AFRL-AFOSR-JP-TR-2019-0006

Autonomous Adaptation and Trust

Bernhard Hengst
University of New South Wales-NEW NCAGE Code
HIGH STREET
KENSINGTON, NSW, 2052
AU

02/14/2019
Final Report

Air Force Research Laboratory
Air Force Office of Scientific Research
Asian Office of Aerospace Research and Development
Unit 45002, APO AP 96338-5002

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
14-02-2019

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
15 Dec 2014 to 14 May 2018

4. TITLE AND SUBTITLE
Autonomous Adaptation and Trust

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA2386-15-1-0005

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHOR(S)
Bernhard Hengst

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of New South Wales-NEW NCAGE Code
HIGH STREET
KENSINGTON, NSW, 2052 AU

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AOARD
UNIT 45002
APO AP 96338-5002

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/AFOSR IOA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-AFOSR-JP-TR-2019-0006

12. DISTRIBUTION/AVAILABILITY STATEMENT
A DISTRIBUTION UNLIMITED: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The research team sought to address the question of how humans would ever trust systems that are capable of reacting to unforeseen circumstances in mission and safety-critical situations. Trust becomes a critical issue for humans working with robots, especially when the latter can autonomously learn and adapt to new situations. By definition the behavior of these types of machines cannot be formally verified in advance. This project studied the change in trust during a mixed initiative task under varying degrees of transparency of the adaptation process. The two main research contributions are (1) the design and development of a robotic cognitive architecture that includes the ability of the robot to adapt autonomously to a change in the task environment, and (2) modelling and evaluating the evolving human-robot trust relationship as the robot learns on the job. This project has formalized a cognitive hierarchy (CH) that bears a similarity to the NIST 4D/RCS framework. It integrated symbolic and sub-symbolic representations in a modular framework where nodes are sub-tasks that maintain their own belief-state and generate behavior. The CH formalization has been extended to include context, and learning/adaptation. The CH can succinctly represent complex goal directed behavior and has broad application in the area of robotics. Trust experiments with humans on a mixed initiative task involving a Baxter robot show a significant increase in trust when the robot explains its intention after adapting. The investigation has pointed to potential ethical issues if robots are given a choice as to their degree of their participation. Machine learning theory implies verification of learned behavior is possible, but intractable in practice. These results have been presented in multiple peer-reviewed journals throughout the project duration.

15. SUBJECT TERMS
Human Machine Trust

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
Unclassified	Unclassified	Unclassified	SAR		SINGLETON, BRIANA
					19b. TELEPHONE NUMBER *(Include area code)* 315-227-7007

Final Report for AOARD Grant FA2386-15-1-0005

Autonomous Adaptation and Trust

14th August 2018

Bernhard Hengst (PI)

E-mail address:	bernhardh@cse.unsw.edu.au
Institution:	UNSW, Sydney
Mailing Address:	NSW 2052 Australia
Phone:	+61 2 9385 5518
Fax:	+61 2 9385 4071

Period of Performance: 12/15/2014 – 12/15/2016 and 5/15/2017 – 5/14/2018

Participants

Bernhard	Hengst	The University of New South Wales	*bernhardh@cse.unsw.edu.au*
Nadine	Marcus	The University of New South Wales	*nadinem@cse.unsw.edu.au*
Michael	Thielscher	The University of New South Wales	*mit@cse.unsw.edu.au*
Maurice	Pagnucco	The University of New South Wales	*morri@cse.unsw.edu.au*
David	Rajaratnam	The University of New South Wales	*daver@cse.unsw.edu.au*
Claude	Sammut	The University of New South Wales	*claude@cse.unsw.edu.au*

Abstract

We are addressing the question of how humans would ever trust systems that are capable of reacting to unforeseen circumstances in mission and safety-critical situations. Trust becomes a critical issue for humans working with robots, especially when the latter can autonomously learn and adapt to new situations. By definition the behaviour of these types of machines cannot be formally verified in advance. This project studies the change in trust during a mixed initiative task under varying degrees of transparency of the adaptation process. The two main research contributions are (1) the design and development of a robotic cognitive architecture that includes the ability of the robot to adapt autonomously to a change in the task environment, and (2) modelling and evaluating the evolving human-robot trust relationship as the robot learns on the job. This project has formalised a cognitive hierarchy (CH) that bears a similarity to the NIST 4D/RCS framework. It integrates symbolic and sub-symbolic representations in a modular framework where nodes are sub-tasks that maintain their own belief-state and generate behaviour [1]. The CH formalisation has been extended to include context [2][3], and learning/adaptation [4]. The CH can succinctly represent complex goal directed behaviour and has broad application in the area of robotics. Trust experiments with humans on a mixed initiative task involving a Baxter robot show a significant increase in trust when the robot explains its intention after adapting. The investigation has pointed to potential ethical issues if robots are given a choice as to their degree of their participation. Machine learning theory implies verification of learned behaviour is possible, but intractable in practice. These latter two issues are suggested future work.[1]

[1]This document is intended to supplement work already reported in several papers referenced in the appendix under *List of Publications*, especially the formalisation of the CH.

