Modelling Affordances for the Control and Evaluation of Intrinsically Motivated Robots

Kathryn Merrick
School of Engineering and Information Technology
University of New South Wales
Australian Defence Force Academy
k.merrick@adfa.edu.au

Thomas Scully
School of Engineering and Information Technology
University of New South Wales
Australian Defence Force Academy
t.scully@student.adfa.edu.au

Abstract

In psychological theory, affordances provide a way to describe an environment in terms of the opportunities it provides an organism to act. Affordance-based models have been applied to robotics in areas such as tool-use, interaction and vision, as an alternative to hybrid control architectures. This paper introduces a model of affordances for controlling and evaluating intrinsically motivated robots. We present an implementation and experimental evaluation of the model on four critter-bots using the *Lego Mindstorms NXT* platform. Results show that the critter-bots can evolve measurable, structured behaviour cycles adapted to their individual physical structures.

1 Introduction

Intrinsically motivated robots use computational models of motivation – such as novelty [Huang and Weng, 2007], interest [Merrick and Huntingon, 2008] or curiosity [Oudeyer et al., 2007] – to select salient environmental stimuli on which to focus their attention. Their capacity for autonomous, open-ended goal-selection gives intrinsically motivated robots the potential to adapt to unexpected changes in their environment. In addition, they have the potential to develop novel or creative behaviours not explicitly programmed by their engineers.

Two key challenges developing intrinsically motivated robots are the design of the motivation module and the evaluation of the robot's emergent behaviour. The motivation module must be open-ended enough to inspire exploration, but focused enough to permit the robot to exploit salient discoveries and evolve structured behaviour.

Evaluation techniques for intrinsically motivated robots must also be open-ended enough to identify any emergent behaviours, but specific enough to quantify those behaviours in a meaningful way [Merrick, 2009]. Evaluation is difficult for intrinsically motivated robots because these robots can select and change their own goals. This means that traditional, task-oriented evaluation can be inappropriate as there is no fixed set of 'correct' tasks to be addressed.

This paper explores an approach to these two challenges based on affordances. We begin in Section 2 by surveying literature describing affordances in robotic control and evaluation. Section 3 presents a generic model of affordances for use in robots. This model is then used to develop both a control system for an intrinsically motivated robot and an evaluation approach for such robots. Section 4 describes an implementation of the affordance-based control model on four critter-bots using the *Lego Mindstorms NXT* platform. The affordance-based evaluation model is then used to measure the performance of the control model. Results show that the critter-bots can evolve measurable, structured behaviour cycles adapted to their individual physical structures. Section 5 concludes with a discussion of the future directions for this work.

2 Affordances

The concept of affordances [Gibson, 1979] is an approach to understanding visual perception in natural systems. The underlying theory is that organisms perceive their environment, or objects in their environment, in terms of the opportunities those objects provide for the organism to act. That is, the meaning of an object is closely related to an organism's motor intent.

While there is no single definition or notation for affordances in robotics, the concept has been considered as an approach to a range of robotic problems [Rome et al., 2008a]. The following sections consider this existing work in two categories: affordance-based robotic control and affordance-based evaluation.

2.1 Affordance-Based Control

Affordance-based robotic control has been considered in some depth by the MACS[1] project. The central hypothesis of MACS is that affordance-based control architectures will allow robots to perceive more interaction possibilities than traditional architectures relying on appearance-based object recognition alone [Rome et al., 2008b]. The aim of the MACS project is to build robots that can adapt more readily in problem domains such as home or industrial automation.

The MACS hypothesis has also been explored beyond the MACS project in areas including tool-use [Stoytchev, 2005], interaction [Hafner and Kaplan, 2008], machine vision [Paletta and Fritz, 2008; Modayil and Kuipers, 2008] and navigation [Modayil and Kuipers, 2008; Hertzberg et al., 2008]. Our paper extends existing work to the problem of intrinsically motivated robots and their evaluation.

Tool-Use

Stoytchev [2005] presents a behaviour-grounded approach to learning tool affordances in robots. He argues that the notion of affordances is well suited to implementing tool-use because it takes into account not only the object being used as a tool, but the robot that will use it. That is, different robots may perceive different affordances for an object, based on their physical structure. For example, a particular object should only be perceived as a tool by a robot equipped with the correct actuators to grasp and manipulate it.

Vision and Perception

Due to the close relationship between vision, perception and affordances, affordances provide a strong foundation for the exploration of machine vision, perception and acquisition of object knowledge. A number of examples of work in this area use learning techniques so that robots can learn affordances through interaction with their environment [Paletta and Fritz, 2008; Modayil and Kuipers, 2008]. The advantage of this approach, as with the affordances in tool-use, is that the knowledge produced is grounded on the experiences of the robot. This type of grounded knowledge has been found to be more flexible and more useful than human generated knowledge, as it is specifically tailored around the experiences of the robot [Modayil and Kuipers, 2008].

This paper also uses a learning approach in conjunction with affordances. However, our robots learn which affordances to act on, rather than which affordances are available.

Navigation

Affordances have also been explored with relation to navigation tasks such as simultaneous localisation and mapping (SLAM) [Modayil and Kuipers, 2008] and navigation in 2D and 3D spaces [Hertzberg et al., 2008]. Results in this area have been mixed. Modayil and Kuipers [2008], for example, report the benefits of affordance-based experiential knowledge in SLAM, while Hertzberg et al. [2008] conclude that affordance-based representations do not benefit the navigation problem.

[1] Multisensory Autonomous Cognitive Systems interacting with dynamic environments for perceiving and using affordances. European Commission Sixth Framework Programme

Interaction

In a different domain, Hafner and Kaplan [2008] describe how the concept of affordances can be applied to the interactive behaviour of robots. They introduce the notion of 'interpersonal maps' as a common representation space to allow robots to compare their behaviour to that of others. The maps can be used to detect certain types of interactions such as imitation.

Hafner and Kaplan's [2008] work differs from the previous examples because it uses affordances to model the relationship between two robots, rather than between a robot and its environment. Our affordance-based model of intrinsic motivation presented in Section 3 further extends existing work by using affordances to model the internal relationship between a robot and its physical structure, as well as the relationship between a robot and its environment.

2.2 Affordance-Based Evaluation

In addition to affordance-based control, some work has also considered affordances as a means of evaluating the behaviour of robots. Oudeyer et al. [2007], for example, use the idea of 'affordant' and 'non-affordant' behaviour for a particular task to evaluate an intrinsic motivation system for autonomous mental development in a robot. This allows them to evaluate the success of their system in terms of the increase in affordant behaviours and the decrease in non-affordance behaviours for a particular task.

In this paper we present a more general approach in which the performance of a robot is characterised in terms of its ability to execute or act on affordances in structured, cyclic patterns. This allows evaluation of the emergent behaviour of a robot independent of a specific task.

3 Modelling Affordances

In robotics, affordances describe an environment in terms of the opportunities it provides a robot to act. That is, affordances can be thought of as mappings or relationships between some aspect of the robot's environment – such as an object [Stoytchev, 2005; Modayil and Kuipers, 2008] or another agent [Hafner and Kaplan, 2008] and the actions a robot can perform.

