your Step: Biological Evidence and an Initial Model", in From Animals to Animats 3, Proceedings of the Third International Conference on Simulation of Adaptive Behavior, D. Cliff, P. Husbands, J. Meyer & S.W. Wilson (ed), MIT Press.

Griffin, D.R. (1984), Animal Thinking, Harvard University Press.

Hart, A.P., Nilsson, H.J. & Raphael, B. (1968), "A formal basis for the heuristic determination of minimum cost paths", IEEE Trans. Systems Sci. and Cybern 4.

Haykin, S. (1991), Adaptive Filter Theory, 2nd ed. Englewood Cliffs, NJ: Prentice Hall.

Hertz, J., Krogh, A. & Palmer, R.G. (1991), Introduction to The Theory of Neural Computation, Addison-Wesley.

Hinton, G.E. (1987), "Connectionist Learning Procedures", Computer Science Department, Carnegie-Mellon University. Technical Report CMU-CS-87-115.

Hopcroft, J.E. & Ullman, J.D. (1979), Introduction to Automaton Theory, Languages and Computation, Addison-Wesley.

Hopkin, D. & Moss, B. (1976), Automata, MacMillan Press Ltd.

Iberall, T. (1987), "A ballpark approach to modelling human prehension", IEEE Conf on Neural Networks vol. 4, 535-544.

Igarashi, E., Sato, K., Okada, S., Hozumi, H., Shimada, H., Okano, H., & Ozaki, O. (1995), "Hierarchical Autonomous Mobile Control System of a Patrol Robot for Nuclear Power Plants", IEEE International Conf on Robotics and Automation, 837-842.

James, W. (1890), Principles of Psychology, H. Holt, New York.

Jones, J.L. & Flynn, A.M. (1993), Mobile Robots--Inspiration to Implementation, A.K. Peters publishers.

Jorgenson, C.C. (1987), "Neural network representation of sensor graphs in autonomous robot path planning", IEEE Conf. on Neural Networks vol. 4, 507-516.

Kaelbling, L.P., Littman, M.L. & Moore, A.W. (1996), "Reinforcement Learning: A Survey", Journal of Artificial Intelligence Research 4, 237-285.

Kaspar, P. (1994), "NSI Supports Dante II Robot Experiment in Alaska", Ames Research Centre Technical Report.

Klopf, A.H. (1982), The Hedonistic Neuron, Hemisphere, Washington D.C., Chapter 7.

Leahay, T. H., & Harris, R. S. (1993), Learning and Cognition, 3d ed, Prentice Hall, New Jersey.

Lin, L. (1992), "Self-Improving Reactive Agents Based on Reinforcement Learning, Planning and Teaching", Machine Learning 8, 293-321.

Lucas, J.L. (1966), Principles of Politics, Oxford University Press, 101.

Maes, P. (1994), "Agents that reduce work and information overload" Communications of the ACM vol 37:7, July, 30-39.

Mataric, M. J. (1991), "Behavioral synergy without explicit integration", SIGART Bulletin 2, 85-88.

Mataric, M. J. (1992), "Behavior-based Systems: Main Properties and Implications", Proceedings, IEEE International Conference on Robotics and Automation, Workshop on Architectures for Intelligent Control Systems, Nice, France, 46-54.

Manning, A. (1979), An Introduction to Animal Behavior 22, 3rd ed. Addison-Wesly Publ. Co., Massachusetts.

Martinez, J. L., and Kesner, R. P., (ed) (1991), Learning and Memory: A Biological View, 2d ed, Academic Press, Orlando.

McFarland, D. (1981), The Oxford Companion to Animal Behavior, Oxford Universtiy Press.

Meystel, A. (1991), Autonomous Mobile Robots-Vehicles with Cognitive Control, World Scientific, Singapore.

Minsky, M. & Papert, S. (1969), Perceptrons, MIT Press, Cambridge.

Mitsumoto, N., Fukuda, T., Shimojima, K., & Ogawa, A. (1995), "Micro Autonomous Robotic System and Biologically Inspired Immune Swarm Strategy as a Multi Agent Robotic System", IEEE International Conf on Robotics and Automation, 2187-2192.

Moravec , H.P. (1981), Robot Rover Visual Navigation, UMI Res. Press.

Najand, S., Lo, Z. & Bavarian, B. (1992), "Application of Self-Organizing Neural Networks for Mobile Robot Environment Learning", in Neural Networks in Robotics, G.A. Bekey and K.Y. Goldberg (ed), Kluwer Academic Press.

