

Predictive Act-R (PACT-R)

Using A Physics Engine and Simulation for Physical Prediction in a Cognitive Architecture

David Pentecost¹, Charlotte Sennersten², Robert Ollington¹, Craig A. Lindley², Byeong Kang¹

Information and Communications Technology¹

Robotic Systems and 3D Systems²

University of Tasmania¹

CSIRO Data 61²

Hobart, Tasmania

Australia

email: davidp12@utas.edu.au

email: charlotte.sennersten@csiro.au

email: robert.ollington@utas.edu.au

email: craig.lindley@csiro.au

email: byeong.kang@utas.edu.au



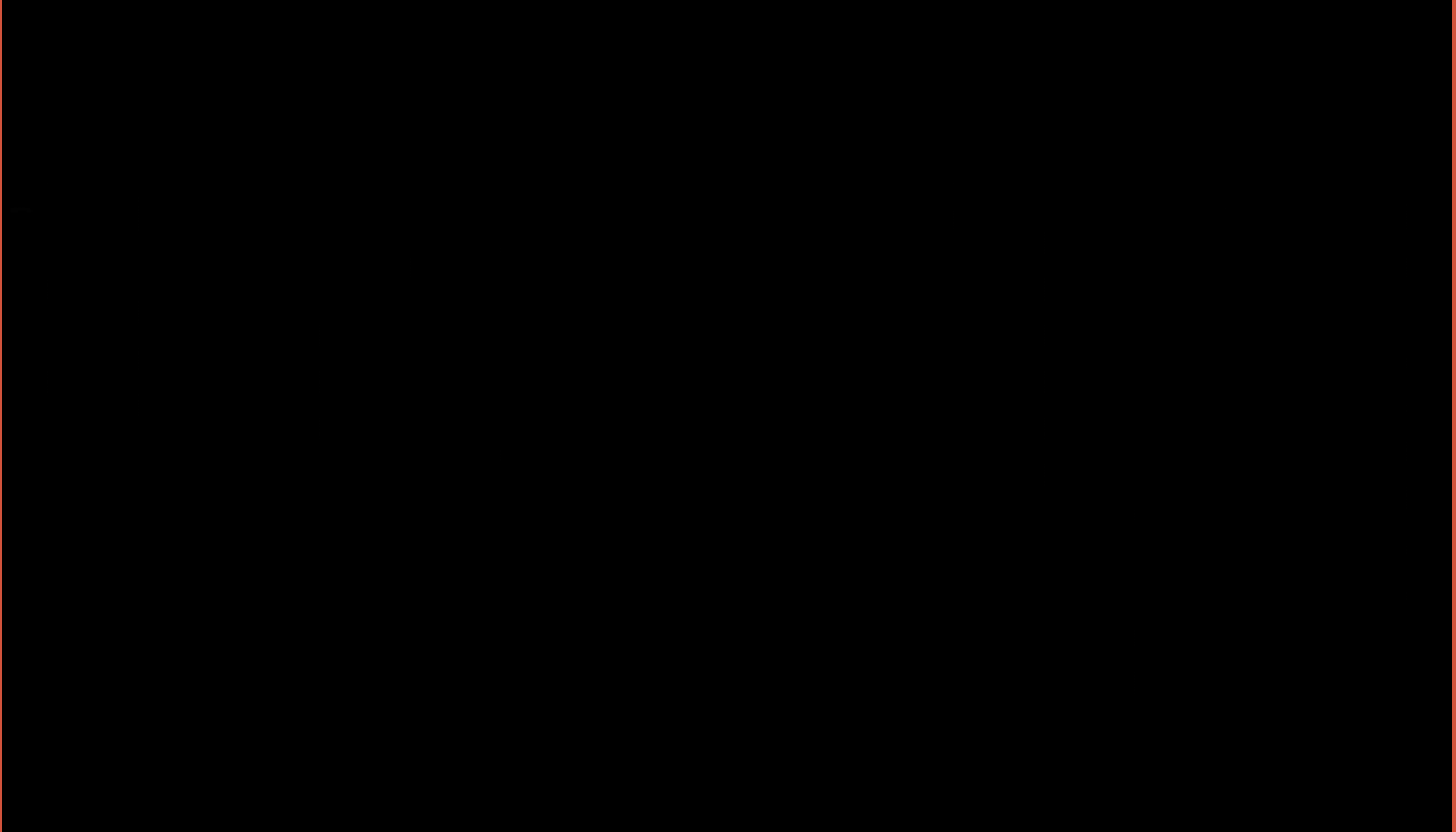
UNIVERSITY *of*
TASMANIA



1. INTRODUCTION
2. COGNITIVE AND NON-COGNITIVE ARCHITECTURES
3. METHODOLOGY
 - A. *Research Design*
 - B. *Implementation*
 - C. *Using Simulation and Prediction within a Cognitive Architecture*
 - D. *PACT-R Module Implementation in ACT-R*
4. AI MODELLING AND PREDICTION
 - A. *Prediction Models*
 - B. *Evaluation and Analysis*
5. RESULTS AND DISCUSSION
 - A. *Basic Random Shot Selection Model*
 - B. *Heuristic Selection Model*
 - C. *Predictive Selection Model*
 - D. *Performance*
6. CONCLUSION
7. FUTURE WORK

DAVID PENTECOST

Short intro



1. INTRODUCTION

The Research Question:

How can simulation and prediction improve decision quality in a cognitive architecture?

Predictive ACT-R (PACT-R)
Using A Physics Engine and Simulation for Physical Prediction in a Cognitive Architecture

David Passmore¹, Charles Spence², Robert Ollinger³, Craig A. Lindley⁴, Byoung Kang¹
¹Information and Communications Technology
Robotics Systems and IIC Systems
University of Tasmania
CRCR Code 01
Hobart, Tasmania
Australia
Email: david.p@utas.edu.au
charles.spence@utas.edu.au
robert.ollinger@utas.edu.au
craig.lindley@utas.edu.au
byoung.kang@utas.edu.au

Abstract
Advanced Cognitive Technologies can use predictive models to help us make better decisions in a complex world. In this paper, we describe a new predictive model for a cognitive architecture, PACT-R, which is designed to help us make better decisions in a complex world. PACT-R is a predictive model that uses a physics engine to simulate the world and predict the outcome of actions. This model is used to help us make better decisions in a complex world. PACT-R is a predictive model that uses a physics engine to simulate the world and predict the outcome of actions. This model is used to help us make better decisions in a complex world.

Keywords: Cognitive Architecture, ACT-R, AI Decision, Introduction
What do you do if you are asked to catch a ball that has been thrown in the air? You make a quick estimate of its trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

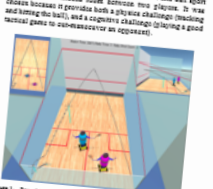


Figure 1: A diagram showing a ball in motion on a court, with a player positioned to catch it. The diagram is used to illustrate the concept of a physics engine in a cognitive architecture.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

Abstract
The human brain is a complex system that is able to make decisions in a complex world. This is a complex task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not. This is a simple task, but it is a complex one. The brain has to make a quick estimate of the ball's trajectory and decide whether to move to catch it or not.

1. INTRODUCTION

The Research Question:

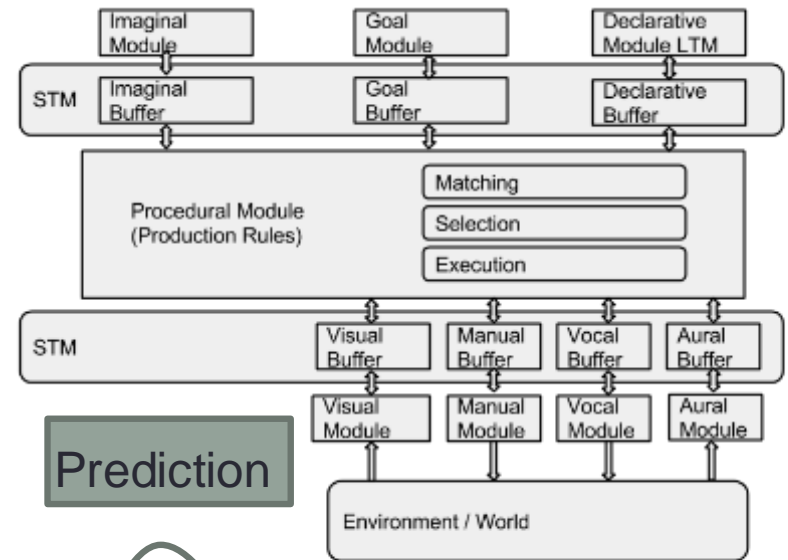
How can simulation and prediction improve decision quality in a cognitive architecture?



1. INTRODUCTION

The Research Question:

How can simulation and prediction improve decision quality in a cognitive architecture?

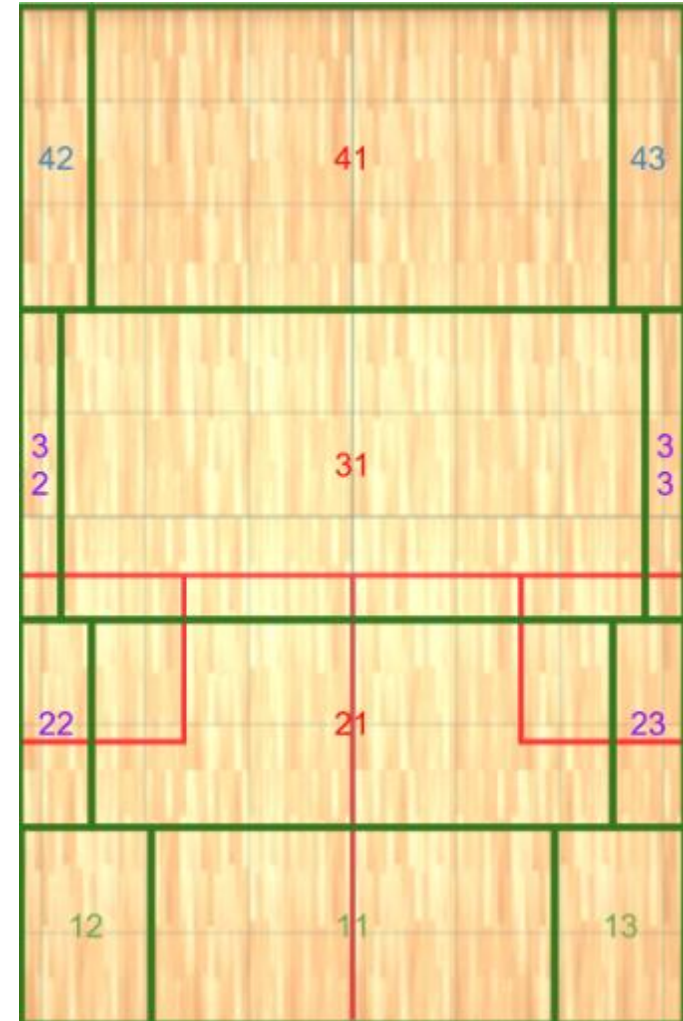


Zones For Shot Selection

Shot selection based on a limited choice per zone

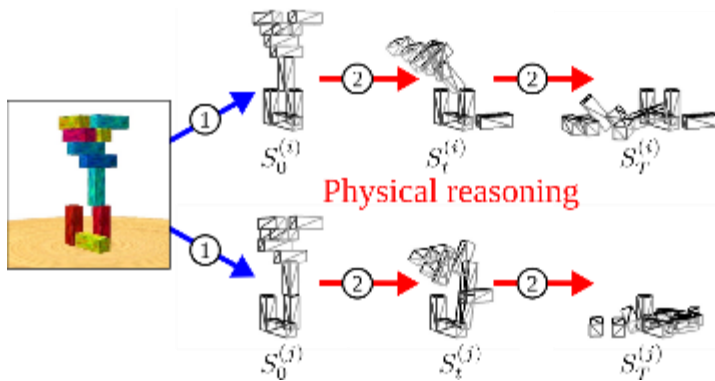
Based on Squash training patterns

Direction	Type	Description
Straight	Drop	Risky attacking shot, only played from a strong position
Cross Court	Drop	Very high risk attacking shot
Straight	Lob	Defensive high shot to back of court, safety shot gives time to recover
Cross Court	Lob	Defensive shot, safer than straight lob as there is less likelihood of the ball going out on the side wall
Boast	Lob (high)	High defensive boast. A shot played into the side wall that gives time to recover but often leads to a weak position for a player.
Boast	Short (low)	Attacking boast that stays low. Usually played from a strong position as a changeup shot.
Straight	Short	Attacking shot played low and hard that stays short and near the side wall.



1. INTRODUCTION

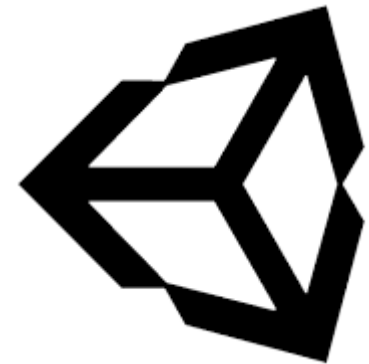
An aspect of human cognition that is not captured in most cognitive architectures is *simulation*. Imagination, and the use of imagined visualisations, constitutes a conscious result of simulation within human cognition. An example of the use of simulation in an artificial cognitive system is the Intuitive Physics Engine (IPE), which uses simulation to understand scenes [7].