This paper focuses on the relationship between the actions a robot can perform, its physical structure and its external environment. Thus, while affordances in natural systems are usually associated with visual perception, in our model we extend the idea to include proprioception. This permits a robot to reason about actions enabled not only by its external environment but also by its own physical structure using the same processes. More formally, a robot's state $S_{(t)}$ at time t is described by its internal state $S_{I(t)}$ and its external state $S_{E(t)}$:

$$S_{(t)} = S_{I(t)} + S_{E(t)}$$

This paper assumes an attribute-based representation $S = (s_1, s_2, s_3, \ldots)$. The set $\mathbf{A}$ of actions afforded by a state S at time t is:

$$F(S_{(t)}) = \mathbf{A}$$

Because a robot's sensor data, and thus the values of its sensory attributes, are often continuous and noisy, we

further introduce the idea of a situation. A situation Φ is an attribute-based cluster centre that generalises over all states within a radius ρ.

The set of actions afforded by a situation Φ is the union of the sets of actions afforded by all states satisfying the validation constraint:

$$\text{dist}(S, \Phi) < \rho$$

For example, using an attribute-based state representation the validation constraint could be Euclidean distance between a normalised state and a normalised situation vector. So the set of actions afforded by a situation Φ at time *t* is:

$$F(\Phi_{(t)}) = \bigcup_{S|\text{dist}(S,\Phi_{(t)})<\rho} F(S)$$

This notation implies that actions afforded by a situation are determined by both the situation itself and the time at which the situation occurs.

While a robot may perceive, or learn to perceive, a number of affordances in any situation, its emergent behaviour is defined by which of these affordance it chooses to act on or execute at each time-step. We denote the affordance executed at time *t* by:

$$X_{(t)} = \{\Phi, A\}$$

The difference between perception and execution of affordances is illustrated in Figure 1.

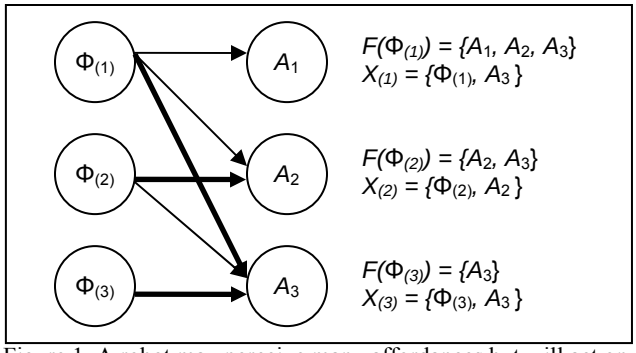

Figure 1. A robot may perceive many affordances but will act on only one. Each line represents a perceived affordance. Bold lines indicate the affordance acted upon.

3.1 Affordances, Intrinsic Motivation and Reinforcement Learning

The choice of which affordance to execute may be determined in different ways including use of pre-programmed rules, planning, mimicry or trial-and-error. In this paper, robots perceive a fixed set of affordances in each state. They then use a motivated reinforcement learning (MRL) approach to select which affordances to act on. The intrinsic motivation used is motivation to repeat execution of selected affordances as cycles. The aim is to build robots that can experiment with their physical structure and develop structured behaviour cycles adapted to their individual shape. The approach is inspired by the behaviour cycles evident in natural systems [Ahlgren and Halberg, 1990], which permit them to synchronise their behaviour with their environment for survival, efficiency or evolutionary advantage.

Reinforcement Learning

Reinforcement learning (RL) and intrinsically motivated reinforcement learning (MRL) are well suited as affordance-based control models because their basic approach is to develop affordance-like mappings between states, actions and a value describing the utility of taking a given action in a particular state. Utility is generally a function of a reward signal provided to the robot for encountering certain states or performing certain actions.

In Q-learning [Watkins and Dayan, 1992], for example, utility is a function of past, present and future expected reward. Utility-values are accumulated in a policy mapping each state to each afforded action and a Q-value representing the learned utility. For example, the utility of taking action *A* in state *S* is $Q(S, A)$. In this paper, we redefine Q-values as $Q(\Phi, A)$ or the utility of taking an action *A* in a situation Φ.

MRL is distinguished from RL by the use of a dynamic, experiential motivation function to compute reward. One such intrinsic motivation function, which rewards cycles of affordances, is discussed in the next section.

Affordance Cycles as Intrinsic Motivation

Natural systems have behaviour cycles such as the circadian rhythm [Dunlap et al., 2003], learning cycles [Kolb et al., 1984] and socio-demographic cycles [Nefadov, 2004] to name a few. Biocycles such as the circadian rhythm are of particular relevance to developmental robotics because they suggest a way for robots to progressively synchronise their behaviour with their internal structure or external environment. Using the notation developed above, the simplest behaviour cycle is repeated execution of the same affordance:

$$X_1, X_1, X_1, X_1, X_1, X_1 \ldots$$

This represents a robot acting to maintain itself in a certain situation. It can be thought of as the robot repeatedly performing a maintenance task.

A more complex behaviour cycle might repeatedly execute a sequence of affordances:

$$X_1, X_2, X_3, X_1, X_2, X_3, X_1, X_2, X_3, \ldots$$

This represents a robot acting to cause change. It can be thought of as the robot repeatedly performing a sequence of achievement tasks.

To motivate learning of behaviour cycles using MRL, a reward signal is required that rewards the robot for repeatedly performing maintenance or achievement tasks. Our basic approach identifies two properties of affordances to reward:

- The affordance must be repeatable;
- Repetition must cause the agent to learn.

A MRL robot can identify repeatable affordances by querying its policy for the utility-value of the situation-action pair comprising the affordance. Likewise, the MRL robot can identify affordances that cause learning by analysing the change in utility-value (learning error) when it updates the policy with a given situation-action pair. A reward-signal using these ideas is:

$$R_{(t)} = \begin{cases} 1 & \text{if } \exists \tau \text{ such that } X_{(t-\tau)} = X_{(t-1)} \text{ and} \\ (\Delta Q(\Phi_{(t-1)}, A_{(t-1)}) > \varepsilon \text{ or } 0 > \Delta Q(\Phi_{(t-1)}, A_{(t-1)}) > -\varepsilon) & \\ -1 & \text{otherwise} \end{cases}$$

This reward signal assumes that utility $Q(\Phi, A)$ is

optimistically initialized at 1 and learning error $\Delta Q(\Phi, A)$ at 0, as shown in Figure 2. The first few times an affordance occurs, reward is low (–1) until the agent recognises the affordance as repeatable (indicated by a small, negative learning error). On subsequent occurrences of the affordance, reward has the higher value of 1. When learning error is very low and positive (indicating there is little being learned) reward drops back to –1. Reward increases to 1 again when learning error is small and negative, indicating that the robot has almost completely forgotten the affordance (so there is once more something to be learned).

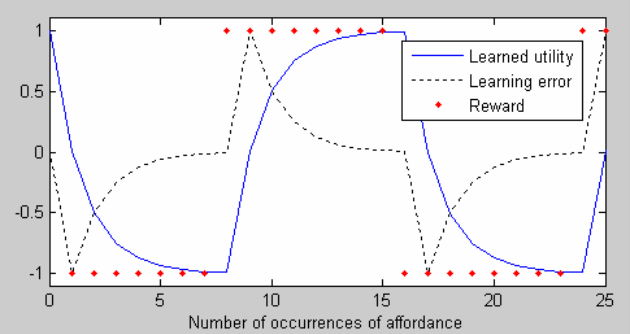

Figure 2. A reward signal that motivates repetition of affordances. For simplicity, this diagram assumes that utility is a function of past and present reward.

This reward signal motivates cycles of learning and forgetting about affordances. While the robot is forgetting about one affordance it may be learning about another, so its focus of attention changes over time. The reward signal is dynamic because any repeatedly executed affordance will only be rewarded until the robot has learned how to repeat it. The robot will then become bored and look for other affordances on which to focus its attention. As the robot learns it will exhibit periods of exploitation of learned, cyclic behavioural policies, interspersed with periods of exploration to discover new situations that can become parts of such cycles.

