Cañamero, L., Oudeyer, P.-Y., and Balkenius, C. (Eds) Proceedings of the Ninth International Conference on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems. Lund University Cognitive Studies, 146.

Evaluating Intrinsically Motivated Robots using Affordances and Point-Cloud Matrices

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

Abstract

A key challenge developing intrinsically motivated robots is evaluation of the robots' emergent behaviour. Evaluation techniques for intrinsically motivated robots must be open-ended enough to identify any emergent behaviours, but specific enough to quantify those behaviours in a meaningful way. This paper describes a novel use of point-cloud matrices for detecting cycles of affordances in robots' behaviour. The technique is demonstrated by evaluating two motivated reinforcement learning algorithms on four *Lego Mindstorms NXT* critter-bots. Results show that the evaluation technique can identify changing attention focus, periods of exploration and exploitation and repetitive, cyclic behaviour.

1. Introduction

Intrinsically motivated robots are characterised by their ability to select their own goals. They use an embedded computational model 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. The capacity for autonomous, open-ended goal-selection gives intrinsically motivated robots the potential to adapt to unexpected changes in their environment. In addition, they can develop novel or creative behaviours that were not explicitly programmed by engineers.

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

This paper adapts a technique used to evaluate repetitive patterns in human motion for use with robots. Point-cloud matrices are used to visualise cycles of affordances acted on by a robot. Section 2 begins with a brief survey of techniques for evaluating the behaviour of intrinsically motivated robots and natural systems. Section 3 describes how point-cloud matrices and affordances can be used to evaluate robots' behaviour. Section 4 demonstrates the technique by evaluating two motivated reinforcement learning algorithms on four critter-bots using the *Lego Mindstorms NXT* platform. Results show that the evaluation techniques can identify changing attention focus, periods of exploration and exploitation and repetitive, cyclic behaviour learned by a robot.

2. Evaluating Intrinsically Motivated Robots

Various techniques have been used to evaluate intrinsically motivated robots. For example, Oudeyer et al. (2007), 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.

Other approaches to the evaluation of intrinsically motivated robots include algorithm specific approaches (Huang and Weng, 2007), case studies and bifurcation diagrams (Merrick and Huntingon, 2008). In contrast, this paper presents a general approach in which the performance of a robot is characterised in terms of its ability to act in structured, cyclic patterns. This allows evaluation of the emergent behaviour of a robot, independent of a specific task or controlling algorithm.

The importance of cyclic behaviour in natural systems such as animals has been identified by biologists (Ahlgren and Halberg, 1990; Dunlap et al., 2003).

Common examples include the circadian rhythm, migratory cycles, and cycles associated with seasons or tides. A number of techniques for identifying repetitive cyclic patterns in human motion have been proposed (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) to identify patterns in robot motion.

2.1 Point-Cloud Matrices

Tang et al. (2008) use point-cloud matrices to visualise posture similarity in motion-capture data, as shown in Figure 1. 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 at adjacent time periods. Distributed cycles, represented by cross patterns, repeat a sequence of postures at intermittent time periods.

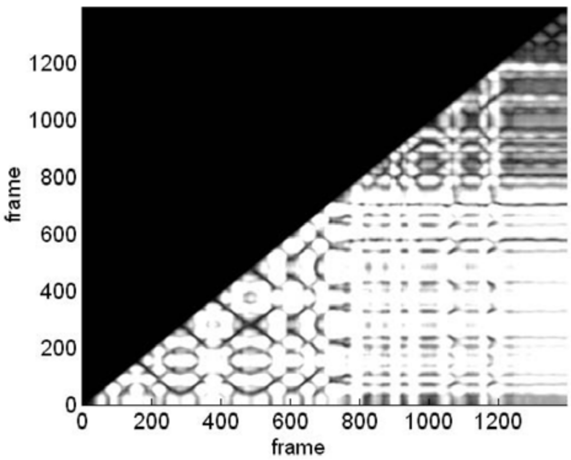

Figure 1. Tang et al. (2008) visualise human motion capture data using a point-cloud matrix showing posture similarity. Diagonal patterns represent cyclic behaviour.