Simulation as an engine of physical scene understanding

[Peter W. Battaglia](#),¹ [Jessica B. Hamrick](#),
and [Joshua B. Tenenbaum](#)

<http://web.mit.edu/~pbatt/www/>



Simulation For Scene Understanding

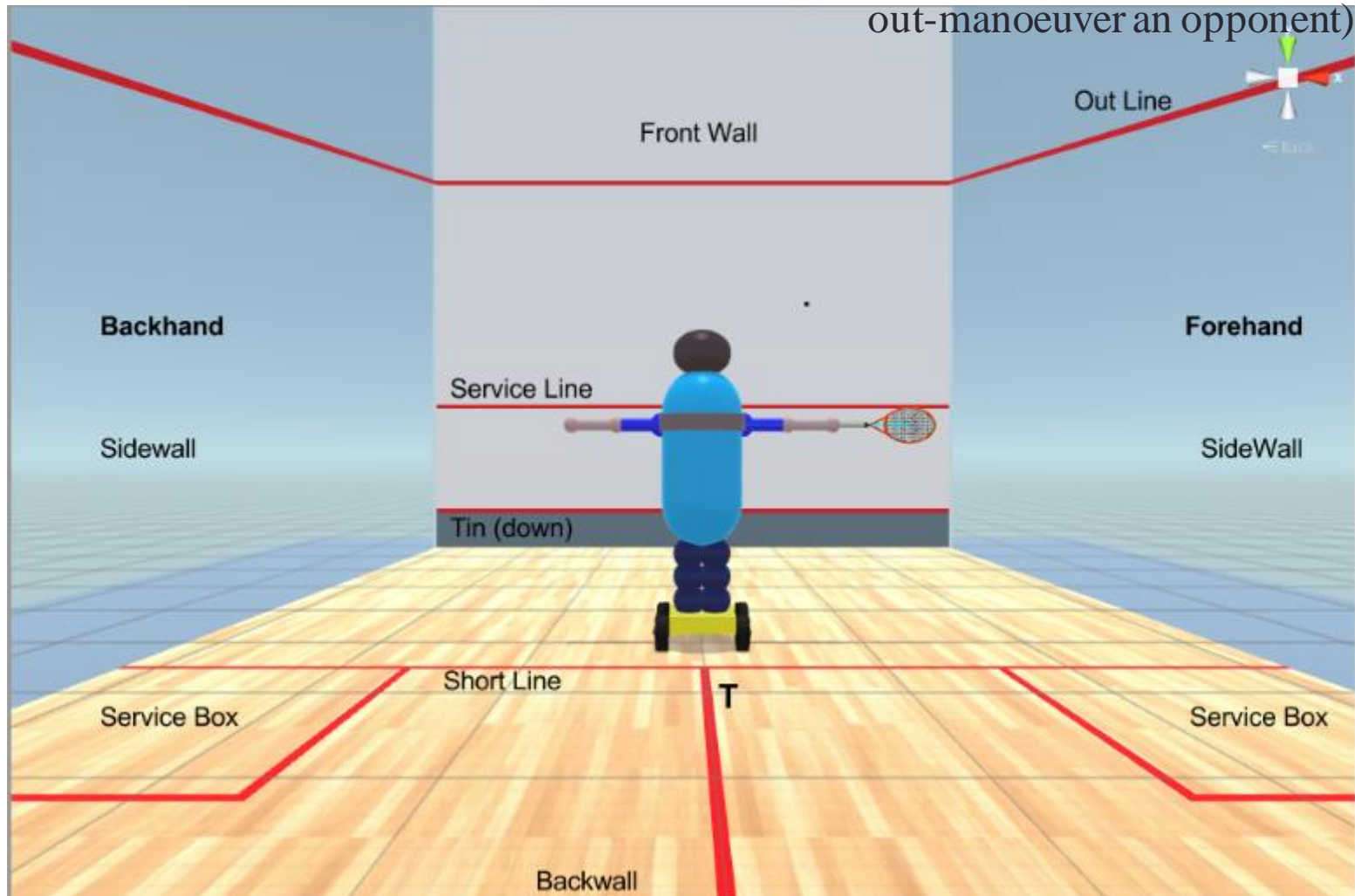
- Imperfect Perception
- Non Deterministic Dynamic Worlds
- (In mathematics and physics, a *deterministic* system is a system in which no randomness is involved in the development of future states of the system).

- Using a 3D Engine
- Low Fidelity Simulation
- How good does it need to be to be useful?

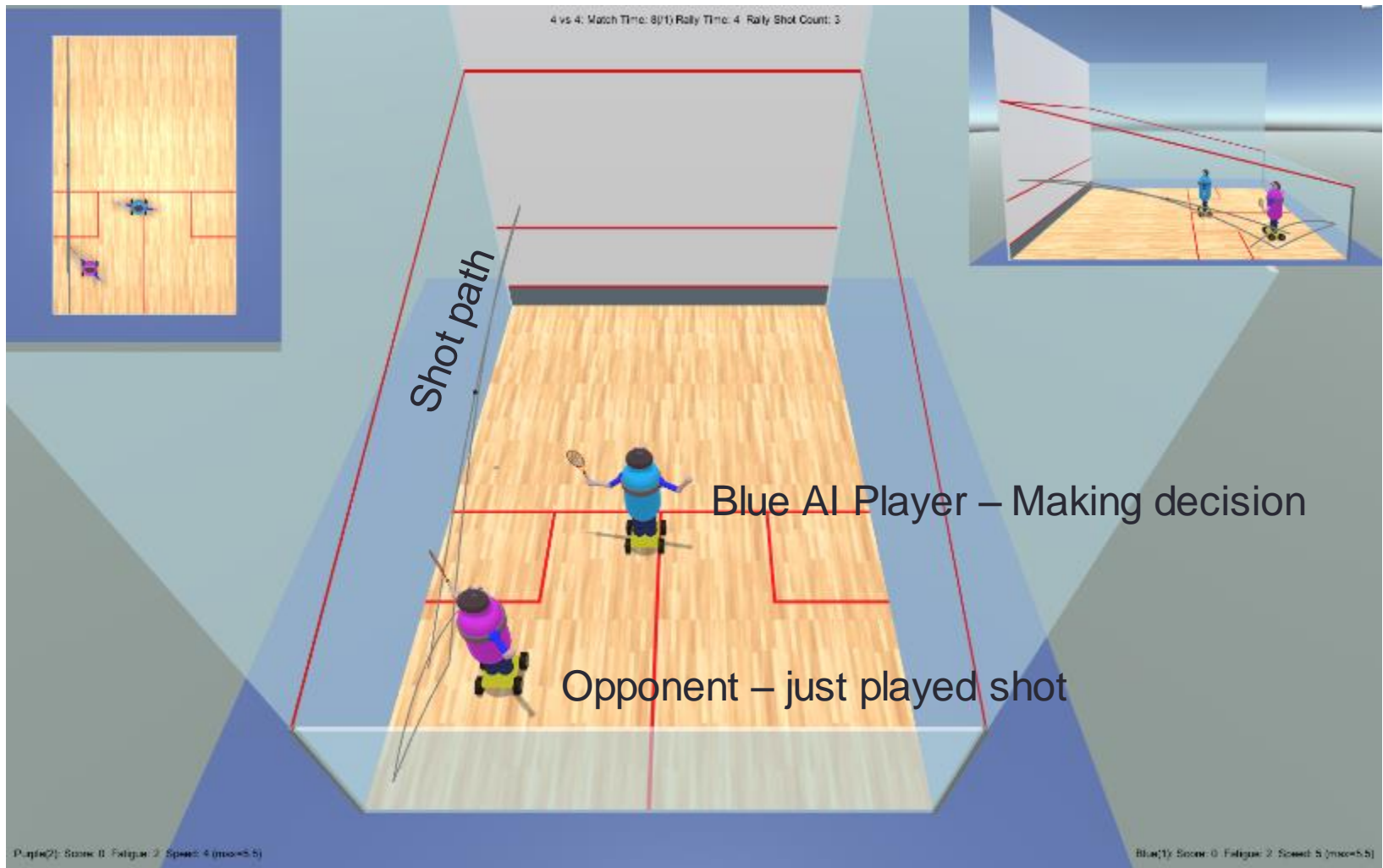
- Giving the AI Something to Reason With

Squash Terminology

It provides both a physics challenge (tracking and hitting the ball), and a cognitive challenge (playing a good tactical game to out-maneuver an opponent).

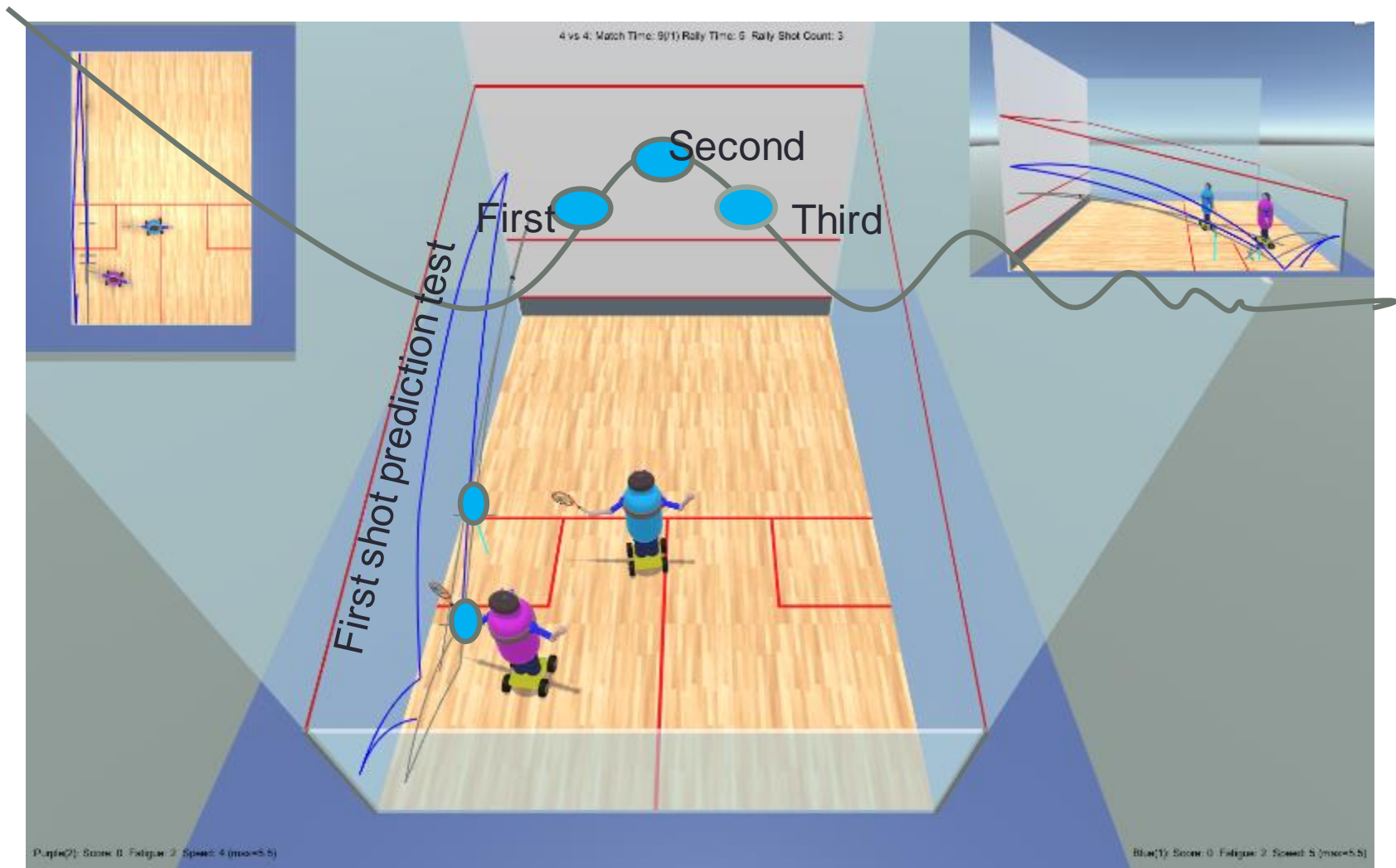


Predictive Thinking



Predicting

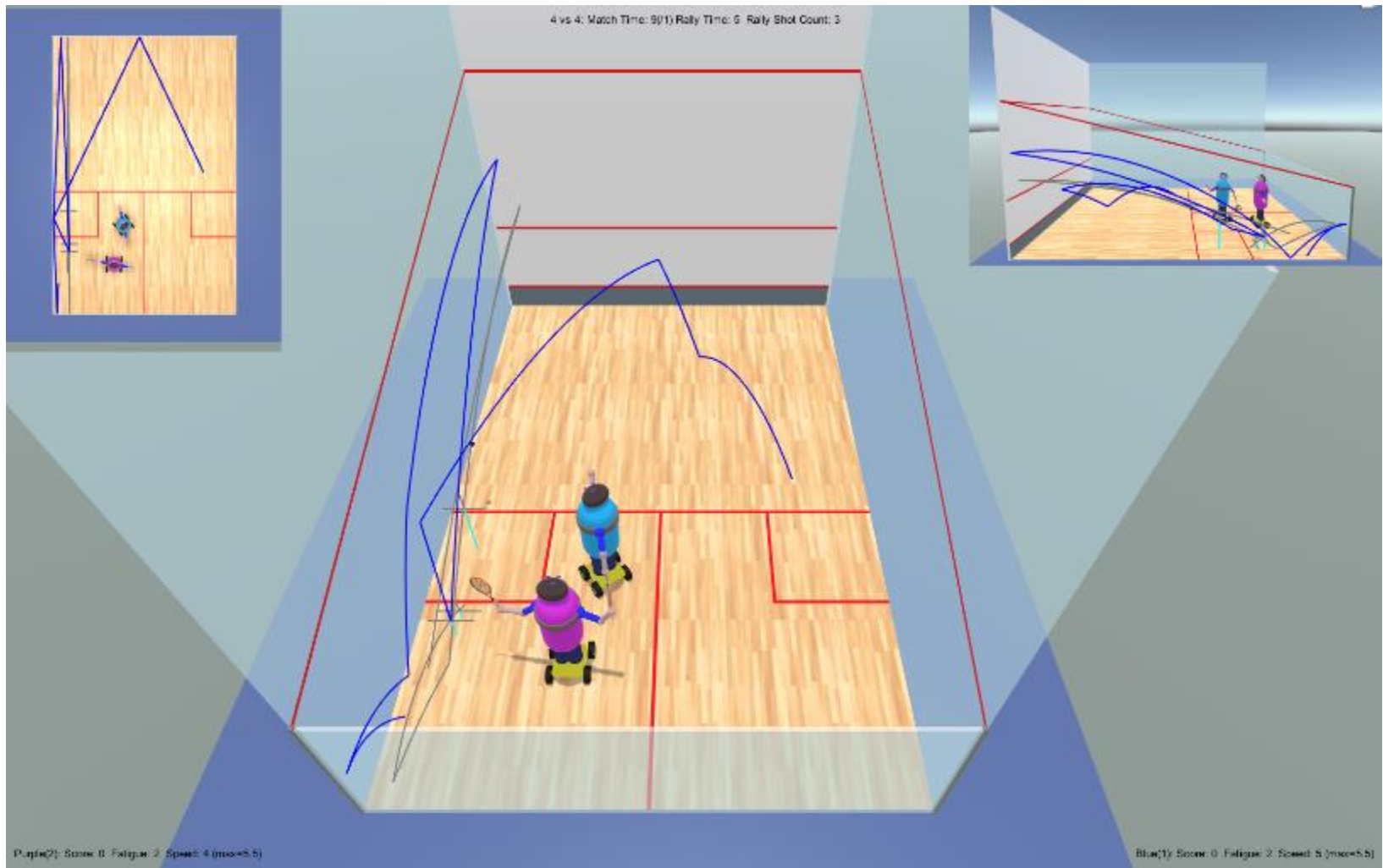
9.050 PLAYER-ONE PROCEDURAL PRODUCTION-FIRED TEST-
SHOT-Z22-Z23-STHI
Testing shot 51 0
better predicted value 2 for 51