Motivated Reinforcement Learning using Affordances

The complete algorithm for MRL using affordances is shown in Figure 3. Initially the robot has an empty policy with no situation-action mappings. At each time-step, the robot senses the current state $S_{(t)}$. If $S_{(t)}$ does not satisfy the validation constraint for any existing situation, a new situation is created, centred on $S_{(t)}$ and entries added to the policy for all afforded actions. Otherwise, the closest existing situation $\Phi_{(t)}$ that satisfies the validation constraint is updated to bring it closer to $S_{(t)}$, using the update:

$$\Phi_{(t)} \leftarrow \Phi_{(t)} + \eta \, \mathrm{dist}(\Phi_{(t)}, S_{(t)})$$

η is the learning rate for situations. The policy is also updated to include any actions afforded by $S_{(t)}$ not already associated with $\Phi_{(t)}$.

Reward is then computed using an intrinsic motivation function based on that illustrated in Figure 2. The reward signal in Figure 2 will motivate either maintenance tasks or achievement tasks. In the experiments in this paper we focus on achievement tasks by including an additional negative reward component of –10 if a situation is repeated at adjacent time-steps.

Next, the Q-learning update is computed as a function of past, present and future expected reward:

$$Q(\Phi_{(t)}, A_{(t)}) \leftarrow Q(\Phi_{(t)}, A_{(t)}) + \Delta Q(\Phi_{(t)}, A_{(t)})$$

where:

$$\Delta Q(\Phi_{(t)}, A_{(t)}) = \alpha[R_{(t)} + \gamma \max_{A \in F(\Phi_{(t+1)})} Q(\Phi_{(t+1)}, A) - Q(\Phi_{(t)}, A_{(t)})]$$

α governs the rate of change in Q, γ is the discount factor for future expected reward and $R_{(t)}$ is the reward at time t. α is further divided into two parameters, α_L and α_F, governing the rate of learning (increase in utility when $R_{(t)} > 0$) and forgetting (decrease in utility when $R_{(t)} \leq 0$) respectively. Differentiating between learning and forgetting allows the robot to trade-off between exploration and exploitation. For example, if the learning rate is faster than the forgetting rate the robot can continue to exploit learned behaviour while exploring to discover new affordances.

repeat (for $t > 0$):
  sense $S_{(t)}$
  find $\Phi_{(t)} = \underset{\Phi}{\mathrm{argmin}}\ (\mathrm{dist}(\Phi, S_{(t)}))$ and $d = \min_{\Phi} \mathrm{dist}(\Phi, S_{(t)})$
  if ($d < \rho$)
    update $\Phi_{(t)} \leftarrow \Phi_{(t)} + \eta \, \mathrm{dist}(\Phi_{(t)}, S_{(t)})$
    ensure $Q(\Phi_{(t)}, A)$ and $\Delta Q(\Phi_{(t)}, A)$ initialised $\forall A \in F(S_{(t)})$
  else
    $\Phi_{(t)} = S_{(t)}$
    initialise $Q(\Phi_{(t)}, A) = 1$ and $\Delta Q(\Phi_{(t)}, A) = 0$ $\forall A \in F(S_{(t)})$
  if ($t > 1$)

$$R_{(t)} = \begin{cases} -10 & \text{if } \Phi_{(t-1)} = \Phi_{(t)} \\ 1 & \text{if } \exists \tau \text{ such that } X_{(t-\tau)} = X_{(t-1)} \text{ and} \\ & (\Delta Q(\Phi_{(t-1)}, A_{(t-1)}) > \varepsilon \text{ or } 0 > \Delta Q(\Phi_{(t-1)}, A_{(t-1)}) > -\varepsilon) \\ -1 & \text{otherwise} \end{cases}$$

    if ($R_{(t)} > 0$)
      $\Delta Q(\Phi_{(t-1)}, A_{(t-1)}) = \alpha_L[R_{(t)} + \gamma \max_{A \in F(\Phi_{(t)})} Q(\Phi_{(t)}, A) - Q(\Phi_{(t-1)}, A_{(t-1)})]$
    else
      $\Delta Q(\Phi_{(t-1)}, A_{(t-1)}) = \alpha_F[R_{(t)} + \gamma \max_{A \in F(\Phi_{(t)})} Q(\Phi_{(t)}, A) - Q(\Phi_{(t-1)}, A_{(t-1)})]$
    $Q(\Phi_{(t-1)}, A_{(t-1)}) \leftarrow Q(\Phi_{(t-1)}, A_{(t-1)}) + \Delta Q(\Phi_{(t-1)}, A_{(t-1)})$
  $A_{(t)} = \underset{A \in F(\Phi_{(t)})}{\mathrm{argmax}}\ (Q(\Phi_{(t)}, A))$
  $\Phi_{(t-1)} = \Phi_{(t)}$; $A_{(t-1)} = A_{(t)}$; $X_{(t-1)} = \{\Phi_{(t-1)}, A_{(t-1)}\}$;
  execute $A_{(t)}$

Figure 3. Intrinsically motivated Q-learning using affordances and function approximation.

Because this intrinsic motivation function inspires both explorative and exploitative behaviour, a greedy action-selection strategy is adequate. For example we use:

$$A_{(t+1)} = \underset{A_{(t+1)} \in F(\Phi_{(t+1)})}{\mathrm{argmax}} \ Q_{(t)}(\Phi_{(t+1)}, A_{(t+1)})$$

This algorithm for MRL combines the function approximation approach used in Simplified Adaptive Resonance Theory (SART) networks [Baraldi and Alpaydin, 1998] with MRL. When ρ = 0 the algorithm reduces to standard MRL without function approximation.

3.2 Affordances and Evaluation of Robot Behaviour

A common criticism of intrinsically motivated robots is the difficulty of evaluating their behaviour. Because these robots select goals in response to their changing experiences, these goals may change over time. Traditional metrics that evaluate behaviour with respect to a fixed set of known tasks thus become inappropriate.

In this paper we evaluate the behaviour of intrinsically motivated robots in terms of the affordance cycles they execute. The importance of identifying repetitive patterns – particularly cyclic patterns – in human motion can be seen in a number of studies [Li and Holstein, 2002; Kovar and Gleicher, 2004; Forbes and Fiume, 2005; Tang et al., 2008]. In this paper we adapt the point-cloud technique proposed by Tang et al. [2008] for human motion to use identifying patterns in robot motion. Tang et al. [2008] use point-cloud matrices to visualise posture similarity in motion-capture data, as shown in Figure 4. Diagonal patterns represent cycles of repeated postures. Cycles can be either continuous or distributed. Continuous cycles, represented by bottom-left to top-right diagonals, repeat a sequence of postures (states) at adjacent time periods. For example:

$$[S_1, S_2, S_3], [S_1, S_2, S_3], [S_1, S_2, S_3], [S_1, S_2, S_3] \ldots$$

Distributed cycles, represented by cross patterns, repeat a sequence of postures at intermittent time periods:

$$[S_1, S_2, S_3], [S_3, S_2, S_1], [S_1, S_2, S_3], [S_3, S_2, S_1], \ldots$$

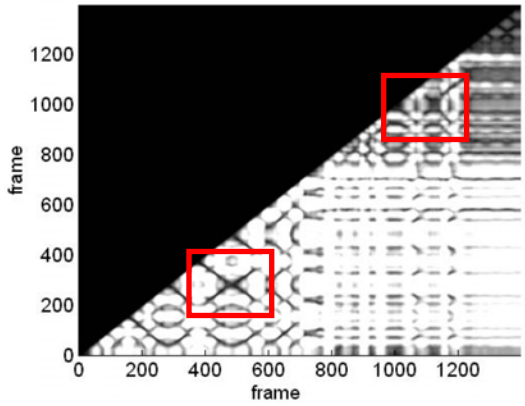

Figure 4. Tang et al. [2008] visualise human motion capture data using a point-cloud matrix showing posture similarity. Diagonal patterns, such as those highlighted, represent cyclic behaviour.