Tang et al. (2008) use point-cloud matrices to visualise posture similarity for human dancers. Because we are often interested not only in the posture of a robot, but also the actions it performs, this paper proposes an alternative technique where affordance similarity rather than posture similarity is visualised.

2.2 Affordances

The concept of affordances is generally attributed to Gibson (1979) as an approach to understanding visual perception in natural systems. His 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. Thus affordances capture both the state of an environment and the actions available to an organism in that state.

While there is no universal 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; Rome et al., 2008b). These include 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). This paper extends existing work with affordances in robotics to the challenge of evaluating the behaviour of intrinsically motivated robots.

3. Cyclic Evaluation of Robot Behaviour using Affordances and Point-Cloud Matrices

Affordances can be thought of as mappings or relationships between some aspect of a 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. The total environment of a robot is considered to comprise data describing both its internal and external state. For example, a *Lego* robot such as the one shown in Figure 3(b) may be described by the internal state of its motor (on/off, power level etc.) and by the state of its external environment detected by its colour sensor (red level, green level, blue level etc.). More formally, a robot's state $S_{(t)}$ at time t is described by its internal state $S_{I(t)}$ and its external state $S_{E(t)}$:

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

An attribute-based representation $S = (s_1, s_2, s_3, \ldots)$ is required for application of the technique in this paper. The set $\mathbf{A}$ of actions afforded by a state S at time t is:

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

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

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

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

where A is a numeric action identifier.

The point-cloud visualization is constructed by computing the Euclidean distance $\mathrm{dist}(X_{(t)}, X_{(t')})$ 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 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 2. Their meanings are discussed in the following sections.

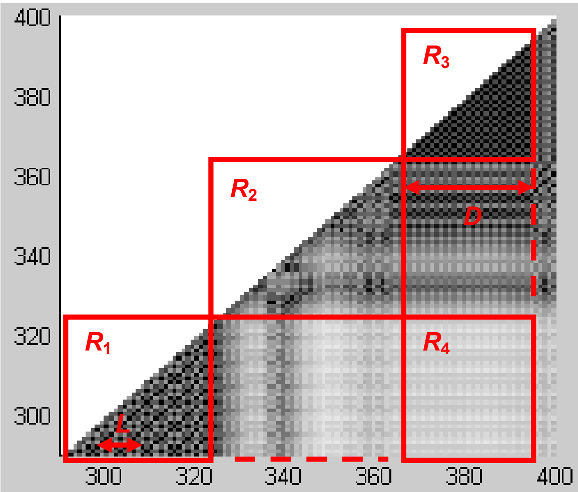

Figure 2. Characteristic patterns on point-cloud diagrams of robotic motion. See text for detailed description.

3.1 Cyclic Behaviour in Robots

In Figure 2, the dark triangles of parallel diagonals in the highlighted regions R_1 and R_3 are cycles of affordances. 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. R_1 shows a distributed cycle while R_3 shows a continuous cycle.

Continuous Cycles

Continuous cycles repeat the same sequence of affordances at adjacent time periods. For example:

$\boxed{X_1, X_2, X_3}, \boxed{X_1, X_2, X_3}, \boxed{X_1, X_2, X_3}, \boxed{X_1, X_2, X_3} \ldots$

Continuous cycles appear as parallel diagonals on a point-cloud matrix, such as that shown in R_3.

Distributed Cycles

Distributed cycles repeat two or more sequence of affordances at intermittent time periods. The following example interleaves two sequences:

$\boxed{X_1, X_2, X_3}, \boxed{X_3, X_2, X_1}, \boxed{X_1, X_2, X_3}, \boxed{X_3, X_2, X_1} \ldots$

Distributed cycles appear as cross patterns on a point-cloud matrix, such as that shown in R_1.

3.2 Exploration versus Exploitation

Intrinsically motivated systems, both natural and artificial, must exhibit periods of both explorative and exploitative behaviour. Exploration is required to find new, motivating things to learn about. Exploitation is required to carry out learned behaviours. In Figure 2, the structured patterns in regions R_1 and R_3 indicate that the robot is exploiting a learned cycle. In contrast, the unstructured, random pattern in R_2 indicates that the robot is exploring to find something new to learn about.