Predicting

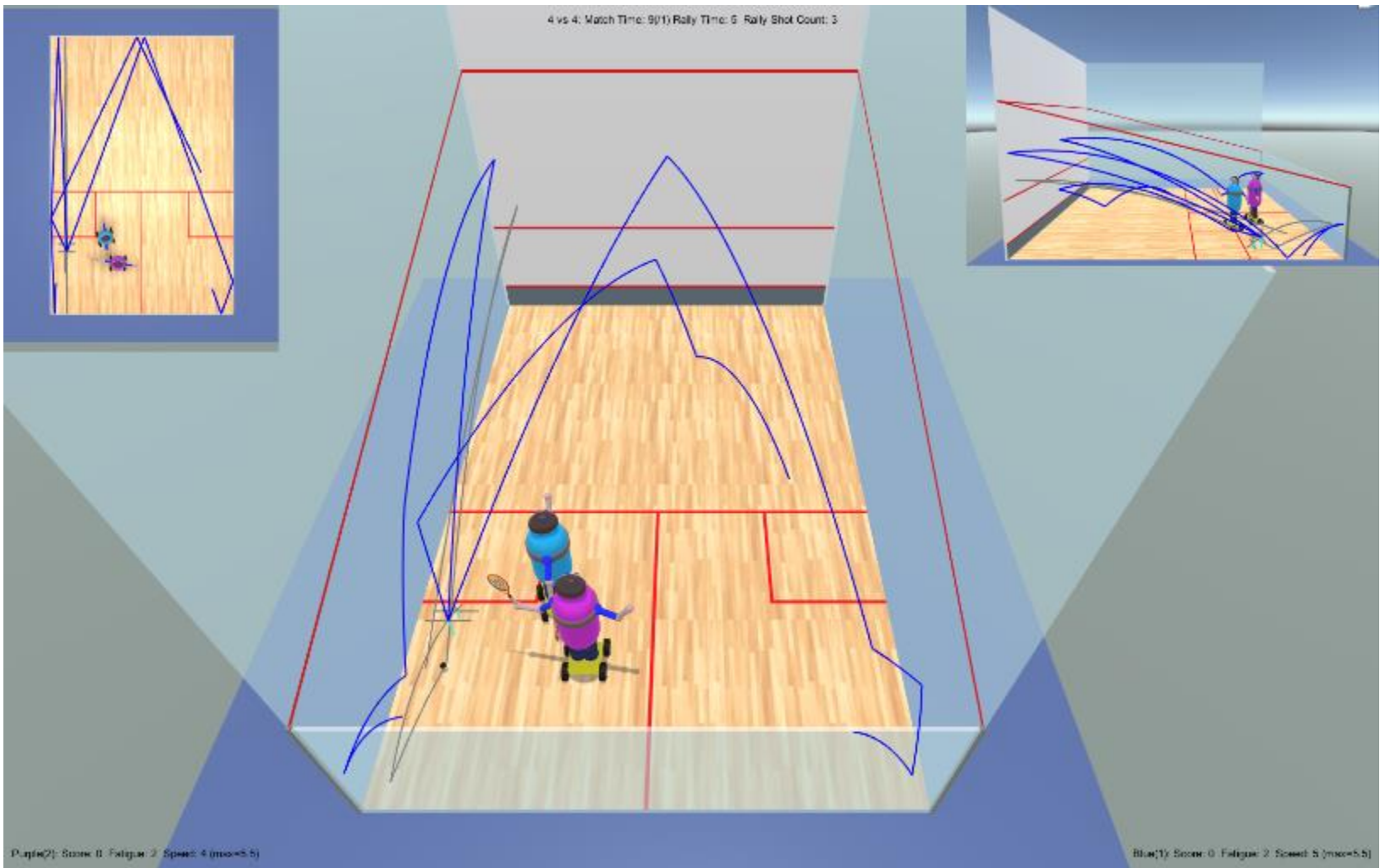
9.250 PLAYER-ONE PROCEDURAL PRODUCTION-FIRED TEST-
SHOT-Z22-Z23-BODF

Testing shot 23 1
predicted value 1 for 23



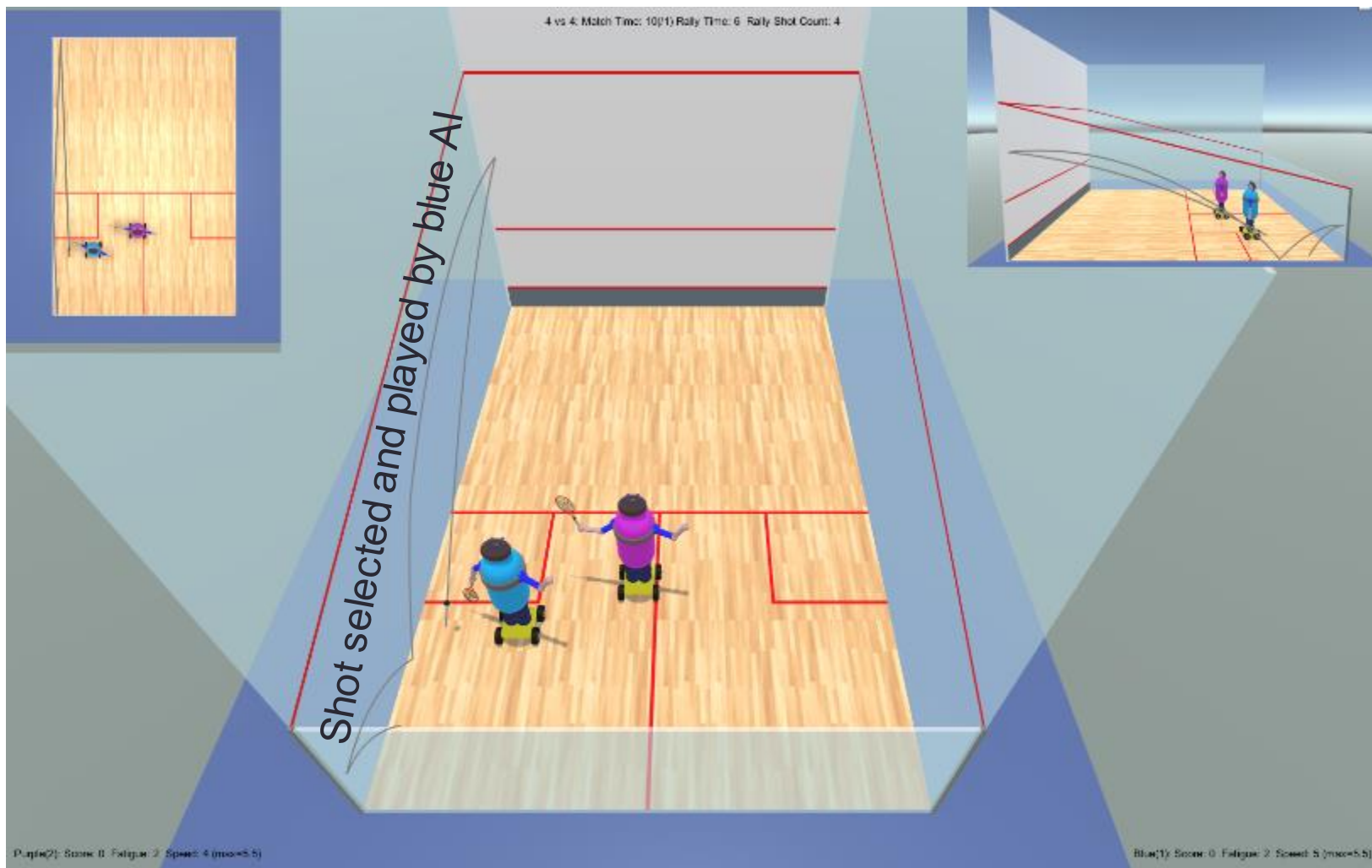
Predicting

9.450 PLAYER-ONE PROCEDURAL PRODUCTION-FIRED TEST-
SHOT-Z22-Z23-CRHI
Testing shot 52 2
predicted value 1 for 52

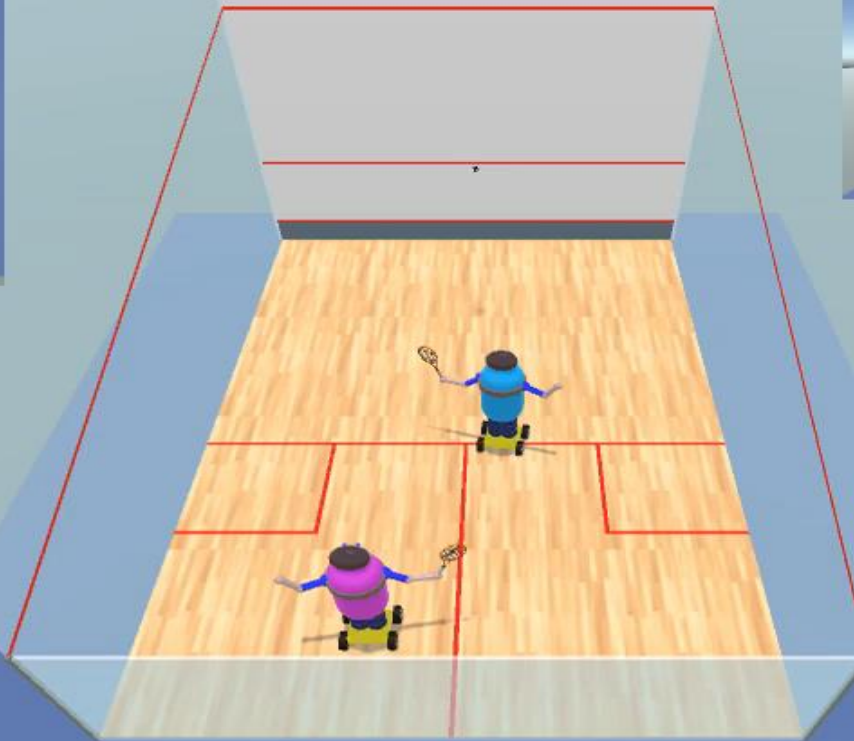
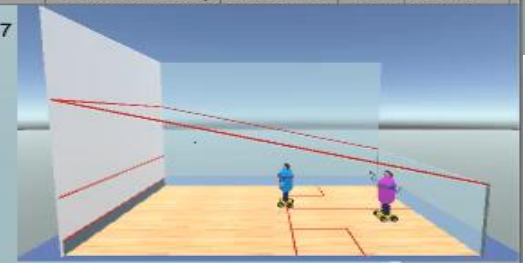


Predicting

9.850 PLAYER-ONE PROCEDURAL PRODUCTION-FIRED
FINAL-SHOT-SELECTION



4 vs 6: Match Time: 364/(/1) Rally Time: 9 Rally Shot Count: 7



Purple(2): Score: 4 Fatigue: 7 Speed: 1 (max=5.5)

Blue(1): Score: 8 Fatigue: 7 Speed: 5 (max=5.5)

418 0

```

370.200 PLAYER-TWO PTAM SET-BUFFER-CHUNK SPATIAL SPATIAL-STATE1850
370.200 PLAYER-TWO PTAM SET-BUFFER-CHUNK SITUATIONAL SITUATIONAL-STATE1850
370.250 PLAYER-ONE PROCEDURAL PRODUCTION-FIRED WAIT-REQUEST-STATE
370.250 PLAYER-TWO PROCEDURAL PRODUCTION-FIRED WAIT-REQUEST-STATE
370.400 PLAYER-ONE PTAM SET-BUFFER-CHUNK SPATIAL SPATIAL-STATE1851
370.400 PLAYER-ONE PTAM SET-BUFFER-CHUNK SITUATIONAL SITUATIONAL-STATE1851
370.400 PLAYER-TWO PTAM SET-BUFFER-CHUNK SPATIAL SPATIAL-STATE1851
370.400 PLAYER-TWO PTAM SET-BUFFER-CHUNK SITUATIONAL SITUATIONAL-STATE1851
370.450 PLAYER-ONE PROCEDURAL PRODUCTION-FIRED WAIT-REQUEST-STATE
370.450 PLAYER-TWO PROCEDURAL PRODUCTION-FIRED WAIT-REQUEST-STATE
370.600 PLAYER-ONE PTAM SET-BUFFER-CHUNK SPATIAL SPATIAL-STATE1852
370.600 PLAYER-ONE PTAM SET-BUFFER-CHUNK SITUATIONAL SITUATIONAL-STATE1852
370.600 PLAYER-TWO PTAM SET-BUFFER-CHUNK SPATIAL SPATIAL-STATE1852
370.600 PLAYER-TWO PTAM SET-BUFFER-CHUNK SITUATIONAL SITUATIONAL-STATE1852
370.650 PLAYER-ONE PROCEDURAL PRODUCTION-FIRED WAIT-REQUEST-STATE
370.650 PLAYER-TWO PROCEDURAL PRODUCTION-FIRED WAIT-REQUEST-STATE

```

2. COGNITIVE AND NON-COGNITIVE ARCHITECTURES

more physical pressure, but if they reach it with a bit of time to spare it opens up a lot of attacking shots.
Squash is also a game of angles, much like a real-time game of soccer. Judging and playing the angles is an important part of the game.