Nehmzow, U., Hallam, J. & Smithers, T. (1989), "Really Useful Robots", Intelligent Autonomous Systems 1, T. Kanade, F.C.A Graen & L.O. Hertzberger (ed) , 284-293,

Nehmzow, U., Smithers, T. & McGonigle, B. (1993), "Increasing the Behavioural Repertoire in a Mobile Robot", From Animals to Animats, Proceedings of the Second International Conference on Simulation of Adaptive Behavior, J Meyer, H.L. Roitblat and S.W. Wilson (ed), MIT Press, 291-297.

Pati, Y. (1988), "Neural networks for tactile perception", IEEE Conf. on Robotics and Automation, 134-139.

Pearson, K.G. (1976), "The Control of Walking", Scientific American vol 235, 72-86.

Peretto, P. (1984), "Collective Properties of Neural Networks: A Statistical Physics Approach". Biological Cybernetics vol 50, 51-62.

Peters, R.S. (1972), "Freedom and Development of the Free Man", in Education and the Development of Reason, R.F. Dearden (ed), Routledge and Kegan, London, 130.

Ram, A., Arkin R., Boone, G. & Pearce, M. (1994), "Using Genetic Algorithms to Learn Reactive Control Parameters for Autonomous Robotic Navigation", Adaptive Behavior vol 2:3, 277-304.

Rashotte, M.E. (1985), "Behavior in Relation to Objects in Space: Some Historical Perspectives" in Cognitive Processes and Spatial Orientation in Animals and Man, Ellen, C. Thinus-blanc (eds), vol 1, NATO ASI series #36, 39-54.

Rawls, J. (1971), A Theory of Justice, Harvard University Press, Cambridge, 516.

Rissanen, J. (1978), "Modeling by shortest data description", Automatica 14, 465-471.

Rosenblatt, F. (1962), Principles of Neurodynamics, Spartan, New York.

Rumelhart, D.E., Hinton, G.E. & Williams, R.J. (1986), "Learning representations by back-propagating errors", Nature (London), 323, 533-536.

Saito, H., Sugiura, H. & Yuta, S. (1995), "Development of Autonomous Dump Trucks System (HIVACS) in Heavy Construction sites", IEEE International Conf on Robotics and Automation, 2524-2529.

Santos, A., Rives, P., Espiau, B. & Simon, D. (1995), "Dealing in Real Time with A Priori Unknown Environment on Autonomous Underwater Vehicles (AUVs)", IEEE International Conf on Robotics and Automation, 1579-1584.

Scanlon, T. (1972), "A Theory of Freedom of Expression", Philosophy and Public Affairs vol 1, 215.

Shavlik, T.W. (1996), "Providing advice to Neural Network Agents that Learn from Reinforcement", http://www.stats.wisc.edu/~yandell/help/satellite (www).

Shepherd, G.M. (1988), Neurobiology, Oxford University Press, New York.

Snaith, M. & Holland, O. (1991), "Quadrupedal Walking Using Trained and Untrained Neural Models", TAG technical Report.

Stacey, D.A. (1994), The University of Waterloo, Personal Communication concerning her implementation of an ARP network at the University of Washington.

Sutton, R.S., Barto, A.G. and Williams, R.J. (1991), "Reinforcement learning is direct adaptive optimal control". Proceedings of the American Control Conference, Boston, 2143-2146.

Sutton, R.S. (1988), "Learning to Predict by the Method of Temporal Difference", Machine Learning vol 3, 9-44.

Toates, F.M., & Oatley, K. (1970), "Computer simulation of thirst and water balance", Medical and Biological Engineer 8, 71-87.

Varela, F.J. (1979), Principles of Biological Autonomy, The North Holland series in general systems research, Elsevier North Holland Inc.

Watkins, C.J.C.H & Dayan, P. (1992), "Q-learning", Machine Learning, vol 8:3, 279-292.

Widrow, B., & Hoff, M.E. (1960), "Adaptive Switching Circuits", WESCON Conv. Record, Part IV, 96-104.

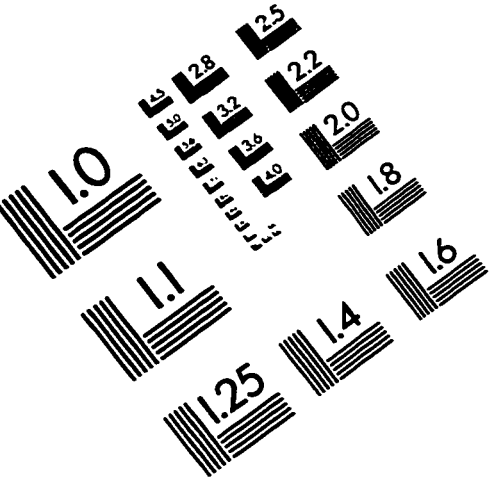
Widrow, B., & Smith, F.W. (1964), "Pattern-recognizing control systems", Computer and Information Sciences, J.T. Tow & R.H. Wilcox (ed), Clever Hume Press, 288-317.