3.3 Attention Focus

As a robot explores, its focus of attention shifts. The robot may focus on exploiting entirely new behaviours or return its focus to a previous behaviour. The colour of the rectangular regions linking dark triangles can be used to identify these different types of shifts in attention focus. For example, the light rectangular region R_4 in Figure 2 indicates that the affordances in R_1 are generally dissimilar to those in R_3. If R_4 were dark in colour it would indicate that the affordances in R_1 and R_3 were similar.

4. Demonstration

This section demonstrates the evaluation technique by evaluating two motivated reinforcement learning (MRL) algorithms on four *Lego Mindstorms NXT* critter-bots, shown in Figure 3.

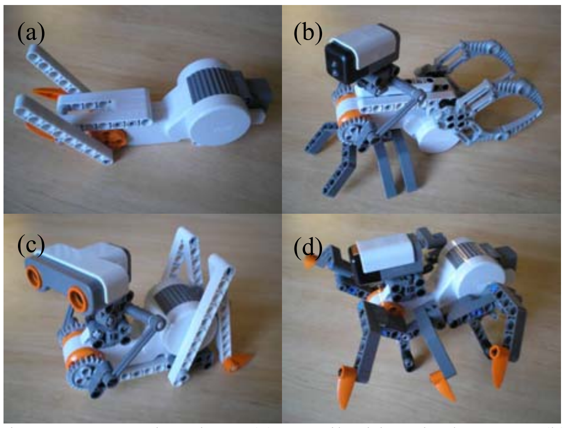

Figure 3. Four critter-bots: (a) a snail with a single motor; (b) a bee with a motor and colour sensor; (c) a cricket with a motor and ultrasonic sensor; (d) an ant with a motor and accelerometer.

The first algorithm, called MRL, is a table-based approach (Merrick and Maher, 2009). The second algorithm, called SART-MRL uses a function approximation technique based on simplified adaptive resonance theory (SART) networks (Baraldi and Alpaydin, 1998) to generalise over the robot's state space. It is hypothesised that the latter approach will be able to learn more effectively and exhibit more structured behaviour on the robots, because of its ability to generalise over the noisy state space of the robots. The evaluation technique should thus reflect this hypothesis by revealing the presence of structured behaviour cycles by the robots using SART-MRL.

The two algorithms – MRL and SART-MRL – were each run for 1,200 time-steps (approximately six minutes) on each critter-bot. The following paragraphs describe the state and action spaces of each robot and some of the emergent behaviours with reference to point-cloud diagrams for each run.

4.1 The Snail

The first critter-bot, shown in Figure 3(a), 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^{o} and 360^{o} from the initial position of the motor. The movement reading is enumerated such that 0 means the motor is stopped, 200 means the motor is moving forwards and 100 means it is moving backwards.

Every state encountered by the snail 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. The control algorithms respond to an intrinsic motivation function to learn which actions to select in each state.

Figure 4 visualises the behaviour of the snails using each algorithm. Figure 4(a) shows the MRL algorithm and Figure 4(b) the SART-MRL algorithm. The white rectangular regions in Figure 4(a) indicate that this robot is focusing attention on different affordances at different times. Inspection of the log file for this critter-bot 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, zooming in on the darker triangular region (Figure 5) 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, zooming in on some of the dark triangles for the snail using SART-MRL shows a number of different diagonal patterns. Figure 6 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 are visible before R_1, between the exploitative behaviour in R_1 and R_2 and after R_2.