In this paper we use the point-cloud matrix to visualise the similarity between affordances executed by the robot at different times [Merrick, 2009]. The point-cloud visualization is constructed by computing the Euclidean distance $\text{dist}(X_{(t)}, X_{(t')})$ between pairs of affordances at all times t and t'. The intensity of a pixel (t, t') on the point-cloud diagram is determined by $\text{dist}(X_{(t)}, X_{(t')})$. A darker colour indicates more similar affordances.

Because small robots such as *Lego Mindstorms* tend to move faster than humans, the point-cloud visualisations tend to be denser and have various different characteristic patterns, in addition to those seen in the human motion plots. A number of important characteristic patterns are identified in the sample diagram in Figure 5:

- The dark triangles of parallel diagonals in the highlighted regions R_1 and R_3 are cycles of affordances. R_1 shows a distributed cycle and R_3 shows a continuous cycle.
- The structured patterns evident in regions R_1 and R_3 indicates that the robot is exploiting a learned cycle. In contrast, the unstructured, random pattern in region R_2 indicates that the robot is exploring to find a new cycle.
- The light rectangular region R_4 indicates that affordances in R_1 are dissimilar to those in R_3. If R_4 were dark in colour it would indicate that affordances in R_1 and R_3 were similar.
- The distance L between diagonals within a triangle indicates the cycle length. The length D of the side of a triangle indicates the cycle duration. Duration divided by cycle length gives the number of repetitions of a cycle.

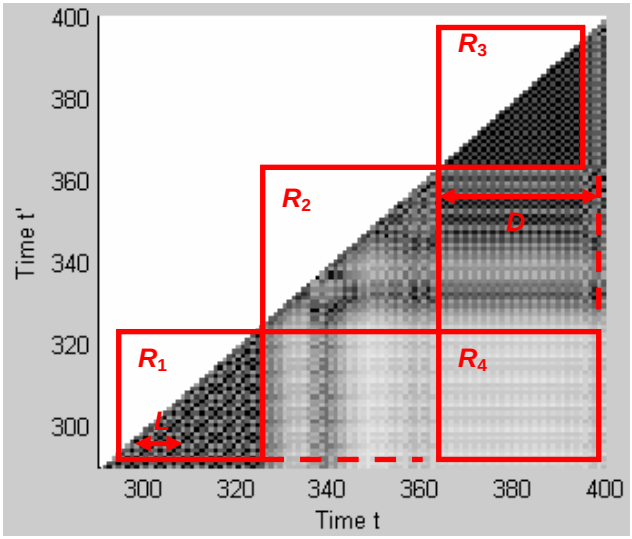

Figure 5. Four characteristic patterns found on point-cloud diagrams of affordances during robotic motion.

4 The Critter-Bot Experiment

This section describes an experiment with the affordance-based control model on four critter-bots using the *Lego Mindstorms NXT* platform. We evaluate the performance of the control model from Section 3.1 using the point-cloud diagrams described in Section 3.2.

4.1 Experimental Setup

The aim of this experiment is to show how an affordance-based model of intrinsic motivation can be used to build robots that can evolve measurable, structured behaviour adapted to their individual physical structure. We thus use four robots with different physical structures as the platform for this experiment. These are shown in Figure 6.

The first critter-bot, shown in Figure 6(a), is the simplest. It is a snail with a single motor controlling the height of its antennae. The snail can sense the rotation of the motor from its built-in tachometer, and whether the motor is moving or not. The tachometer reading is an angle between -360 and 360 degrees from the initial position of the motor. The movement reading is an enumerated set where 0 means the motor is stopped, 200 means the motor is moving forwards and 100 means the motor is moving backwards. An example sensed state for the snail is:

$$S_{(12)} = \texttt{(tacho:187.0, mov:200.0)}$$

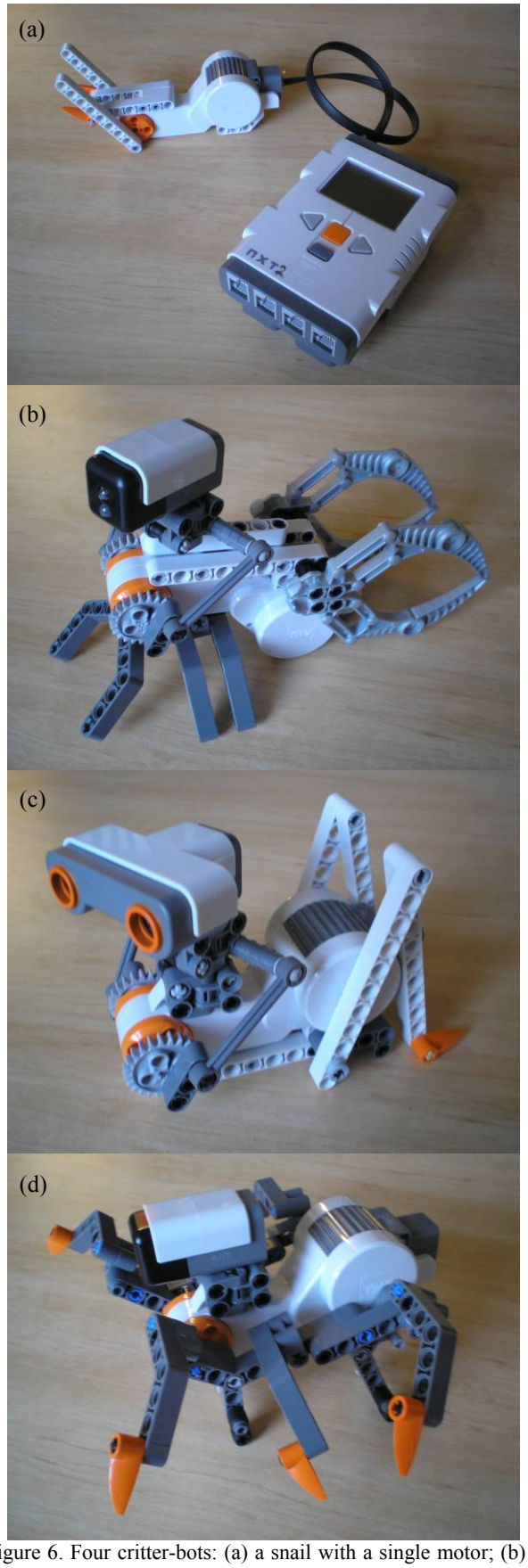

Figure 6. Four critter-bots: (a) a snail with a single motor; (b) a bee with a motor and colour sensor; (c) a cricket with a motor and ultrasonic sensor; (d) an ant with a motor and accelerometer.