Using squash as the test scenario provides a known rule set for the game and existing tactical knowledge for implementing the AI models.

Two predictive elements were added to the existing ACT-R architecture. The predictive module always provided a prediction of the ball's flight path for the purpose of intercepting and hitting the ball. A further predictive element was added that allowed the AI model to evaluate the possible actions with a simulation to determine the likely outcome of those actions. Essentially, the model was able to ask very simple "what if" questions about how its own actions might play out in the future. Performance change due to the ability to simulate and predict actions was the metric for answering the research question.

The cognitive models implemented included three different mechanisms for choosing shots to play during a game of squash: 1) a pure random shot selection to act as a shot selection model; 2) a model that used rules to implement a predict shot outcome; and 3) a model that used simulation to predict shot outcomes before selecting a shot type.

The models were evaluated by playing them against one another. Data gathered from the squash play/simulation sessions recorded detailed information about shot selection, allowing analysis of the behaviour of the models and the effectiveness of their respective shot selection methods.

Section II, of this paper, gives some background to cognitive and non-cognitive architectures. In Section III a description of the research undertaken and methodology used is given. Section IV describes the AI modelling and how prediction was incorporated. Section V discusses the results obtained.

II. COGNITIVE AND NON-COGNITIVE ARCHITECTURES

Cognitive architectures are based on theories of how the human mind reasons to solve problems. These are AI systems based on human cognitive processes that work through Computational Theory of Mind [2]. They are based on the mind works like a computer running a program, using logic and symbolic information, to work through, and solve, problems.

The cognitivist approach follows a rule-based manipulation of symbols, and uses patterns of symbols, as designed by humans, to represent the world [14]. A key characteristic is that the mapping of perceived objects to their associated symbols is either defined by humans, or learned in a way that can be viewed and interpreted by humans. Decisions about which actions to perform are derived by reasoning of the internal symbolic representations of the

episodic and semantic memory, reinforcement learning, and continuous model learning; it also incorporates a simultaneous localisation and mapping (SLAM) module, rules, and semantic memory implemented as declarative associations. It uses both symbolic and non-symbolic representations. A number of architectures similar to SOAR and ACT-R are reviewed in [16]. Lockner et al. takes an alternative approach to cognitive architecture for robotics, proposing a context-based approach that overcomes the sensor grounding problem by matching perception and sensor data to extensive cloud-based and annotated repositories of images, video, 3D models, etc. [17].

Most operational robots do not use cognitive architectures. Instead, traditional robotic research and control has focused on software solutions that solve problems having well formulated solutions; this can be referred to as the algorithmic approach [1]. These systems are particularly suited to well-defined tasks and domains, and form a foundation for robotic capabilities. However, there is a need for higher level cognitive abilities to deal with less well defined problem solving and uncertain situations where the scope for the development of algorithmic solutions. It is in these situations that cognitive architectures might provide an effective solution.

The **subsumption** architecture is another alternative to cognitive architectures for robot control. The **subsumption** architecture approaches intelligence from a different perspective. Rather than rules that lay out a series of steps to accomplish a task, it uses a very sparse rule set that responds to sensor values to generate control outputs [18][19][20]. Brooks describes **subsumption** as a layered finite state machine where low-level functions, like "avoid obstacles", are subsumed into higher-level functions, like "wander and explore". Each successive layer gives increasing levels of competence. Lower levels pre-empt the higher levels, such that a robot can explore, but will avoid obstacles when necessary.

Key aspects of **subsumption** are: that it contains no high level declarative representations of knowledge; no declarative symbolic processing; no expert systems or rule matching; and it does not contain a problem-solving or learning module [2]. In order to generate corresponding control outputs. So in a canonical **subsumption** architecture, there is no inherent mechanism for problem-solving in an algorithmic way.

Subsumption can be very powerful. It is based on the concept that the environment stands for itself, i.e., the architecture reacts directly to environmental features, without a mediating representation. It is a functional architecture without being, or using, a declarative model of the external world. However, without additional features, like memory and

difficult to implement, and could be easily incorporated into other cognitive architectures.

Society of Mind proposes a theory that intelligence arises from the interactions of large numbers of simple functions [21][22]. This is not an actual architecture, but rather a theory that argues against the idea that a single unified architecture or solution can account for intelligent behaviour. A robotic AI can be created completely within a single architecture, using rules that control every aspect of the decision making process, but those architectures are not always ideal for every style of decision-making. Society of Mind theory argues for a modular approach to implementing intelligence. Implementing simulation as an extension to a cognitive architecture, but using an external 3D engine to model the environment, follows this concept. The simulation is a separate, specialised function for solving problems in dynamic physical situations.

ACT-R is a hybrid cognitive architecture consisting of both symbolic and sub-symbolic components [24][25]. It is a goal-oriented architecture. The symbolic data consists of facts and production rules. The sub-symbolic data is metadata about facts and production rules that control which facts are recalled and which production rules are chosen to fire when multiple facts and rules are available.

ACT-R consists of a number of modules that interact through a production system that selects rules to execute, (Figure 2). Each module has a buffer, which can hold a chunk of data (a key-value pair structure) representing the current state of the module.

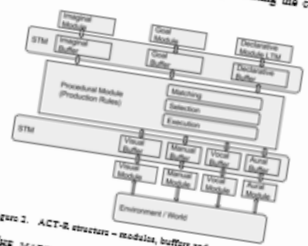


Figure 2. ACT-R structure - modules, buffers and production system.

THE MATCHING SYSTEM LOOKS FOR PATTERNS IN THE ENVIRONMENT THAT IT CAN USE TO SELECT A PRODUCTION RULE TO POTENTIALLY FIRE FROM AMONGST THOSE AVAILABLE. EACH PRODUCTION RULE INCLUDES A PATTERN THAT GIVES THE CONDITIONS UNDER WHICH IT CAN FIRE. PRODUCTION RULES CAN MAKE REQUESTS OF THE MODULES, SO THEY CAN CHANGE THEIR OWN INTERNAL STATE.

ACT-R to constitute the Predictive-ACT-R (PACT-R) architecture.

A. Research Design

The research consisted of developing and implementing a virtual environment for testing; developing a cognitive system; and developing AI models to test the system.

An ACT-R cognitive module was developed that mapped a symbolic representation of a simulated environment into the ACT-R framework. This module gave the required PACT-R functionality for interpreting and acting within the environment, as well as providing simple predictive capabilities using simulation.

The use of prediction and simulation in ACT-R was evaluated by comparing the performance of several models that each implemented different levels of prediction. The aim was to compare not only their performance, but also how easy/simple it was to model and use a predictive AI.

B. Implementation

The research consisted of three components. The first was the design and implementation of a cognitive module within ACT-R. This module gave models access to predictions about physical events, as well as a mechanism to take actions.

The second element was a simulation of the game of squash implemented in the Unity™ game engine. Parts of the PACT-R module were also implemented with Unity™, and communicated with the prediction module in PACT-R. The Unity™ components of ACT-R were the physics simulation and prediction engine.

The final element was modelling squash-playing AI, cross-comparison.

C. Using Simulation and Prediction within a Cognitive Architecture

The research investigated the use of a physics engine to provide prediction for a cognitive architecture. The concept is to provide a physics engine that can model and simulate the environment of a robot controlled by a cognitive AI. The environment provides a symbolic representation of the cognitive model (the production rules) the information it needs to understand and act within its environment.

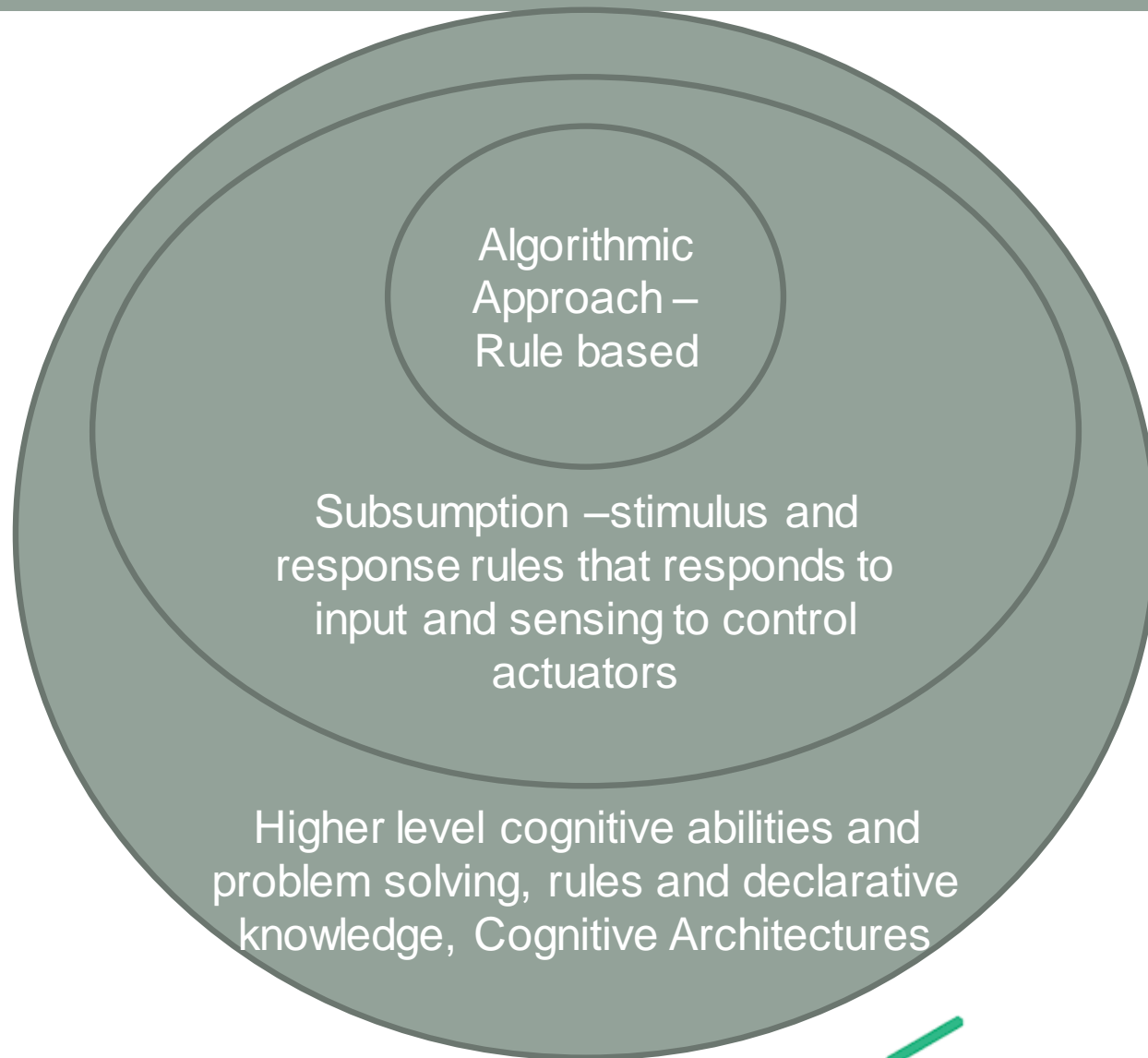
One way of using this information is to explicitly encode rules that check for certain conditions, for example, whether an object is in a certain position, or is moving in a particular direction; or for the relationships between objects in a particular environment, for example, whether an object is to the left of another object [17][26]. From this, the rules can encode appropriate actions for the robot to take.

This research explored an alternative approach. Rather than using explicit rules to interpret and decide actions, a



UNIVERSITY of TASMANIA





3. METHODOLOGY

- A. Research Design
- B. Implementation
- C. Using Simulation and Prediction within a Cognitive Architecture
- D. PACT-R Module Implementation in ACT-R

difficult to implement, and could be easily incorporated into other cognitive architectures.

Society of Mind prepares a theory that intelligence arises from the interactions of large numbers of simple functions [21]. This is not an actual architecture, but rather a theory [22] that serves as the idea that a simple unified architecture can account for intelligent behaviour.

A robotic AI can be conceived completely within a simple decision machine structure, but these architectures are not ideal for every style of domain-specific. Society of Mind theory argues for a modular approach to implementation of intelligence. Implementing simulation as an extension to a cognitive architecture, but using an external JD engine to model the environment, follows this concept. The simulation is a separate, specialised function for solving problems in dynamic physical situations.

ACT-R is a hybrid cognitive architecture consisting of both symbolic and sub-symbolic components [23]. It is a facts and production rules. The symbolic data consists of about facts and production rules that control which facts are recalled and which production rules are chosen to fire when multiple facts and rules are available.

ACT-R consists of a number of modules that interact through a production system that selects rules to execute. Figure 2. Each module has a buffer, which can hold a chunk of data in a hierarchical (tree) structure representing the current state of that module.



Figure 2. ACT-R structure - modules, buffers and production system.

The MATCHING SYSTEM LOOKS FOR PATTERNS IN THE BUFFER THAT IT CAN USE TO SELECT A PRODUCTION RULE TO EXECUTE. THE PRODUCTION RULES ARE STORED IN THE PRODUCTION RULE BUFFER. A PATTERN THAT TRIGGERS A PRODUCTION RULE INDICATES TO THE MATCHING SYSTEM THAT IT CAN EXECUTE. PRODUCTION RULES CAN MAKE REQUESTS OF THE MODULES, SO THEY CAN CHANGE THEIR OWN INTERNAL STATE.