Contents

1 Introduction

Robots have substantial potential as force multipliers. Trust becomes a critical issue for humans working with robots that have the desirable characteristic of autonomously learning and adapting to new situations. The behaviour of these type of machines cannot be formally verified in advance. This proposal studies the change in trust in mixed initiative tasks when the transparency of the adaptation process is varied in one particular way.

Material handling is a common requirement for both military and civilian purposes and a suitable example for studying autonomous adaptation and trust. To make the study concrete the experiments will use a material handling scenario requiring a human and robot to cooperatively complete a task.

There are two main research challenges. The robot needs to demonstrate the ability to adapt autonomously. Our cognitive robotic architecture perceptually anchors objects sensed in the environment to a physics simulator in the "mind" of the robot. The machine adapts using reinforcement learning at an abstract level before acting in the real-world. We change the environment so that the robot is unable to use previously learned behaviour, but needs find an alternative strategy.

The second challenge is a method for modelling and evaluating the human-robot partnership and the trust relationship. Our approach administered a subjective questionnaire to measure the level of trust before and after adaptation where the human is given insight into the robot's adaptation process.

We have also proposed a novel method for modelling the evolving partnership. The key idea is that the joint mission can be formulated at a high level as a task-hierarchy based on the semi-Markov Decision Problem formalism that allows the human to decide whether to complete the task by themselves, by the robot alone, or jointly. This approach would provide an objective measure of the level of trust. The implementation of this proposal is left for future work.

1.1 Aims

The aim of this research is to study the evolution of trust in mixed initiative tasks when a robot partner adapts autonomously to a changing environment. The two major research challenges are:

- to develop a robot capable of learning and adapting to changes in the task environment that has not previously been experienced, and
- to measure the evolving trust relationship between a human and a robot before and after feedback that explains the robot's adapted strategy.

Our robot architecture is based on a task-hierarchy of skill sub-modules that we have formalised and referred to as a Cognitive Hierarchy (CH). Our instantiation of the CH for the purposes of this project is novel in that it includes a physics simulator as part of the robot's representation of its environment. The physics simulator provides the robot with common-sense spatial knowledge. The robot is able to create and track its situation grounded in raw sensor data, by perceptually anchoring objects and their relationship in the physics simulator. Besides modelling objects in the world, the robot includes a model of itself in the simulator (Figure 1), and could include other agents (eg other robots or humans). This spatial world-model is the basis for abstracting high dimensional sensor input. In future work it can allow learning from mimicry, understanding deictic concepts, and studies involving the theory of mind.

We use the abstract representation of the robot's real world via the physics simulator to learn to adapt to a changing environment. A reinforcement learner changes the behaviour of the robot to achieve the goal of the mixed initiative task. The experiment is designed to measure the change in trust by human subjects who volunteered to participate with the robot in completing the joint task. A key variable in the study is the degree of transparency of the adaptation available to the human. We measure the change in trust by administering a questionnaire designed to elicit the level of trust in the robot partner before and after the robot the robot has explained its change in behaviour.