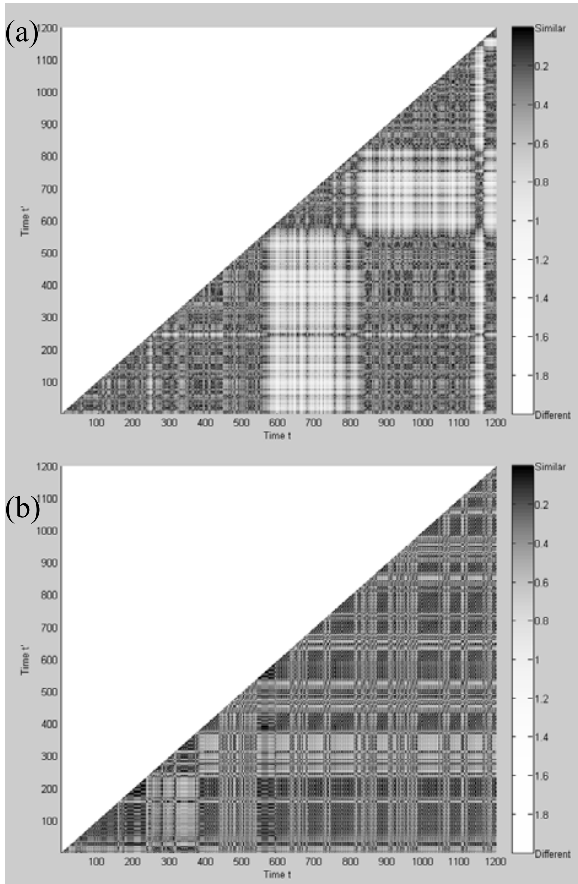

Figure 4. Point cloud visualisations for the snails using (a) MRL and (b) SART-MRL

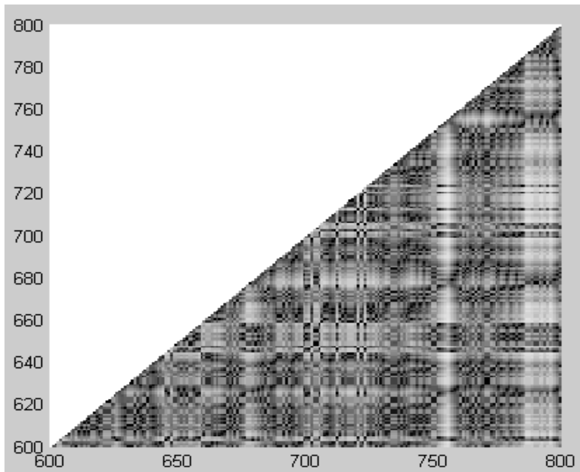

Figure 5. Zoomed region of Figure 4(a). No structured behaviour cycles (dark diagonals) are evident.

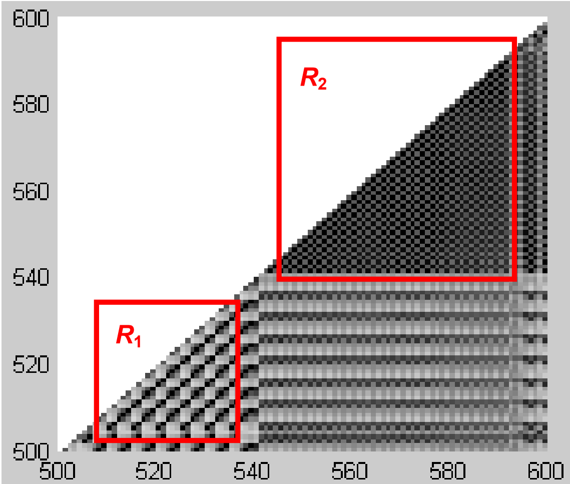

Figure 6. Zoomed region of Figure 4(b). Two structured behaviour cycles are evident as diagonal patterns in regions R_1 and R_2. Explorative behaviour is evident before, between and after these regions.

One of the weaknesses of using point-cloud diagrams to analyse robots is that only short time periods can reasonably be displayed on a screen or page. This is addressed in this paper by zooming in on regions of interest. One direction for future work is automated analysis of the point-cloud diagrams to identify such regions of interest and to generate statistical data about the long-term characteristic behaviour of the robot. This is discussed further in Section 5.

4.2 The Bee

The second critter-bot, shown in Figure 3(b), is a bee with a motor and colour sensor. The motor allows the bee to turn its colour sensor 'head' through 45^{o} to both the left and right. The bee can sense the rotation of the motor and whether the motor is moving or not. The colour sensor provides data describing red, blue and green intensities of the critter'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 for the snail, every state encountered by the bee 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.

Figure 7 shows the point-cloud diagrams for the bees. As with the snail using MRL, Figure 7(a) again shows the characteristic patterns of shifting attention focus. Also like the snail, however, this plot shows little structured, cyclic behaviour emerging in the bee using 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 7(b) indicate that some structured behaviour is occurring in the bee running SART-MRL. Figure 8 zooms in on 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 as it alternates between high and low red intensity readings.