III. Methodology

This section describes the research design and the implementation of the prediction and simulation extensions to

ACT-R to constitute the Predictive-ACT-R (PACT-R) architecture.

A. Research Design

The research consisted of developing and implementing a virtual environment for testing, developing a cognitive system, and developing AI models to test the system.

An ACT-R cognitive model was developed that merged a symbolic representation of a simulated environment into the ACT-R framework. This model gave the required PACT-R environment, as well as providing simple predictive functionality for interpreting and acting within the capabilities of using simulation.

The use of prediction and simulation in ACT-R was evaluated by comparing the performance of several models that each implemented different levels of prediction. The aim was to compare not only their performance, but also how easy it was to model and use a predictive AI.

B. Implementation

The research consisted of three components. The first was the design and implementation of a cognitive model within physical events, as well as a mechanism to take actions.

The second element was a simulation of the same of squash implemented in the Unity™ game engine. Parts of the PACT-R module were also implemented with Unity™ and components of ACT-R were implemented in PACT-R. The prediction engine.

The final element was modelling squash-playing AI. Three evaluation models were developed for testing and cross-comparison.

C. Using Simulation and Prediction within a Cognitive Architecture

The research investigated the use of a physics engine to provide prediction for a cognitive architecture. The concept environment of a robot controlled by a cognitive AI. The simulation provides a symbolic representation of the environment to a cognitive architecture. This gives the cognitive model (the production rules) the information it needs to understand and act within its environment.

One way of using this information is to implicitly create rules that check for certain conditions, for example, whether an object is in a certain position, or is moving in a particular direction, or for the relationships between objects in the environment. For example, whether an object is to the left of another object [17]. From this, the rules can create appropriate actions for the robot to take.

This research explored an alternative approach. Rather than using explicit rules to interpret and decide actions, a simulation of the environment was used to test actions. Figure 3 shows a high-level diagram of this approach. First, the environment was modelled in a physics engine that uses environment and state information to a cognitive model. From the information available, the cognitive model can determine

what actions might be appropriate. Rather than determining the best, with rules, it passes the choices back to the physics engine to be simulated, which then generates a prediction of the outcome of that action. The results of each prediction are passed back to the cognitive model, which then decides which one is the most appropriate, and will therefore be used.



Figure 3. Diagram of PACT-R concept, environment is simulated and simulated actions are used under the control of a cognitive model.

D. PACT-R Module Implementation in ACT-R

The prediction system is implemented as an ACT-R module that both controls a robot and does a simulation of the robot's environment. The module is, logically, a functional part of the implementation in the ACT-R framework, but in the implementation it is broken out into the other module the Unity™ game engine, which includes a physics engine and also uses the virtual world of the robot's environment.

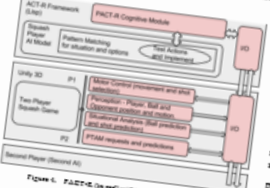


Figure 4. PACT-R (in red) within the ACT-R and Unity.

The ACT-R component of the system maintains the current simulation and prediction state for use by the AI module, while the Unity™ component of the system contains

a **simulated** physics engine that can simulate both the squash ball's path and the outcome of those played by the robot. The two components of the module connect via a Universal Protocol (UDP), a standard part of the Internet Protocol (IP).

For PACT-R, the cognitive module represents implicit knowledge of the sort that a squash player learns over many years. Part of this implicit knowledge is the muscle memory that allows a player to move correctly and hit a ball properly. Another part is an explicit understanding of the tactical situation. Coding this implicit knowledge into an AI model would be difficult and counterproductive. A squash player does not think about this, but rather uses it as a base to decide between 'how do you do something?' and 'what should you do?' Implicit knowledge encodes the 'how', while the simulation provides a base for deciding 'what'.

The PACT-R module has to work through ACT-R, therefore, implemented as an extended predictive module in a production system buffer, one that corresponds not just to, and the environment. The prediction module communicates with the simulation engine to both receive predictions and to request predictions based on possible actions of the AI model. Figure 4 shows the modified ACT-R framework with the additional prediction module.

IV. AI Model and Analysis Framework

This section presents the outline of the AI model as a modular level, rather than dealing with the details of prediction module in ACT-R. Then, the implementation of the simulation with the AI model is presented, together with its evaluation and analysis framework for these models.

A. Prediction Module

The simulated task playing squash that the AI has to perform is dynamic; the ball is in continuous motion, and can follow complex paths as it interacts with the walls and floor. Likewise, the AI's robotic avatar is moving, as is its opponent. ACT-R is designed to look for, and recognize, patterns in information in its buffers. The buffers hold information representing both the current world, and the AI model's internal state. ACT-R can work with values and do simple comparisons, but does not perform calculations and functions (through its internal state). It is possible to call Lisp routines to break a situation into a simple symbolic representation that the AI model can reason about, by using explicit rules to interpret and decide actions.

For a complex, dynamic situation this may present a problem, since an AI model requires deliberation (i.e. 'thinking') time. That is, it needs time to **reprocess** a pattern and fire a prediction for the situation the pattern represents. For a dynamic situation, by the time a pattern has been processed, and acted upon, the situation may have already changed to something different.



UNIVERSITY of TASMANIA

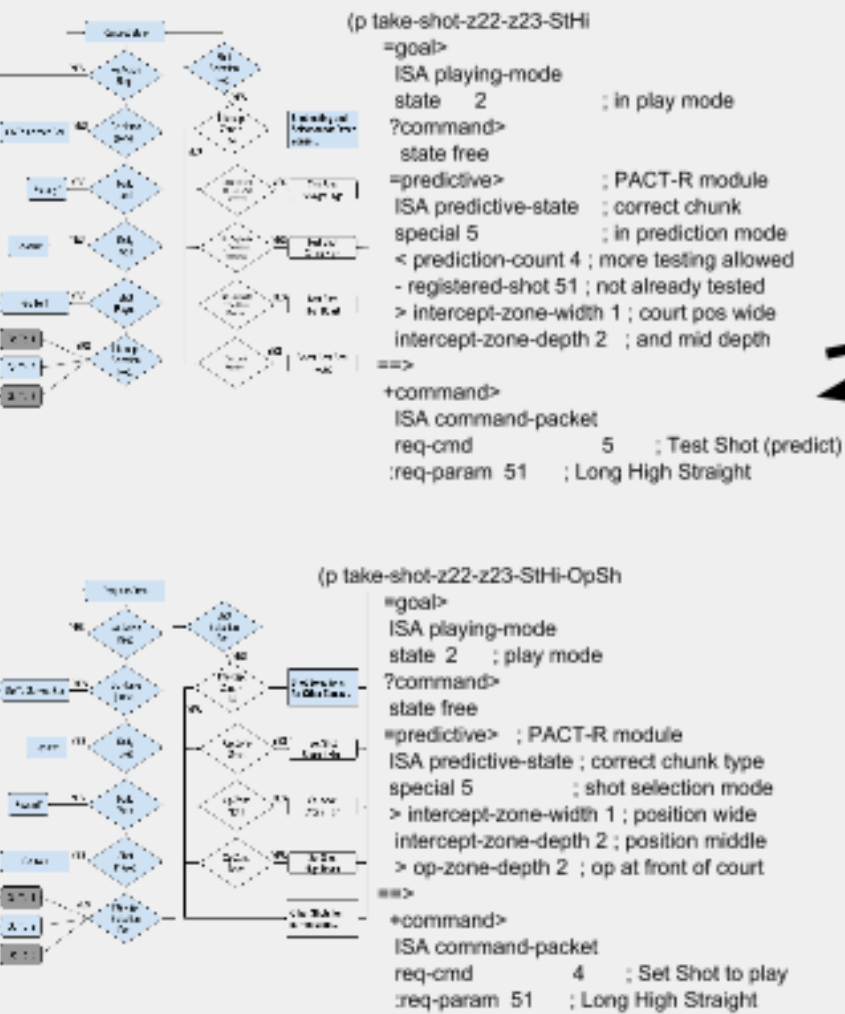


Research Design

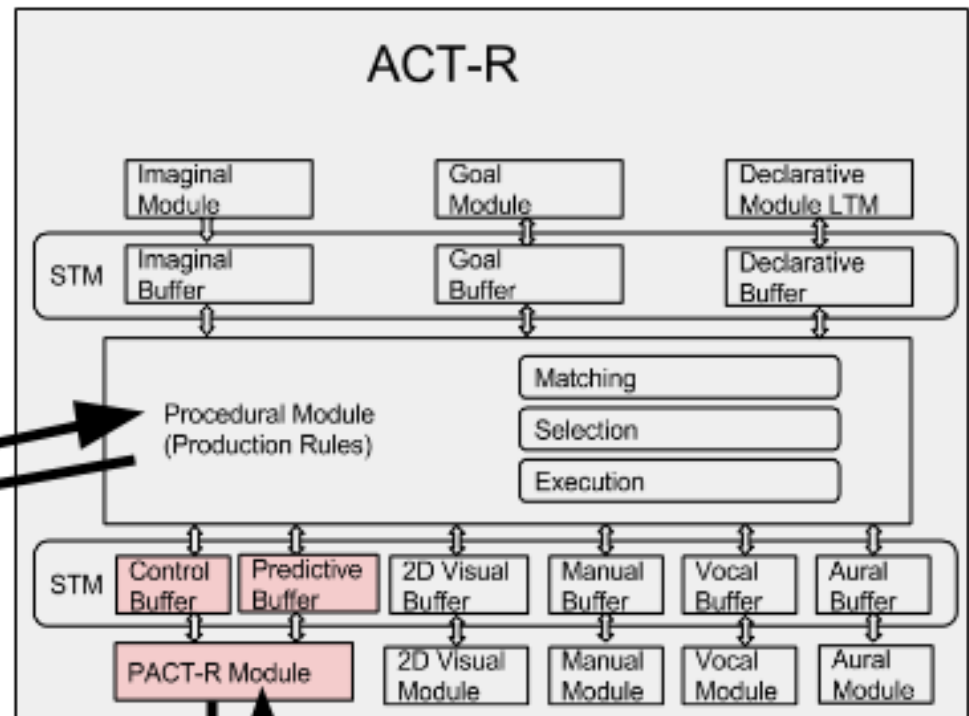
- Experimental software development
- Implement a new cognitive module for ACT-R
- Module gives an interpretation of a simulated world.
- Implement multiple ACT-R AI models and have them compete head to head
- Implement a suitable simulated environment and scenario
- Base line AI model and enhanced heuristic models for comparative testing
- Model with speculative prediction that tests prospective actions for positive or negative outcomes
- Statistical analysis of comparative results.

3. METHODOLOGY *A. Research Design*

Rules



ACT-R



Zones

Unity 3D

Object detection - Zones for Player, Ball and Opponent.

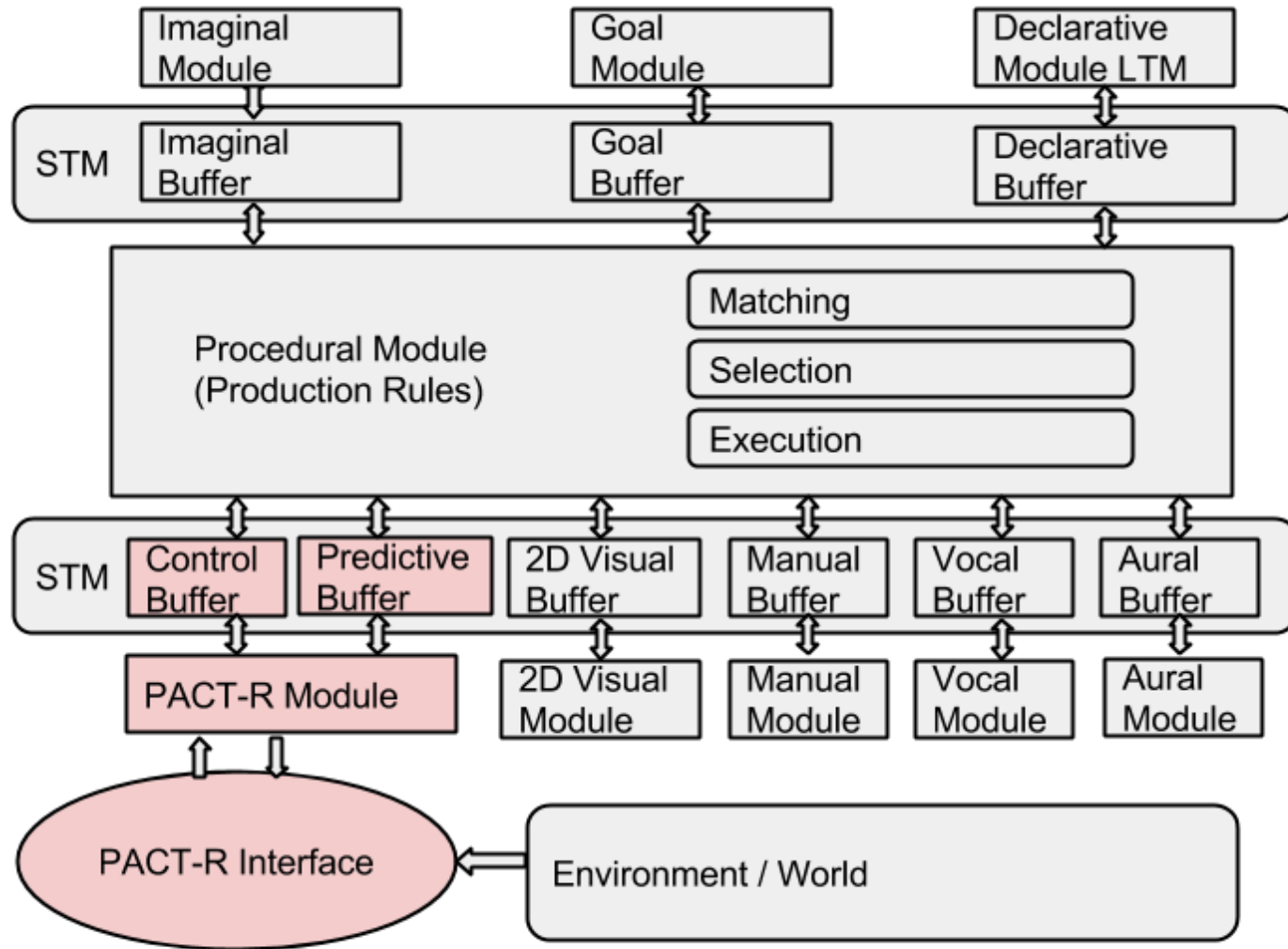
Motor Control - Movement and Shot Selection

Prediction - Ball path; Interception and Hit positions; Action (Shot) testing and evaluation



UNIVERSITY
TASMANIA

3. METHODOLOGY *B. Implementation*



4. AI MODELLING AND PREDICTION

A. Prediction Models

B. Evaluation and Analysis

A conceptual physics engine that can simulate both the squash ball's path, and the outcome of shots played by the robot. The two components of the module connect via a Universal Protocol (UP), a standard part of the Internet.