The second critter-bot, shown in Figure 6(b), is a bee with a motor and colour sensor. The motor allows the bee to turn its colour sensor 'head' through 90^o – 45^o to both the left and right. The bee can sense the rotation of the motor from its built-in tachometer and whether the motor is moving or not. The colour sensor provides data describing red, blue and green intensities of the bot's environment in the direction the colour sensor is pointing. These readings range between 0 and 255. The bee was placed between two colour panels, one red and the other green as shown in Figure 7(a). An example sensed state for the bee is:

$$S_{(226)} = (\text{tacho:139.0, mov:100.0, red:0.0, green:80.0, blue:0.0})$$

Figure 6(c) shows the third critter-bot, a cricket, with a motor and ultrasonic (distance) sensor. As with the bee, the motor allows the cricket to turn its ultrasonic sensor 'head' through 45^o to both the left and right. The cricket can sense the rotation of the motor from its built-in tachometer whether the motor is moving or not. The ultrasonic sensor provides eight ping values describing the distance of any object in the direction the ultrasonic sensor is pointing. The cricket was placed in a corner such that it was further from one wall than from the other as shown in Figure 7(b). An example sensed state for the cricket is:

$$S_{(20)} = (\text{tacho4:-303.0, mov:100.0, dist0:41.0, dist1:48.0, dist2:161.0, dist3:166.0, dist4:0.0, dist5:0.0, dist6:0.0, dist7:0.0})$$

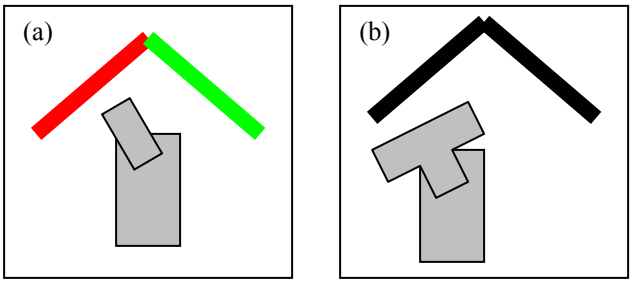

Figure 7. Experimental setup for (a) the bee and (b) the cricket.

Finally, the fourth critter-bot in Figure 6(d) is an ant with a motor and accelerometer. The motor moves the ant's legs, which can grip the surface it is on and propel the robot forwards or backwards. The ant can sense whether the motor is moving or not. In addition it can sense six values from the accelerometer. Three of these describe its acceleration in three dimensions. These values range between 0 and 981. The other three values describe the bot's tilt from the horizontal in the same dimensions. These values range from 0 to 254. An example sensed state for the ant is:

$$S_{(20)} = (\text{mov:100.0, xacc:1.0, xtilt:0.0, yacc:3.0, ytilt:0.0, zacc:19.0, ztilt:4.0})$$

Every state encountered by each critter-bot affords three actions:

- A_1 – move the motor forward at a fixed speed;
- A_2 – move the motor backwards at a fixed speed;
- A_3 – stop the motor.

All four of the critter-bots are also designed such that there are no limitations on the rotation of the motor. For

example continuous forward (or backward) motion of the motor on the snail will result in the antennas rising and falling. Likewise continuous motion of the bee's motor will oscillate the head between its left and right extremes. This sort of physical design is important for intrinsically motivated robots that can experiment with their physical structure. Without this design the robots could damage themselves unless additional logic is pre-programmed to prevent over-extension of their joints. (For example, in natural systems this 'additional logic' is a pain response.)

4.2 Results and Discussion

The affordance-based control algorithm described in Section 3.1 was implemented in Java. Due to the virtual machine and memory limitations of the *Mindstorms NXT* intelligent brick, the control algorithm was run on a PC. The intelligent brick, shown in Figure 6(a), receives commands from the PC via Bluetooth and triggers the motor. Sensor data is returned to the PC via Bluetooth.

The algorithm was run for 1,200 time-steps (approximately six minutes) on each critter-bot. Two versions of the algorithm were run, the first with ρ=0, representing standard MRL. We refer to this algorithm as MRL. The second version used ρ=0.2 so that similar states were clustered as situations. We refer to this as SART-MRL. A summary of all parameters used and their experimental values is given in Table 1.

Table 1. Experimental parameters and their values

Symbol	Description	MRL	SART-MRL
ε	Cycle forgetting threshold	0.01	0.01
α_L	Q-Learning rate	0.9	0.9
α_F	Q-Forgetting rate	0.1	0.1
γ	Discount factor	0.9	0.9
ρ	Validation threshold	0	0.2
η	Situation update rate	0	0.1

Figure 8 visualises the behaviour of a snail controlled by each version of the algorithm. Figure 8(a) shows the MRL version and Figure 8(b) the SART-MRL version. The white regions R_1 and R_2 in Figure 8(a) indicate that this robot is focusing attention on different affordances at different times. For example, inspection of the log file for this robot shows that it is focusing on affordances in states with positive tachometer readings until approximately t = 550, and states with negative tachometer readings from t = 550-850. However, while some ability to focus attention is evident in the snail using MRL, a close-up of the darker triangular region in Figure 9 reveals an absence of diagonals. This indicates that structured, cyclic behaviour is not occurring. Rather the robot is continually exploring in an effort to find a region of the environment in which it can learn.

In contrast, a close-up of some of the dark triangles for the snail controlled by SART-MRL shows a number of different diagonal patterns. Figure 10 shows two such patterns. The first pattern in R_1 is a continuous cycle of length L=5 and duration D=30. Inspection of the log file for this robot shows that the cycle is repeating actions: A_1 A_1 A_2 A_3 A_2 ... This corresponds to the snail raising its antennae twice then lowering them twice. The second pattern in R_2 has L=2 and D=49 and repeats actions: A_1 A_2 This corresponds to the snail raising its antennae once then lowering them once.

Periods of exploration occur before R_1, between the exploitative behaviour in R_1 and R_2 and after R_2 as shown in Figure 10. This demonstrates that affordance-based intrinsic motivation function can shift the robot's focus of attention so that it can learn different behaviours at different times.

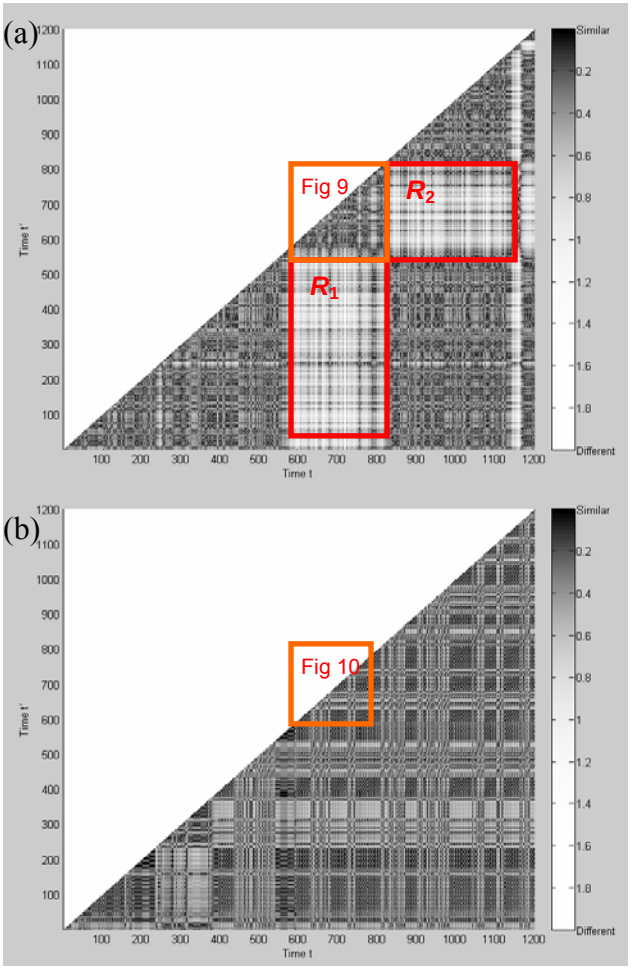

Figure 8. Point-cloud visualisations for the snail controlled by (a) MRL and (b) SART-MRL