Between R_1 and R_2 is a period of exploration as the robot seeks different motivating stimuli. The mid-range grey colour of the linking square region indicates that some of the affordances in R_2 are similar to those in R_1.

The 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.

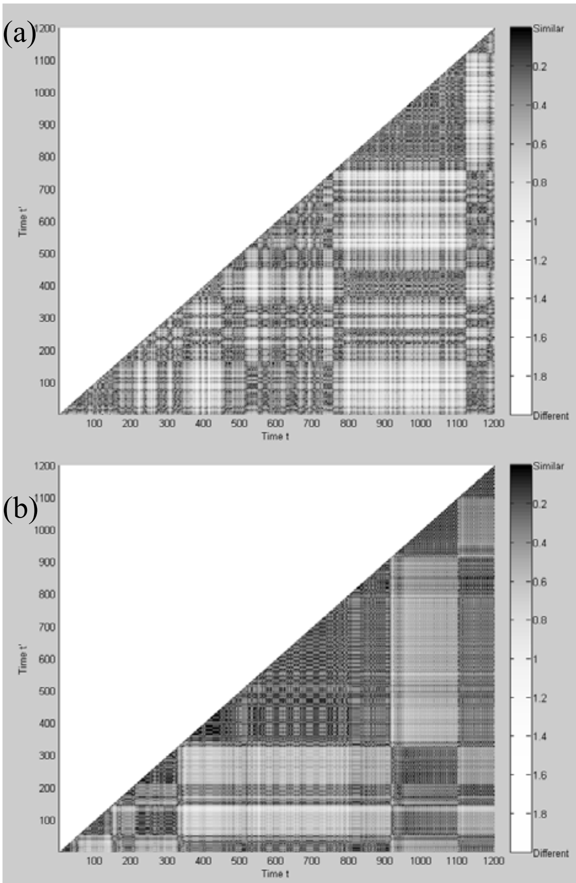

Figure 7. Point cloud visualisations for the bees using (a) MRL and (b) SART-MRL

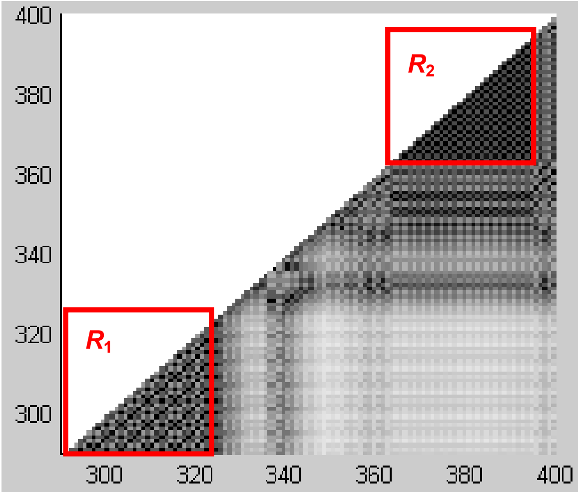

Figure 8. Zoomed region of Figure 7(b). Two periods of exploitation are evident, separated by exploration.

4.3 The Cricket

Figure 3(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° to both the left and right. The cricket can sense the rotation of the motor and 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.

Once again, every state encountered by the cricket 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.

Figure 9 shows the point-cloud diagrams for the two crickets. Once again, the cricket using MRL (Figure 9(a)) shows 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 to Figure 9(a), the point-cloud diagram for the cricket using SART-MRL (Figure 9(b)) does have the characteristics of shifting attention focus, with a number of light-coloured square regions evident. In addition, zooming in on dark triangular regions, such as that in Figure 10, shows the characteristic diagonal patterns of structured behaviour cycles. Figure 10 shows a continuous cycle with L=3 and D=153, using actions: A_1 A_2 A_3 …This represents the cricket experimenting with its motor settings. Around t=590 the robot begins to explore once more.