For FACT-R, the cognitive module represents implicit knowledge of the sort that a squash player learns over many years. Part of this implicit knowledge is the muscle memory that allows a player to move correctly and hit a ball properly. Another part is an implicit understanding of the spatial situation. Coding this implicit understanding into an AI model would be difficult and counterproductive. A squash player does not think about this, but rather uses it as a base to decide what they should do next. Essentially, the difference resides between 'how do you do something?' and 'what should you do?'. Implicit knowledge encodes the 'how', while the simulation provides a basis for deciding the 'what'.

The FACT-R module has to work through ACT-R, therefore, implemented as an additional cognitive module. It provides two buffers, one that commands are sent to, and other that gives the model access to a simplified view of the environment. The prediction module communicates with the simulation engine to both receive conditions and to update predictions based on possible actions and to update a shows the modified ACT-R framework with the additional prediction module.

IV. AIMODELLING AND PHENOMENA

This section presents the outline of the AI models at a conceptual level, rather than dealing with the details of modelling them in ACT-R. Then, the implementation of the prediction module in ACT-R is presented, together with its connections with the AI models, followed by a description of the evaluation and analysis framework for these models.

A. Prediction Model

The simulated task, playing squash, that the AI has to perform is dynamic; the ball is in continuous motion, and can follow complex paths as it interacts with the walls and floor. Likewise, the AI robotic avatar is moving, so its opponent's movement in its buffers. The buffers hold information concerning both the external world, and the AI model's internal state. ACT-R can work with values and do simple calculations, but since complex calculations and functions are required, ideally, the modules should do the hard work of breaking a situation into a simple symbolic representation that the AI model can reason about, by searching for patterns and relationships.

For a complex dynamic situation this may present a problem, since an AI model requires deliberation ('thinking') time. That is, it needs time to reason about, and fire a prediction for the situation the pattern represents. For a dynamic situation, by the time a pattern has been recognized and acted upon, the situation may have already changed to something different.

The simulation-based module described here abstracts away the details of the environment into a simple set of relationships and events representing the elements in the simplified FACT-R, the abstraction focuses on the specifics of the game of squash.

For squash, FACT-R identifies three actors: rally opponents about the approximate locations of these actors within the court and information about what is happening, is about to happen, or what might happen. Conspicuously absent from the information is real coordinates and vectors of motion. While ACT-R can work with this sort of information, it would lead to a set of rules with a lot of special case calculations and conditions that might not be processed rapidly enough for real-time performance.

For this research, a baseline capability of the prediction module included a prediction about the immediate incoming flight path that the AI model could use to intercept the ball, an appropriate court position, in order to play a shot. This prediction was made following the opponent's shot when the ball's position and velocity could be determined. The ball's path was simulated in the physics engine, which tracked where the ball would travel if it was determined that the ball would be bounced on the floor for a second time. This path would have been used in the prediction module to determine ball paths were then used in the prediction module to determine locations where the player would intercept and hit the ball, based on their own movement ability.

The interest positions were placed in the prediction module buffer used by the AI model, which allowed the model to intercept the ball without any further processing. This would have introduced more complexity to the model, but this would have introduced more complexity to the model, making it difficult to determine cause and effect. For this reason, AI control and reasoning was limited only to the shot selection strategy.

To know where the ball and the player were within the court, the squash court was broken into strategic zones and all positions were given some numbers. The first strategy implemented in the module was also based on zones, where the ball was intercepted. The zones and shots are based on a squash training skills commonly used to teach players basic strategy.

B. Evaluation and Analysis

These models were developed and evaluated. The first model was a basic random shot selection model that functioned as the baseline to determine whether shot selection by the other models was better than random chance. The second model was a heuristic model that had an explicit shot selection rule-set derived from the human developer's experience of playing squash. This model's purpose was to provide an alternative method to the prediction

The third model used the predictive features of FACT-R to test shots for their likely outcome.

In order to evaluate the performance of the three models, a large amount of automatic data gathering and logging was conducted from the virtual environment. This data gave both how they were or lost.

The data collected from the experiment was the result of player rallies between two competing AI models. The models were tested over a large number of rallies to produce data for a statistical analysis of the relative performance of the models.

For each test session the only variables were the shot selection strategies of the two competing AI models. Ten sessions consisted of two AI models (out of three) loaded into the ACT-R environment, playing against each other over a series of rallies. A rally is where the two players alternate shots until one is unable to retrieve or return the shot, and therefore loses. Data recorded included shot selection and state during the rally, and the final results of each rally. This generated a large sample set of data.

Squash starts with a serve from one player to another. For a test run, the serve was alternated so there was no bias to advantage to either model. Player 1 always started on the forehand side (right), and player 2 on the backhand. The players were ambidextrous with no advantage to either side (unlike human squash players).

V. RESULTS AND DISCUSSION

The three models discussed here all follow the same basic strategy. They have to choose from three or four shots available for the zone where the ball is to be hit. The basic model did not use any additional logic to choose a shot. The other two models tried to choose a shot that would force the opponent to have to travel the furthest to reach the ball in order to play their next shot.

A. Basic Random Shot Selection Model

The first AI model developed was a random shot selection model. This model was a random shot selection strategy where shots for each court zone for ACT-R to choose with its production rules. With no additional conditions in the model other than the court zone, a shot would be chosen at random from those available.

This model acted as a baseline control. It was also the only model used during development and balancing of the simulation and physics engine.

B. Heuristic Shot Selection Model

The second model was a heuristic model that used ACT-R production rules that implemented a simple squash strategy, which tried to choose shots that would be directed to an area of the court where the opponent was not present. For example, if the opponent was deep in the court (i.e. close to the front wall of the court), it would choose a shot close to the front opponent was on the forehand side, it would choose a shot selection rule for each zone were



UNIVERSITY of TASMANIA



Three models compared:

1. **Random** Squash Shot Selection Model
2. **Heuristic** Squash Shot Selection Model
3. **Predictive** Squash Shot Selection Model



5. RESULTS AND DISCUSSION

- A. Basic Random Shot Selection Model
- B. Heuristic Selection Model
- C. Predictive Selection Model
- D. Performance

implemented using this simple strategy. In real squash, this approach is a good starting point for any human player.

Figure 5 is a flow chart representation of the heuristic model, although it only shows one shot selection choice, rather than the many that were required to model all court areas. It should be noted that for ACT-R, step-by-step fashion like a flow chart does not proceed in a representation is used to show the logic, rather than the functioning of the model.

allowing the AI model to choose a possible shot before passing the information to the prediction model for simulation and predicting its consequences. The model would simulate how the shot would play out to predict where the opponent would be when the shot was played, and how much difficulty they would have in then removing it and giving a counter shot. The prediction was based on how strategy as the heuristic model, trying to find a shot that was as far from the opponent as possible.

The predictive system has one advantage over the heuristic: as it is calculating the path of the shot under test, it also knows that the opponent could not return. This result was passed back to the AI, which allowed the predictive model to find, and choose, three occasional winning shots.

Figure 6 shows the prediction model as a flowchart, and a sample rule. Unlike the heuristic model's 45 rules, this model only requires 26 rules for shot selection. Each rule defines a shot to be tested for a particular zone of the court.

```
CP STRAT-SHOT-222-223-224-225-226-227-228-229-230-231-232-233-234-235-236-237-238-239-240-241-242-243-244-245-246-247-248-249-250-251-252-253-254-255-256-257-258-259-260-261-262-263-264-265-266-267-268-269-270-271-272-273-274-275-276-277-278-279-280-281-282-283-284-285-286-287-288-289-290-291-292-293-294-295-296-297-298-299-300-301-302-303-304-305-306-307-308-309-310-311-312-313-314-315-316-317-318-319-320-321-322-323-324-325-326-327-328-329-330-331-332-333-334-335-336-337-338-339-340-341-342-343-344-345-346-347-348-349-350-351-352-353-354-355-356-357-358-359-360-361-362-363-364-365-366-367-368-369-370-371-372-373-374-375-376-377-378-379-380-381-382-383-384-385-386-387-388-389-390-391-392-393-394-395-396-397-398-399-400-401-402-403-404-405-406-407-408-409-410-411-412-413-414-415-416-417-418-419-420-421-422-423-424-425-426-427-428-429-430-431-432-433-434-435-436-437-438-439-440-441-442-443-444-445-446-447-448-449-450-451-452-453-454-455-456-457-458-459-460-461-462-463-464-465-466-467-468-469-470-471-472-473-474-475-476-477-478-479-480-481-482-483-484-485-486-487-488-489-490-491-492-493-494-495-496-497-498-499-500-501-502-503-504-505-506-507-508-509-510-511-512-513-514-515-516-517-518-519-520-521-522-523-524-525-526-527-528-529-530-531-532-533-534-535-536-537-538-539-540-541-542-543-544-545-546-547-548-549-550-551-552-553-554-555-556-557-558-559-560-561-562-563-564-565-566-567-568-569-570-571-572-573-574-575-576-577-578-579-580-581-582-583-584-585-586-587-588-589-590-591-592-593-594-595-596-597-598-599-600-601-602-603-604-605-606-607-608-609-610-611-612-613-614-615-616-617-618-619-620-621-622-623-624-625-626-627-628-629-630-631-632-633-634-635-636-637-638-639-640-641-642-643-644-645-646-647-648-649-650-651-652-653-654-655-656-657-658-659-660-661-662-663-664-665-666-667-668-669-670-671-672-673-674-675-676-677-678-679-680-681-682-683-684-685-686-687-688-689-690-691-692-693-694-695-696-697-698-699-700-701-702-703-704-705-706-707-708-709-710-711-712-713-714-715-716-717-718-719-720-721-722-723-724-725-726-727-728-729-730-731-732-733-734-735-736-737-738-739-740-741-742-743-744-745-746-747-748-749-750-751-752-753-754-755-756-757-758-759-760-761-762-763-764-765-766-767-768-769-770-771-772-773-774-775-776-777-778-779-780-781-782-783-784-785-786-787-788-789-790-791-792-793-794-795-796-797-798-799-800-801-802-803-804-805-806-807-808-809-810-811-812-813-814-815-816-817-818-819-820-821-822-823-824-825-826-827-828-829-830-831-832-833-834-835-836-837-838-839-840-841-842-843-844-845-846-847-848-849-850-851-852-853-854-855-856-857-858-859-860-861-862-863-864-865-866-867-868-869-870-871-872-873-874-875-876-877-878-879-880-881-882-883-884-885-886-887-888-889-890-891-892-893-894-895-896-897-898-899-900-901-902-903-904-905-906-907-908-909-910-911-912-913-914-915-916-917-918-919-920-921-922-923-924-925-926-927-928-929-930-931-932-933-934-935-936-937-938-939-940-941-942-943-944-945-946-947-948-949-950-951-952-953-954-955-956-957-958-959-960-961-962-963-964-965-966-967-968-969-970-971-972-973-974-975-976-977-978-979-980-981-982-983-984-985-986-987-988-989-990-991-992-993-994-995-996-997-998-999-1000
```

in the top right frame. In subsequent frames blue tracks appear which represent possible shots. In the final frame the eye shadow on the job hand side (bottom in green) is visible.

When developing the model, there was a clear advantage to the basic and predictive models over the heuristic model in the reduced number of rules required to implement the shot selection strategy. The basic and predictive models required 26 and 26 rules, respectively. The heuristic model required 45 rules to implement a simple shot selection strategy. The predictive system did have a disadvantage in the time it took to select a shot, it was not always able to complete its shot selection, and in that case it reverted to a random choice.

The three models that were developed could all play squash. The heuristic and predictive models both outperformed the basic model. The predictive system also outperformed the heuristic model, despite some limitations in its implementation.

VI. CONCLUSION

The research question asked "How can simulation and prediction improve decision quality in a cognitive architecture?" The answer to this is not straightforward. The results show that, within the limitations of the experiment, a predictive model – with an ability to use simulation to test its own actions to determine and evaluate their possible outcome – had a clear advantage over a model that used heuristic rules to select a shot, and in that case it reverted to a random choice.

It is not, perhaps, surprising that an approach that focuses on reasoning about a situation based only on where opponents are, how they were moving, etc., in the moment. The results of the investigation indicated that prediction provided a more effective appraisal of the value of an action, without requiring detailed rules.

There is a caveat here though: the evaluation of the heuristic model was an evaluation of its specific rule set and not more detailed situational knowledge that with a larger rule set and more detailed situational knowledge, it could have outperformed the predictive model. Indeed, both the heuristic and predictive models could have been developed further, to incorporate each other in a virtual sense.

However, there was another sense to the model – the predictive model only required 26 rules.