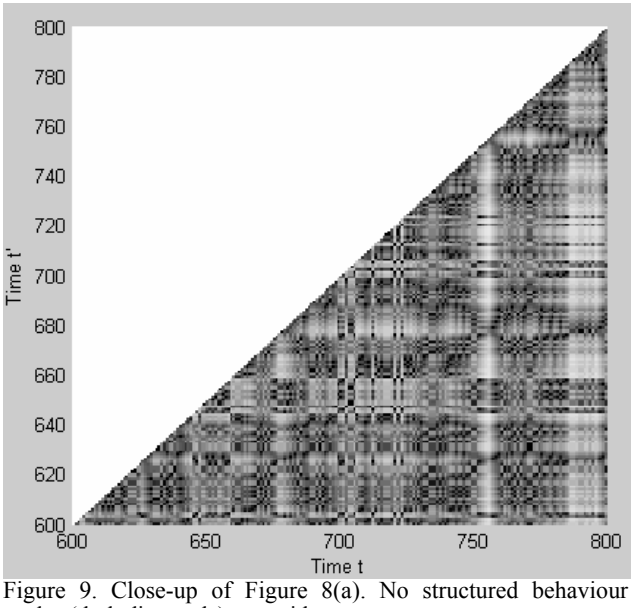

Figure 9. Close-up of Figure 8(a). No structured behaviour cycles (dark diagonals) are evident.

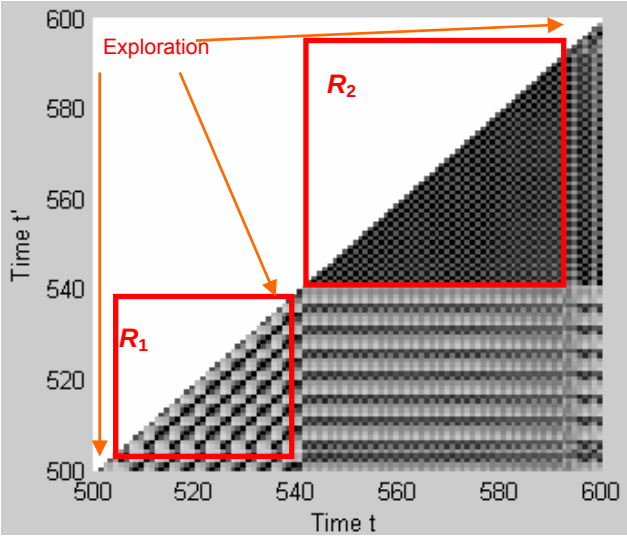

Figure 10. Close-up of Figure 8(b). Two structured behaviour cycles are evident as diagonal patterns in R_1 and R_2.

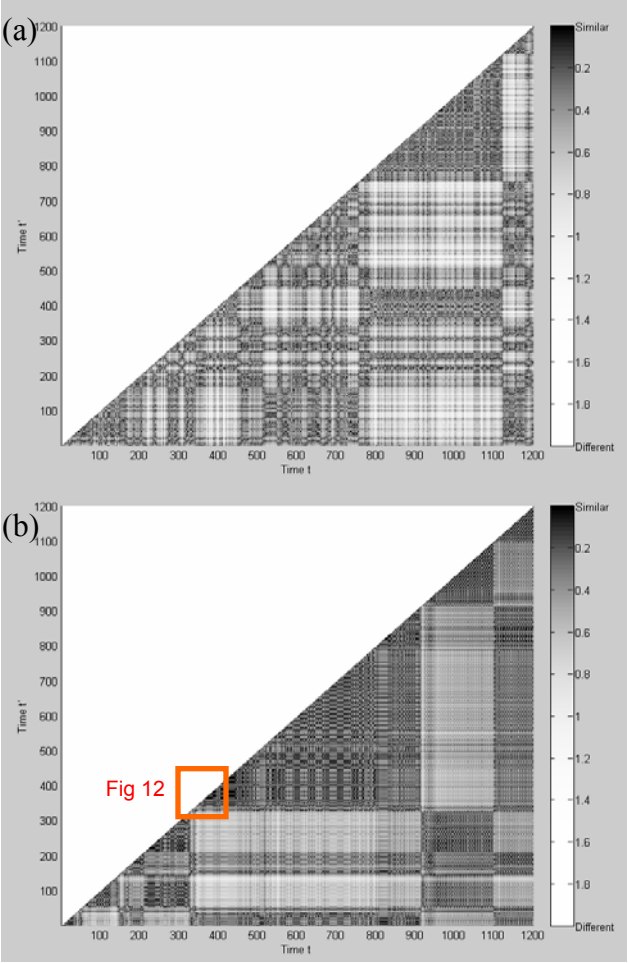

Figure 11. Point-cloud visualisations for the bee controlled by (a) MRL and (b) SART-MRL

Like the snail using MRL, Figure 11(a) shows little structured, cyclic behaviour emerging in the bee controlled by MRL. This is apparent from the light overall colour of the point-cloud diagram, which indicates fewer matching or similar affordances were executed. The reason for the reduction in emergent structured behaviour is the bee's colour sensor. The colour sensor returns particularly noisy readings depending on the distance of the robot to the coloured object being sensed and other factors such as ambient light.

The much darker triangles in Figure 11(b) indicate that some structured behaviour is occurring in the bee controlled by SART-MRL. Figure 12 is a close-up of two of these triangles, which show clear diagonal patterns. The first in R_1 represents a distributed cycle in which the bee repeatedly turns its head between the red panel and the neutral region between the panels. This cycle has L=6 and D=26 and repeats actions: $A_3\ A_1\ A_2\ A_1\ A_3\ A_2$ … This represents the bee experimenting with its colour sensor by turning its head then holding it still on one of the colour panels.

The second cycle in R_2 is a continuous cycle in which the bee is experimenting with its motor in the neutral space between the colour panels. In this space the red and green intensities are both low or zero. This cycle has length L=2, duration D=30 and repeats actions: $A_1\ A_2$. A period of exploration is evident between the two cycles.

These results are important as they show that the robot controlled by the SART-MRL algorithm is able to develop more structured behaviour than the MRL algorithm. This is due to its ability to cluster states as situations and generalise over the complex state space.

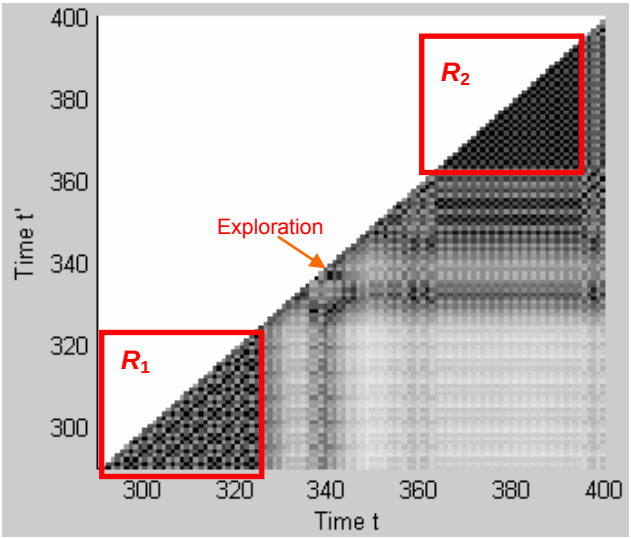

Figure 12. Close-up of Figure 11(b). Two structured behaviour cycles are evident as diagonal patterns in R_1 and R_2.