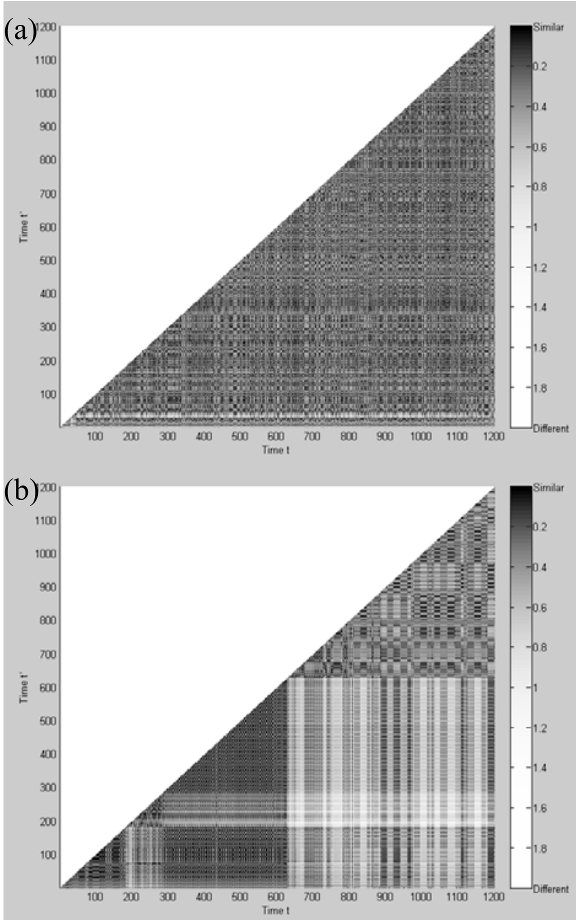

Figure 9. Point cloud visualisations for the crickets using (a) MRL and (b) SART-MRL

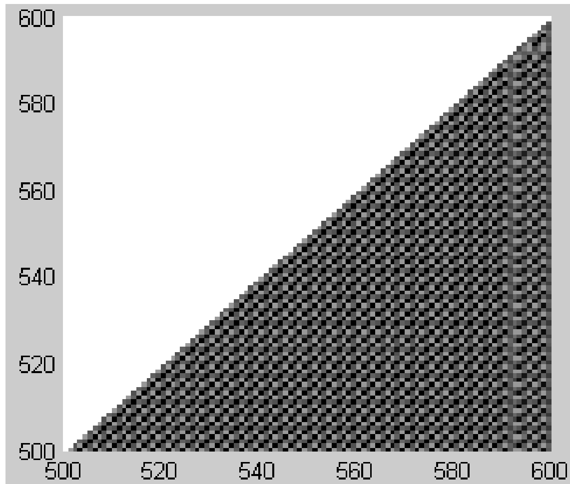

Figure 10. Zoomed region of Figure 9(b). One cycle is evident, followed by exploration from t=590.

4.4 The Ant

Finally, the fourth critter-bot in Figure 3(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. Every state encountered by the ant 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.

Figure 11 shows the point-cloud diagrams for the ants using MRL and SART-MRL. This is the noisiest of the applications as accelerometer readings are affected by 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 using MRL, Figure 11(a) shows that the ant using 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.