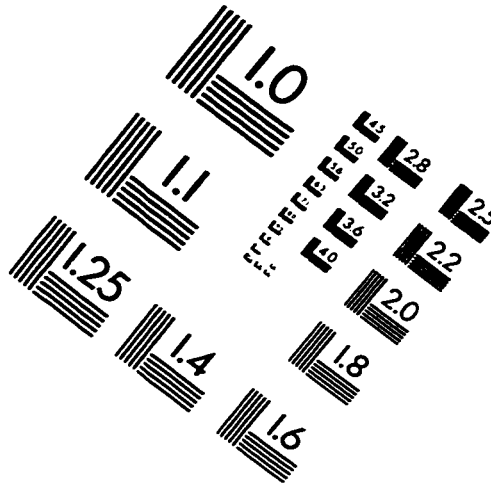
Wolff, R. (1970), In Defense of Anarchism, Harper and Row, New York, 14.

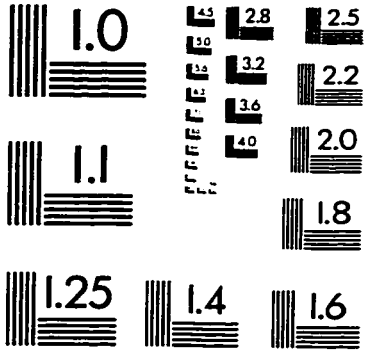
Zanichelli, F., Caselli, S., Natali, A. & Omicini, A. (1994), "A multi-agent framework and programming environment for autonomous robotics", IEEE International Conf on Robotics and Automation, 3501-3507.

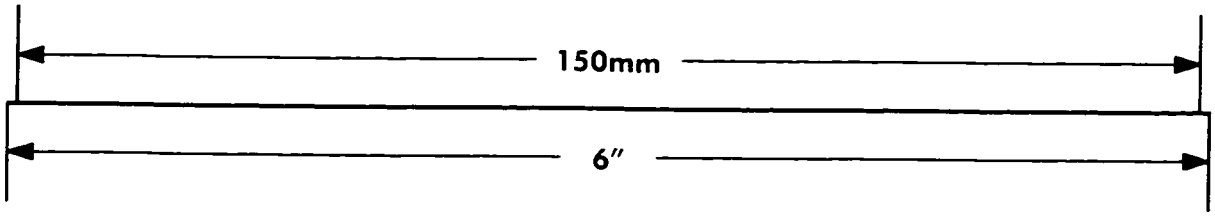
Zapata R., Lepinay, P., Novales, C. & Deplanques, P. (1993), "Reactive Behaviors of Fast Mobile Robots in Unstructured Environments: Sensor-based Control and Neural Networks", in From Animals to Animats, Proceedings of the Second International Conference on Simulation of Adaptive Behavior, J Meyer, H.L. Roitblat and S.W. Wilson (ed), MIT Press, 108-115.

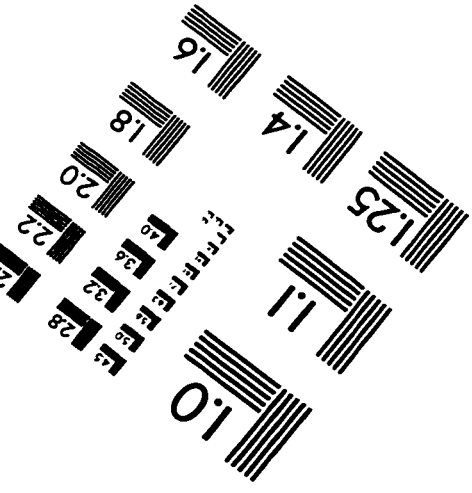
IMAGE EVALUATION
TEST TARGET (QA-3)

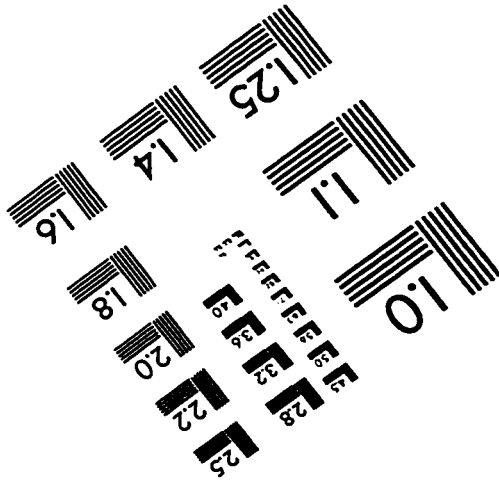












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989