Figure 13 shows the point-cloud diagrams for the cricket. Figure 13(a) shows that once again, the cricket controlled by MRL has little structured behaviour. This is evident from the light overall colour of the diagram. In addition, there is no clear shift in attention focus over the course of the cricket's life. This is indicated by the absence of light coloured rectangular regions or darker triangular regions. In contrast, the point-cloud diagram for the cricket controlled by SART-MRL in Figure 13(b) does have the characteristics of shifting attention focus. In addition, a close-up view of the dark triangular regions, such as that in Figure 14, shows the characteristic diagonal patterns of structured behaviour cycles. Figure 14 shows a continuous cycle with L=3 and D=153, using actions: $A_1\ A_2\ A_3$ …This represents the cricket is experimenting with its motor settings.

Figure 15 shows the point-cloud diagrams for the ant. This is the noisiest of the applications as the

accelerometer readings result from the rocking motion of the ant as it moves. This is influenced by gravity and, to a lesser extent, the wires attaching the robot to the intelligent brick. Like the cricket controlled by MRL, Figure 15(a) shows that the ant controlled by MRL exhibits little or no structured behaviour cycles and has little change in attention focus. This is evidenced by the light, even colours on the point-cloud diagram.

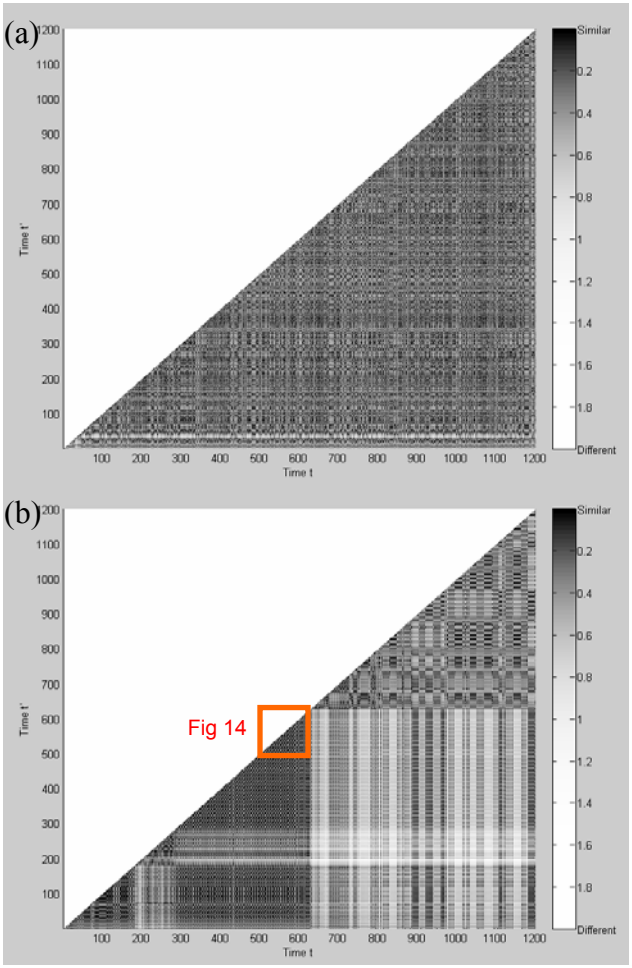

Figure 13. Point-cloud visualisations for the cricket controlled by (a) MRL and (b) SART-MRL

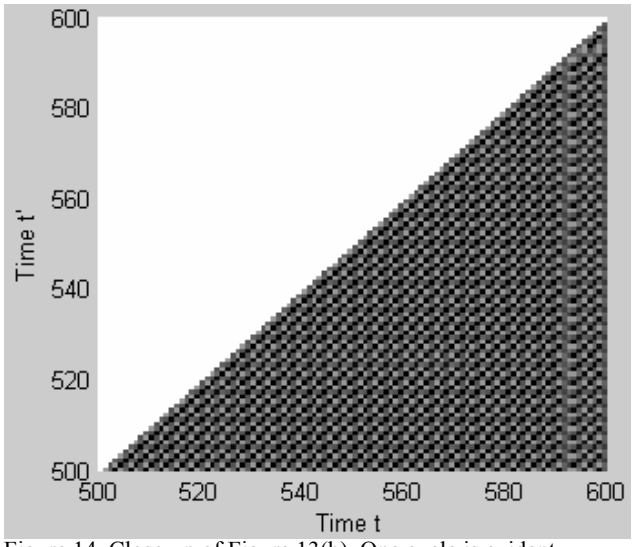

Figure 14. Close-up of Figure 13(b). One cycle is evident.

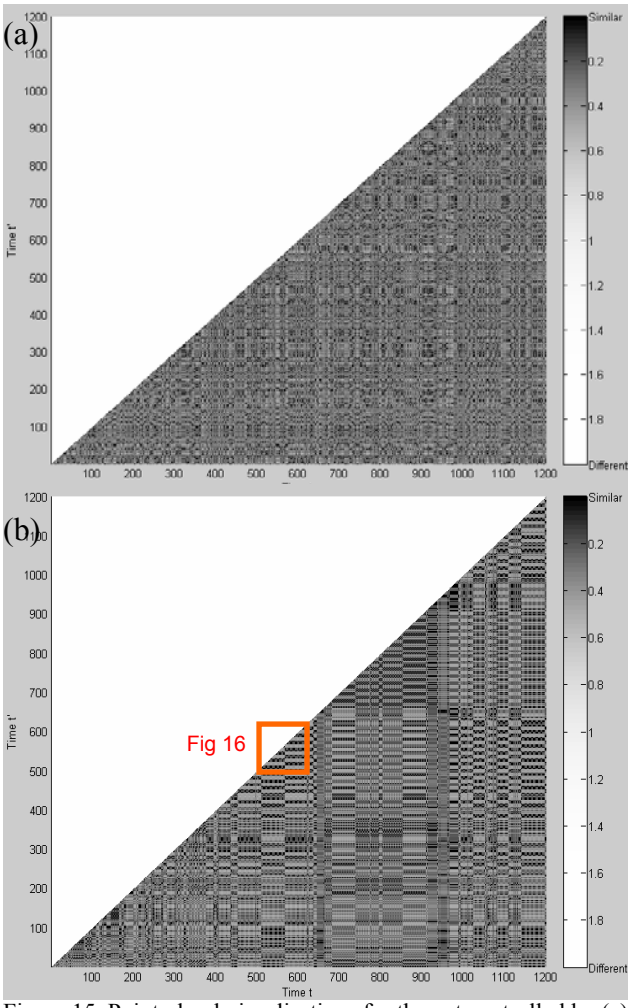

Figure 15. Point-cloud visualisations for the ant controlled by (a) MRL and (b) SART-MRL

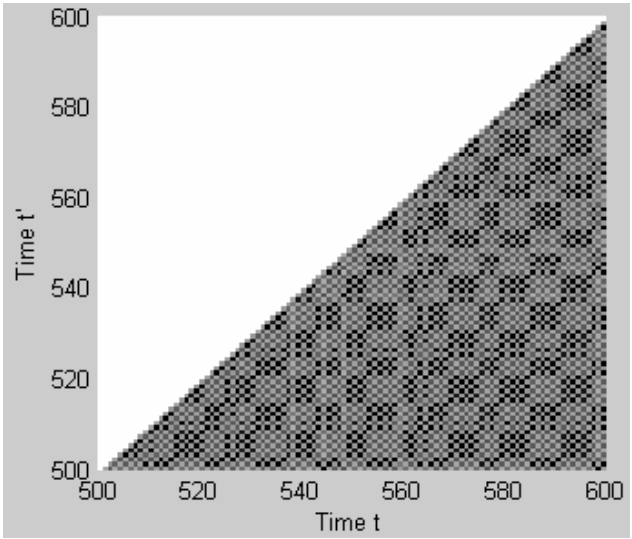

Figure 16. Close-up of Figure 15(b). A noisy, semi-structured behaviour cycle is evident from the diagonal pattern. This behaviour cycle was the robot 'walking' across the room.