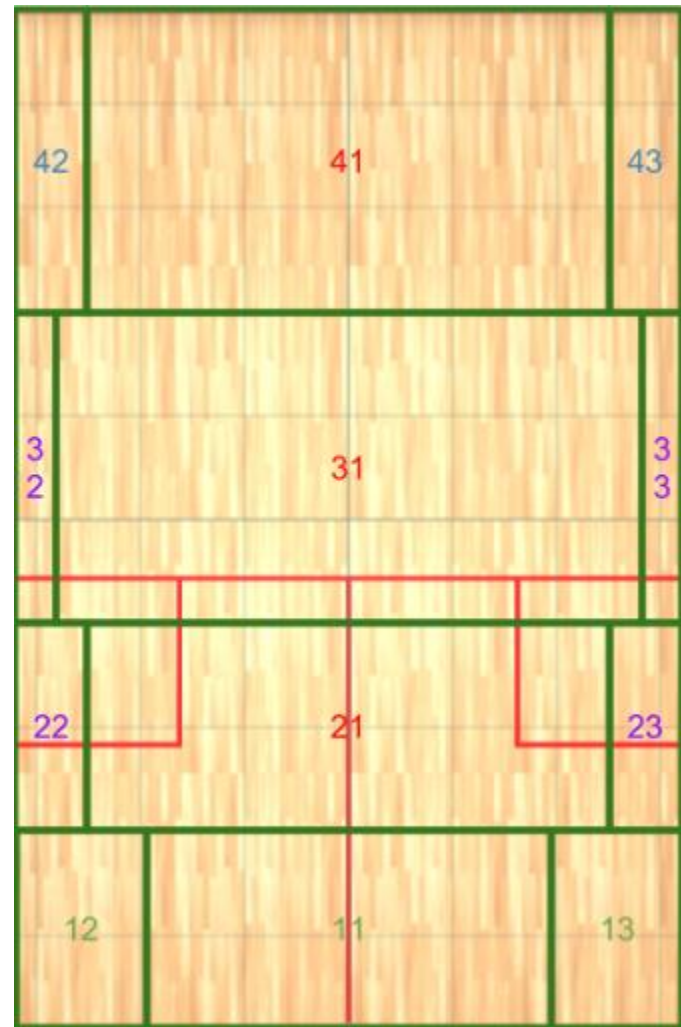
D. Performance

Figure 9 shows the player to player performance of all three models. When playing identical models against each other the results are even, as would be expected. Both heuristic and predictive models win over the basic random selection model, with a score of 815 to 228. The binomial test p-value for this is 0.0003, showing that this is unlikely to be due to random chance.

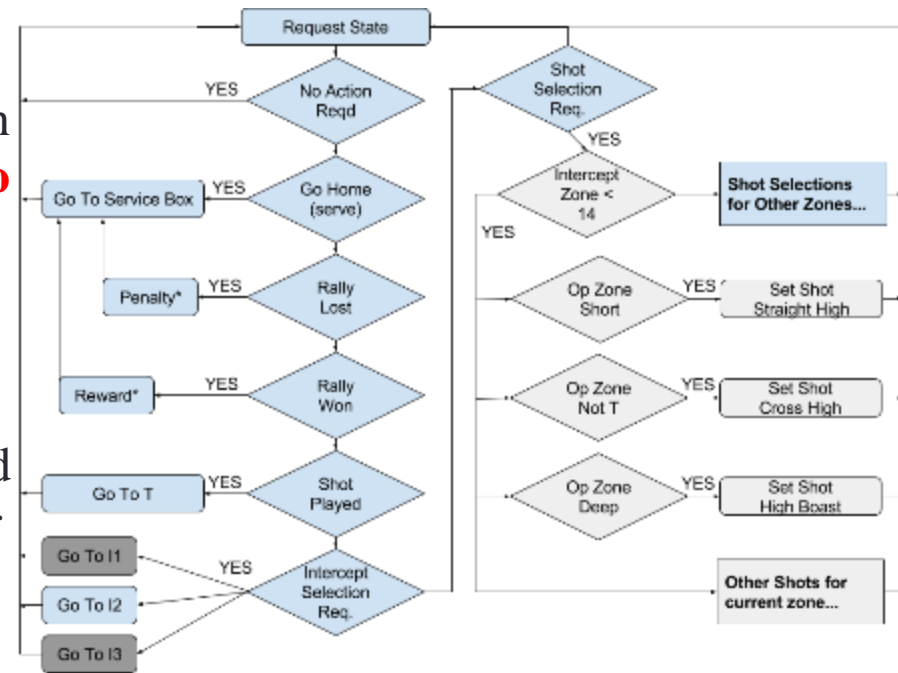
```
CP STRAT-SHOT-222-223-224-225-226-227-228-229-230-231-232-233-234-235-236-237-238-239-240-241-242-243-244-245-246-247-248-249-250-251-252-253-254-255-256-257-258-259-260-261-262-263-264-265-266-267-268-269-270-271-272-273-274-275-276-277-278-279-280-281-282-283-284-285-286-287-288-289-290-291-292-293-294-295-296-297-298-299-300-301-302-303-304-305-306-307-308-309-310-311-312-313-314-315-316-317-318-319-320-321-322-323-324-325-326-327-328-329-330-331-332-333-334-335-336-337-338-339-340-341-342-343-344-345-346-347-348-349-350-351-352-353-354-355-356-357-358-359-360-361-362-363-364-365-366-367-368-369-370-371-372-373-374-375-376-377-378-379-380-381-382-383-384-385-386-387-388-389-390-391-392-393-394-395-396-397-398-399-400-401-402-403-404-405-406-407-408-409-410-411-412-413-414-415-416-417-418-419-420-421-422-423-424-425-426-427-428-429-430-431-432-433-434-435-436-437-438-439-440-441-442-443-444-445-446-447-448-449-450-451-452-453-454-455-456-457-458-459-460-461-462-463-464-465-466-467-468-469-470-471-472-473-474-475-476-477-478-479-480-481-482-483-484-485-486-487-488-489-490-491-492-493-494-495-496-497-498-499-500-501-502-503-504-505-506-507-508-509-510-511-512-513-514-515-516-517-518-519-520-521-522-523-524-525-526-527-528-529-530-531-532-533-534-535-536-537-538-539-540-541-542-543-544-545-546-547-548-549-550-551-552-553-554-555-556-557-558-559-560-561-562-563-564-565-566-567-568-569-570-571-572-573-574-575-576-577-578-579-580-581-582-583-584-585-586-587-588-589-590-591-592-593-594-595-596-597-598-599-600-601-602-603-604-605-606-607-608-609-610-611-612-613-614-615-616-617-618-619-620-621-622-623-624-625-626-627-628-629-630-631-632-633-634-635-636-637-638-639-640-641-642-643-644-645-646-647-648-649-650-651-652-653-654-655-656-657-658-659-660-661-662-663-664-665-666-667-668-669-670-671-672-673-674-675-676-677-678-679-680-681-682-683-684-685-686-687-688-689-690-691-692-693-694-695-696-697-698-699-700-701-702-703-704-705-706-707-708-709-710-711-712-713-714-715-716-717-718-719-720-721-722-723-724-725-726-727-728-729-730-731-732-733-734-735-736-737-738-739-740-741-742-743-744-745-746-747-748-749-750-751-752-753-754-755-756-757-758-759-760-761-762-763-764-765-766-767-768-769-770-771-772-773-774-775-776-777-778-779-780-781-782-783-784-785-786-787-788-789-790-791-792-793-794-795-796-797-798-799-800-801-802-803-804-805-806-807-808-809-810-811-812-813-814-815-816-817-818-819-820-821-822-823-824-825-826-827-828-829-830-831-832-833-834-835-836-837-838-839-840-841-842-843-844-845-846-847-848-849-850-851-852-853-854-855-856-857-858-859-860-861-862-863-864-865-866-867-868-869-870-871-872-873-874-875-876-877-878-879-880-881-882-883-884-885-886-887-888-889-890-891-892-893-894-895-896-897-898-899-900-901-902-903-904-905-906-907-908-909-910-911-912-913-914-915-916-917-918-919-920-921-922-923-924-925-926-927-928-929-930-931-932-933-934-935-936-937-938-939-940-941-942-943-944-945-946-947-948-949-950-951-952-953-954-955-956-957-958-959-960-961-962-963-964-965-966-967-968-969-970-971-972-973-974-975-976-977-978-979-980-981-982-983-984-985-986-987-988-989-990-991-992-993-994-995-996-997-998-999-1000
```



The first AI model developed was a random shot selection model. This created a setup with three or four equally possible shots for each court zone for ACT-R to choose with its production rules. With no additional conditions in the rules, other than **the court zone**, a shot would be chosen at random from those available. This model acted as a baseline control. It was also the only model used during development and balancing of the simulation and physics engine.



The second model was a heuristic model that used ACT-R **production rules that implemented a simple squash strategy**, which tried to choose **shots that would be directed to an area of the court where the opponent was not present**. For example, if the opponent was deep in the court (i.e. close to the front wall of the court), it would favour a short shot; and if the opponent was on the forehand side, it would favour a backhand shot. Shot selection rules for each zone were implemented using this simple strategy. In real squash, this approach is a good starting point for any human player.



Heuristic Production Rules in LISP

(p take-shot-z22-z23-StHi-OpSh

=goal>

ISA playing-mode

state 2

; model is playing?

?command>

state free

; module free to accept command?

=predictive>

ISA spatial-state

special 5

; shot selection requested?

> intercept-zone-width 1

; ball is wide?

intercept-zone-depth 2

; ball is middle of court?

> op-zone-depth 2

; opponent is short?

==>

+command>

ISA command-packet

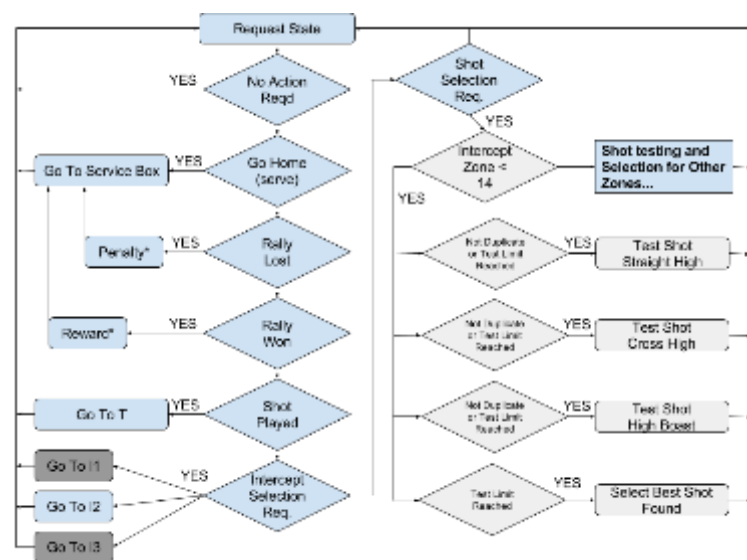
req-cmd 4

; Set Shot

:req-param 51

; Long High Straight

The predictive model went a step further in predicting the outcome of shots the AI model might take. This was done by allowing **the AI model to choose a possible shot before passing that information to the prediction module for simulating and predicting its consequences.** The module would simulate how the shot would play out to predict where the opponent would be when the shot was played, and how much difficulty they would have in then retrieving it and playing a counter shot. The prediction was based on the same strategy as the heuristic model, **trying to find a shot that was as far from the opponent as possible.**



Predictive Production Rule

(p test-shot-z22-z23-StHi

=goal>

ISA playing-mode

state 2

; model is playing?

?command>

state free

; module free to accept command?

=predictive>

ISA spatial-state

special 5

< prediction-count 4

; can test more shots?

- registered-shot 51

; this shot not just tested?

> intercept-zone-width 1

; ball is wide?

intercept-zone-depth 2

; ball is middle of court?