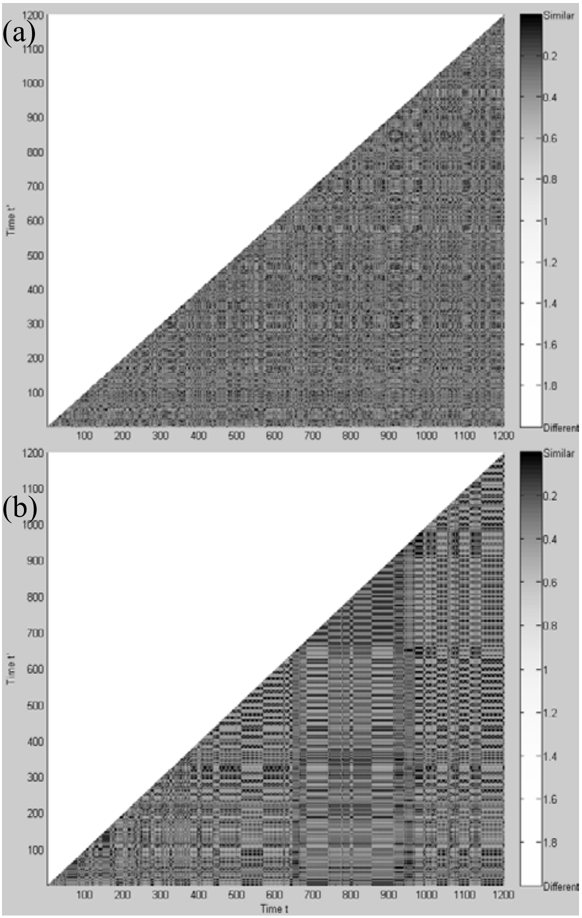

Figure 11. Point cloud visualisations for the ants using (a) MRL and (b) SART-MRL

Figure 11(b) for the ant using SART-MRL also shows relatively mid-range greys, although more triangle patterns are evident in the diagram. Zooming in on parts of the plot, such as in Figure 12, shows that structured behaviour is evident, but the characteristic diagonal patterns are much noisier. This mirrors the fact that the state space for this robot is also much noisier. Figure 12 in fact shows a 'walking' behaviour learned by the ant. The walk was somewhat jerky, with the ant learning to combine a sequence of 'move-forward' and 'stop-motor' actions. Despite this, the structured behaviour was evident both visually when the robot was learning and in the point-cloud diagram. One of the strengths of the point-cloud visualisations is that they can reveal quite noisy, yet still structured, behaviour that would be difficult to identify by analysing the data numerically.

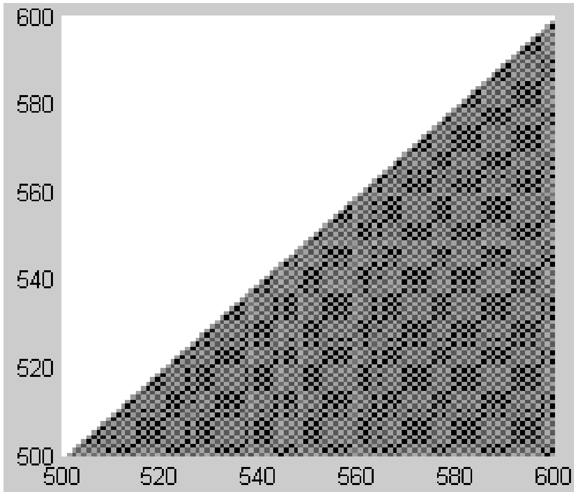

Figure 12. Zoomed region of Figure 11(b). A noisy yet still structured behaviour cycle is evident. This behaviour cycle was the robot 'walking'.

5. Conclusion and Future Work

This paper has presented a novel use of point-cloud matrices and affordances for evaluating intrinsically motivated robots. A demonstration was presented of the evaluation model on two motivated reinforcement learning approaches on four critter-bots using the *Lego Mindstorms NXT* platform. Results show that the evaluation technique can distinguish:

- Changing attention focus by a robot – visible as light coloured, rectangular linking regions;
- Periods of exploration – visible as random patterns;
- Periods of exploitative cyclic behaviour – visible as dark, triangular patterns of diagonals.

In addition the length and duration of cycles can be computed from the diagrams. These results qualitatively confirmed the hypothesis that the SART-MRL control algorithm would exhibit more structured behaviour and greater ability to focus attention than the MRL control algorithm.

While the model in this paper does not seek to evaluate the 'intelligence', 'usefulness' or 'correctness' of a robot's behaviour, it provides an approach that can be used in conjunction with domain specific case studies or other metrics to identify the emergence of structured, cyclic patterns characteristic of learning.

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 the behaviour of a robot. This will complement the visualisations to assist with identifying regions of interest and provide a way to compare the behaviour of different robots numerically. The numerical analysis might include automatically identifying properties such as the number, length and duration of behaviour cycles. This work will further permit the design and analysis of more complex motivated robots running for longer time periods in complex environments.

References

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

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

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

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.

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

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.

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.

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

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.

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.

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.

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.

K. Merrick, M. L. Maher, *Motivated Reinforcement Learning Agents: Curious Characters for Multiuser Games*, Springer-Verlag, Berlin/Heidelberg, ISBN 978-3-540-89186-4, 2009

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

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.

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.

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, 2008a.

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, 2008b.

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

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.