The ant using SART-MRL was perhaps the most interesting of the insect-bots, as it was able to learn to 'walk' motivated only by the cycle-based intrinsic motivation function. The 'walk', shown in Figure 16, was somewhat jerky, with the ant learning to combine a

sequence of 'move-forward' and 'stop-motor' actions. However, this – and the other results from this section – demonstrates that robots can learn basic joint attention behaviours such as 'walking' and 'head movement' without being specifically programmed to learn those behaviours. In addition, even though each robot uses the same intrinsic motivation function, each robot learns different cycles in response to its physical structure.

5 Conclusion

This paper has introduced a model of affordances for controlling and evaluating intrinsically motivated robots. Our control model combines the model of affordances with motivated reinforcement learning and simplified adaptive resonance theory networks to produce intrinsically motivated robots that can generalise over noisy state spaces. The evaluation technique uses point-cloud diagrams to visualise cycles in the affordances executed by the robots.

We implemented and experimented with the control and evaluation models on four critter-bots using the *Lego Mindstorms NXT* platform. Results show that the evaluation technique can distinguish:

- Changing attention focus by a robot;
- Periods of exploration and exploitation;
- Cyclic behaviour.

Qualitatively, we can thus conclude that:

- The SART-MRL algorithm exhibited more structured behaviour than MRL;
- The SART-MRL algorithm exhibited greater ability to exploit learned behaviour the MRL control algorithm.

The main weakness of the point-cloud diagrams as an evaluation technique is the difficulty of interpreting the charts for long time periods. The next phase of this work will focus on developing an automated, numerical analysis of the point-cloud diagrams to permit a quantitative evaluation of behaviour. This will include automatically identifying properties such as the number, length and duration of behaviour cycles and the amount of time spent exploring and exploiting. This will further permit the design and analysis of more complex motivated robots for more complex environments.

References

[Ahlgren and Halberg, 1990] A. Ahlgren and F. Halberg. *Cycles of nature: an introduction to biological rhythms.* Washington DC: National teachers association, 1990.

[Baraldi and Alpaydin, 1998] A Baraldi and E. Alpaydin. Simplified ART: a new class of ART algorithms. *International Computer Science Institute,* Technical Report TR 98-004, Berkley, CA, 1998

[Dunlap et al., 2003] J. Dunlap, J. Loros and P. DeCoursey. *Chronobiology: biological timekeeping*. Sinauer Associates, 2003.

[Forbes and Fiume, 2005] K. Forbes and E. Fiume. An efficient search algorithm for motion data using weighted PCA. *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation.* pp 67-76, CA, 2005.

[Gibson, 1979] J. Gibson. *The ecological approach to visual perception.* Boston, MA, Houghton Mifflin, 1979.

[Hafner and Kaplan, 2008] V. Hafner and F. Kaplan. Interpersonal maps: how to map affordances for interaction behaviour. *Towards affordance-based robot control.* In *Lecture Notes in Computer Science,* J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.

[Hertzberg et al., 2008] J. Hertzbert, K. Lingemann, C. Lorken, A. Nuchter and S. Stiene. Does it help a robot navigate to call navigability an affordance? *Towards affordance-based robot control.* In *Lecture Notes in Computer Science,* J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.

[Huang and Weng, 2007] X. Huang and J. Weng. Inherent value systems for autonomous mental development. *International Journal of Humanoid Robotics,* 4(2):407-433, 2007.

[Kolb et al., 1984] D. A. Kolb, I.M. Rubin, and J.M. McIntyre, (Eds) *Organizational Psychology: Readings on Human Behaviour in Organizations*. Prentice-Hall, Englewood Cliffs, NJ, 1984.

[Kovar and Gleicher, 2004] L. Kovar and M. Gleicher. Automated extraction and parametrization of motions in large data sets. *Proceedings of ACM SIGRAPH 2004 Papers*, pp 559-568, Los Angeles, CA, August, 2004.

[Li and Holstein, 2002] B. Li and H. Holstein. Recognition of human periodic motion – a frequency domain approach. *Proceedings of the 16th International Conference on Pattern Recognition*, pp 311-314, Washington DC.

[Merrick, 2009] Merrick, K., Evaluating Intrinsically Motivated Robots using Affordances and Point-Cloud Matrices, *The Ninth International Conference on Epigenetic Robotics (EpiRob09)*, Venice, Italy (to appear).

[Merrick and Huntington, 2008] K. Merrick and E. Huntington. Attention focus in Curious, Reconfigurable Robots. *Proceedings of the 2008 Australian Conference on Robotics and Automation.* ANU, Canberra, Australia, (CD: no page numbers), December 2008.

[Modayil and Kuipers, 2008] J. Modayil and B. Kuipers. The initial development of object knowledge by a learning robot. *Robotics and Autonomous Systems.* 56:879-890, Elsevier, 2008.

[Nefadov, 2004] S.A. Nefedov. A model of demographic cycles in traditional societies: the case of ancient China. *Evolution and History*, 3(1): 69-80, 2004.

[Oudeyer et al., 2007] P.-Y. Oudeyer, F. Kaplan and V. Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation.* 11(2):265-286, April 2007.

[Paletta and Fritz, 2008] L. Paletta and G. Fritz. Reinforcement Learning of Predictive Features in Affordance Perception. *Towards affordance-based robot control.* In *Lecture Notes in Computer Science,* J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.

[Rome et al., 2008a] E. Rome, J. Hertzberg and G. Dorffner. *Towards affordance-based robot control.* In *Lecture Notes in Computer Science,* J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.

[Rome et al., 2008b] E. Rome, L. Paletta, E. Sahin, G. Dorffner, J. Herzberg, R. Breithaupt, G. Fritz, J. Irran, F. Kintzler, C. Lorken, S. May and E. Ugur. The MACS project: an approach to affordance-inspired robot control. *Towards affordance-based robot control.* In *Lecture Notes in Computer Science,* J. G. Carbonell and J. Siekmann (Eds), Springer-Verlag, Berlin-Heidelberg, 2008.

[Stoytchev, 2005] A. Stoytchev. Behaviour-grounded representation of tool affordances. *Proceedings of the IEEE International Conference on Robotics and Automation,* pp 3060-3065, Barcelona, Spain, 2005.

[Tang et al., 2008] K.-T. Tang, H. Leung, T. Komura, H. Shum. Finding repetitive patterns in 3D human motion captured data. *Proceedings of the Second International Conference on Ubiquitous Information Management and Communication.* pp 396-403, Suwon, Korea, 2008.

[Watkins and Dayan, 1992] C. Watkins and P. Dayan. Q-Learning, *Machine Learning,* 8(3):279-292, 1992.