==>

+command>

ISA command-packet

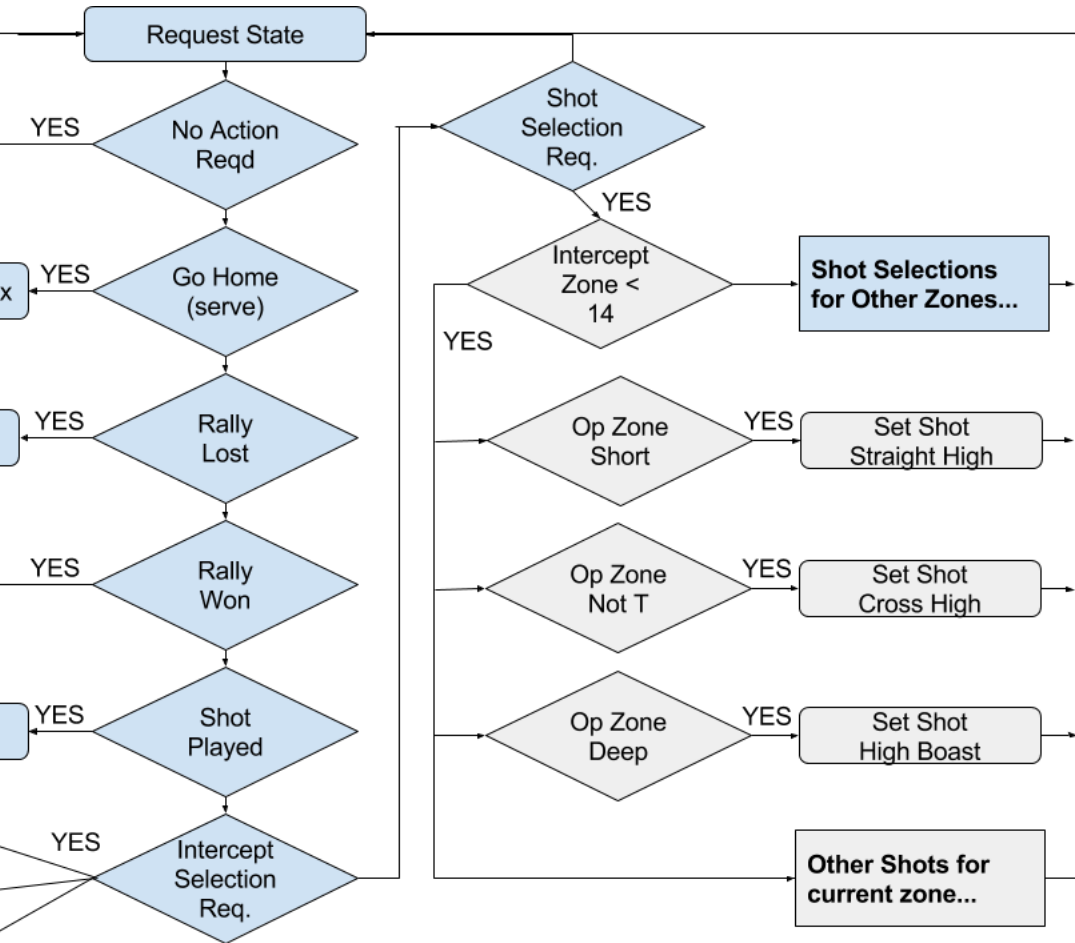
req-cmd 5

; Test Shot

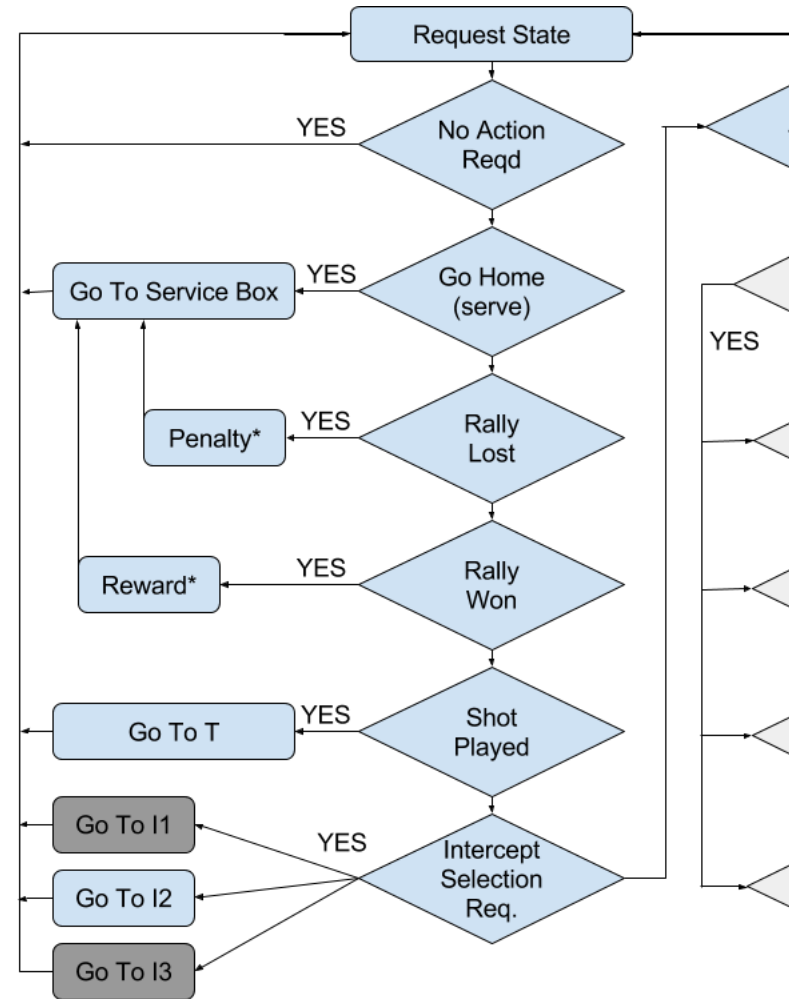
:req-param 51

; Long High Straight

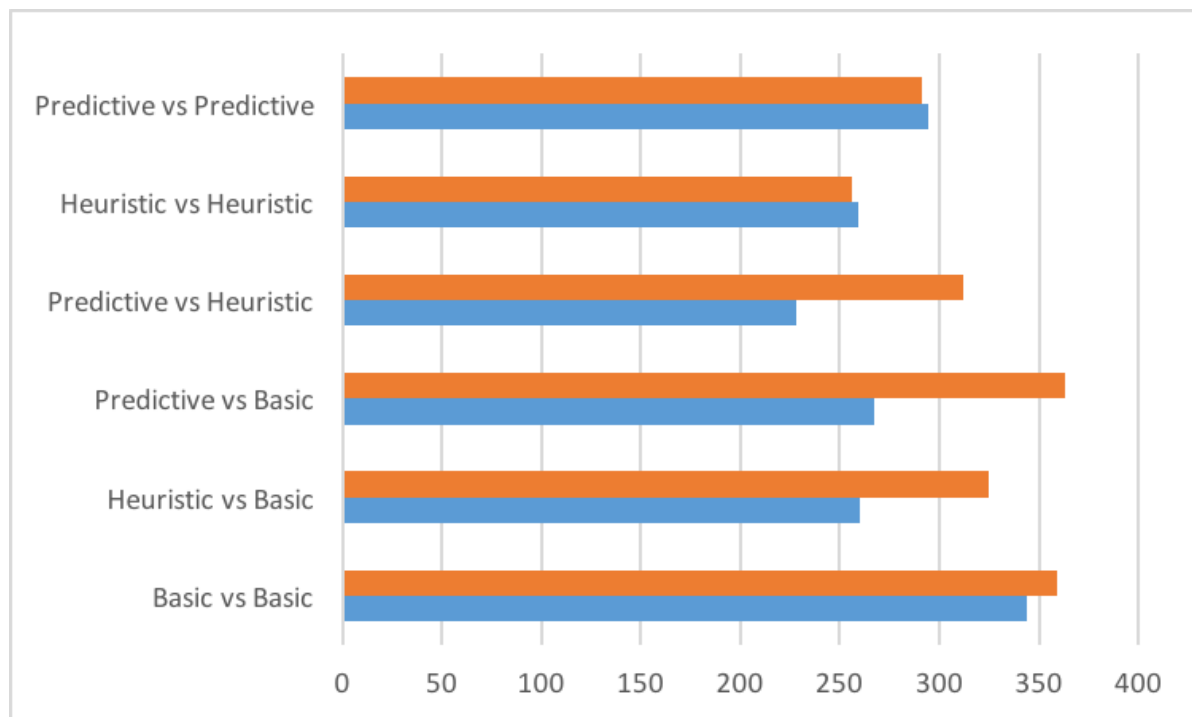
Heuristic



Predictive



Head to head **scores** for all models over six hour duration games.

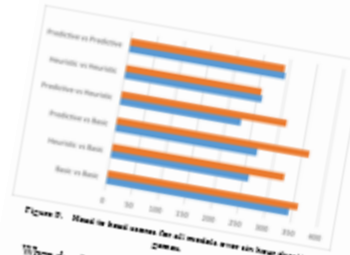


The basic model: 25 rules
The predictive model: 26 rules.

The heuristic model: 45 rules to implement a simple shot selection strategy.

6. CONCLUSION

The results show that, within the limitations of the experiment, a predictive model – with an ability to use simulation to test its own actions to determine and evaluate their possible outcome – held a clear advantage over a model that used heuristics to test relationships between objects in a simulated scenario.



When developing the models, there was a clear advantage to the basic and predictive models over the heuristic model in the reduced number of rules required to implement the shot selection strategy. The basic and predictive models required 25 and 26 rules, respectively. The heuristic model required rules to implement a simple shot selection strategy. The predictive system did have a disadvantage in the time it took to select a shot, it was not always able to complete its shot selection, and in that case it reverted to a random choice. The three models that were developed could all play squash. The heuristic and predictive models both outperformed the basic model. The predictive system also outperformed the heuristic model, despite some limitations in its implementation.

VI. CONCLUSION

The research question asked "How can simulation and prediction improve decision quality in a cognitive architecture?" The answer to this is not straightforward. The results show that, within the limitations of the experiment, a predictive model – with an ability to use simulation to test its own actions to determine and evaluate their possible outcome – held a clear advantage over a model that used heuristics to test relationships between objects in a simulated scenario. It is not, perhaps, surprising that an approach that plunges at the future, however imperfect, would have an advantage over reasoning about a situation based only on where objects are, how they were moving, etc., in the moment. The results of the investigation indicated that prediction provided a more effective appraisal of the value of an action, without requiring detailed rules. There is a caveat here though: the evaluation of the heuristic model was an evaluation of its specific rule set, and it could have been developed further. Its rule set was not very complicated, and it is entirely possible that with a larger rule set and more detailed situational knowledge, it could have outperformed the predictive model. Indeed, both the heuristic and predictive models could have been developed further, to improve each other in a virtual arena race. However, there was another aspect to the modelling. The predictive model only required 26 rules versus the 45 rules of

the heuristic model. Not only were there less rules, they were simpler. Each rule simply stated a possible shot to test, and might be used. In comparison, the heuristic rules required an understanding of squash strategy, and each rule had to be carefully considered as to how it would play out. While both models could have been extended, the effort required to do so would have been considerably different. The heuristic model would require a lot of expert knowledge. The predictive model would have required only finite, some design issues and, perhaps, increasing the fidelity of the predictions. Of course, increasing the fidelity of the simulation engine that can predict outcomes of actions, however imperfectly. Developing the simulation does not require expert knowledge of squash either, but it does require being able to model the physics of the scenario. This is not an insurmountable task and, even in the simple scenario used in this research, more time was spent developing the simulation than was required for the creation of the AI rule set.

VII. FUTURE WORK

The research described above only looked at a highly discrete problem, and the solution was very domain specific. The PACT-R cognitive model gave a more descriptive and methodology of creating a custom model and simulation for every scenario is time consuming, and it would be desirable to accelerate the process by finding a more generic way of describing physical relationships and actions within an environment. It is unlikely that any solution could be truly generic. Such a solution would have to be able to model and simulate a large and arbitrary amount of the real world. Rather, a general generic framework that could be extended and adapted for specific scenarios.

Another area of ongoing research is to use PACT-R in physical robotics. PACT-R is intended for robotics and the considerable challenge of perceiving and simulating at least a small part of the real world. For constrained situations this might not be so difficult. For example, in real-world squash, if you can detect and track the ball, it is then relatively easy to predict where it will go in the rectangular room that squash is played in. The bigger challenge would be predicting the outcome of shots, since this is not as clear-cut in the real world as it was in the simulation, since the simulated shots were simplified, and the virtual robot would be able to play them more accurately than any real robot would be able to. The research also highlighted some issues when working with ACT-R that could be an interesting topic of future work. This task: What alternative learning mechanisms did not work for the decision process? Could some form of tagging (marking low rules in given to the correct rules? How would the modelling need to change to make use of learning? In modelling within ACT-R, rules are tested with a basic set of comparative operators (>, <, =, etc.) While this is



UNIVERSITY of TASMANIA



Ongoing research is to use PACT-R in physical robotics. PACT-R is intended for robotics and embodied AI. Taking this system into the real world presents the considerable challenge of perceiving and simulating at least a small part of the real world.

Also to use PACT-R in combination with machine learning.

the heuristic model. Not only were there less rules, they were simpler. Each rule simply stated a possible shot to test, and required no expert knowledge of how, or when, that shot might be used. In comparison, the heuristic rules required an understanding of squash strategy, not each rule had to be carefully considered as to how it would play out.

While both models could have been extended, the effort required to do so would have been considerably different. The heuristic model would require a lot of expert knowledge. The predictive model would have required only fixing some design issues and, perhaps, increasing the fidelity of the simulation. Of course, the predictive model does require a simulation engine that can predict outcomes of actions, and requires expert knowledge of squash itself, but it does require being able to model the physics of the scenario. This is not an inconsiderable task and, even in the simple scenario used in this research, more time was spent developing the simulation than was required for the creation of the AI rule set.

VII. FUTURE WORK

The research described above only looked at a highly discrete problem, and the solution was very domain-specific. The PACT-R cognitive model gave a some domain-specific methodology of creating a custom model and simulating it every scenario is time consuming, and it would be desirable to describe the process by finding a more generic way of deriving physical relationships and actions within an environment.

It is unlikely that any solution could be truly generic. Such a solution would have to be able to model and simulate a large and arbitrary amount of the real world. Rather, a practical improved implementation of PACT-R would provide a generic framework that could be extended and adapted for specific scenarios.

Another area of ongoing research is to use PACT-R in embodied AI. Taking this system into the real world presents the considerable challenge of perceiving and simulating at least a small part of the real world. For construction situations this might not be so difficult. For example, in real-world situations, if you can detect and track the ball, it is then relatively easy to predict where it will go in the rectangular room that squash is played in. The biggest challenge would be predicting the outcome of shots, since this is not as clear-cut in the real world as it was in the simulation, since the simulated shots were simplified, and the virtual robot was able to play them more accurately than my real robot would be able to.

The research also highlighted some issues when working with ACT-R that could be an interesting topic of future work. ACT-R's reinforcement learning mechanisms did not work for this task. What alternative learning mechanisms did not work for been used? Could some form of tagging (marking key rules in the decision process) be used so that rewards and penalties are given to the correct rule? How would the modeling need to change to make use of learning?

In modeling within ACT-R, rules are tested with a basic set of comparative operators ($>$, $<$, $=$, etc.) While this is suitable for a lot of modeling, when implementing the squash strategy it would have been convenient to have been able to do cool comparisons: hot servers were possible. The matching would bias the rule selection, rather than simply excluding or including specific rules. Getting ACT-R's a fuzzy logic matching system would allow it to work better in situations where there is not a simple black or white answer.

ACT-R also has a declarative memory system (long term memory). This was not used in the research, since it supports a different learning mechanism that did not fit with modeling activation, where recently used memories are more likely to be recalled, and memories that share similar context are also more likely to be recalled (this is the spreading activation). Recently recalled, or similar, memories do not apply to squash, since all shots and outcomes need to be considered equally. However, without the learning, declarative memory could have played a role in the rules in encoding combinations of zones and shots. It was not done this way, since when the decision was made to implement the model as explicit rules, reinforcement learning was still in consideration as a mechanism for improving shot selection.

If declarative memory had been used, how could it have been used, and what sort of learning mechanisms could have been applied? Could there be negative and positive memories, a sort of "positive memories" that are easily recalled, and "negative memories" that are easily suppressed? These considerations may be crucial for applying simulation-based prediction in different robotic applications.



UNIVERSITY of TASMANIA



THANKS!

davidp12@utas.edu.au