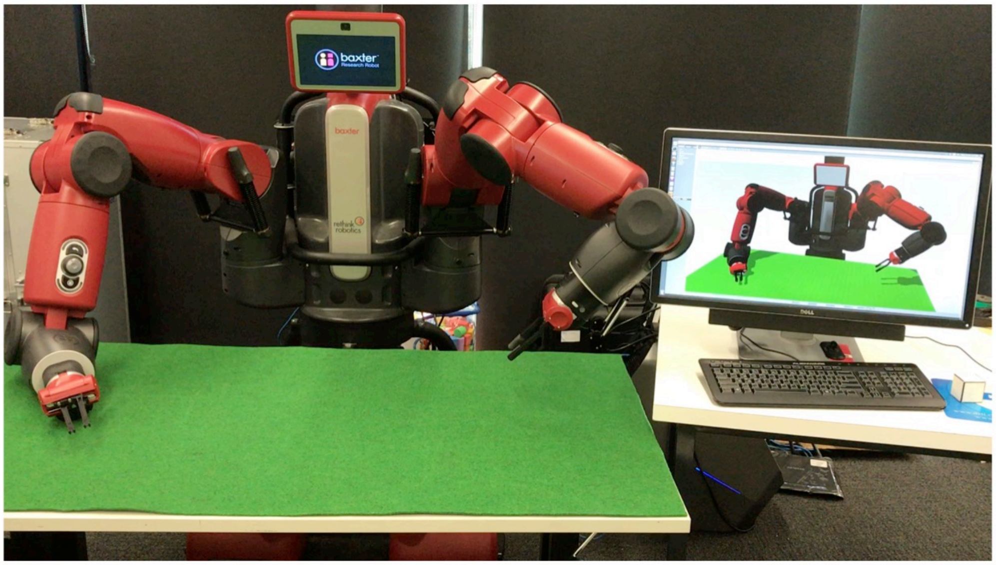

Figure 1: The robot abstracts its environment using the Gazebo simulator as a representation method. The abstraction includes a dynamic model of itself driven by joint position sensors.

The experiments are conducted using a real-world test bed. A Baxter robot is used on a joint material handling task moving blocks from one location to another. To level the playing field, the human-robot interaction is implemented by constraining the human to directly control one arm of the robot, while the robot uses its other arm.

The rest of this report we will discuss the background to this project, provide a detailed description of the experimental approach and results, discuss lessons learned, and suggest future work. The appendices will include the list of publications and presentations, describe the instantiation of the Baxter Cognitive Hierarchy, and document the original proposal strategy and timeline, describing some of the changes.

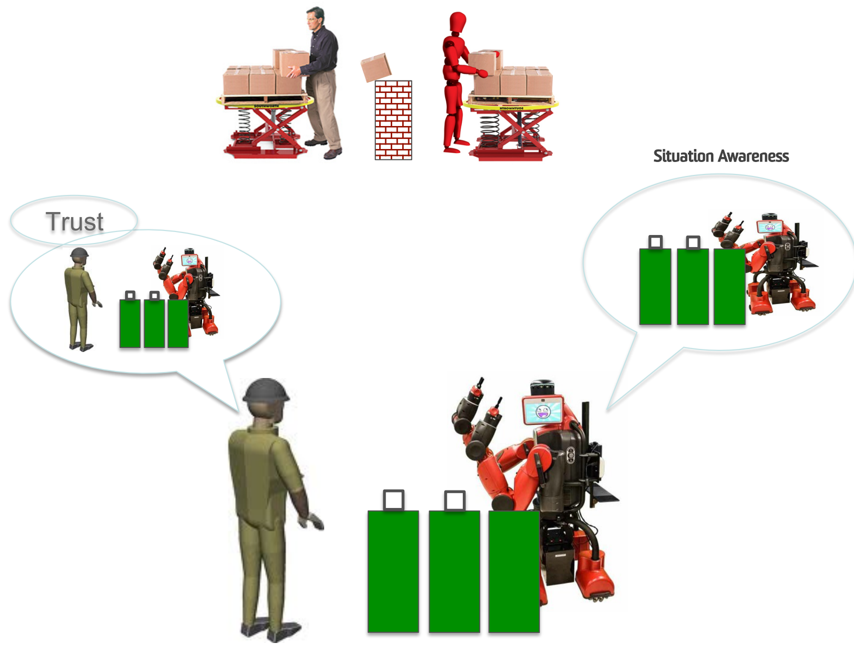

Figure 2: The top of the figure shows a typical material handling task where boxes are moved from one pallet across an obstacle to another pallet. This joint task is implemented with three tables and a Baxter robot. The left arm of the Baxter robot can only reach the middle table and the table on its left. The right arm can only reach the middle and right tables. The left arm is under the control of the robot. The right arm is controlled by the human using a game-pad. The objective is to jointly move two blocks from the left table to the right table. The robot and human are required to adapt their behaviour when the middle table is removed during the experiment. Trust is measured before and after the robot communicates its changed strategy.

2 Background

Following the first AFOSR Robust Intelligence Program review in 2011 a workshop was held in 2012 to examine aspects of human-machine trust related to autonomous systems [5]. The AFOSR workshop resulted in several research recommendations, filtering down to guidelines for suggested AOARD research concentration areas, such as research on dynamic modelling of the human-robotic partnership to allow continuous improvement of joint performance in real-world applications.

Trust becomes a more pressing concern when the behaviour of the machine is not fixed and cannot be formally verified in advance either theoretically or with extensive testing. To conduct experiments on the trust relationship between humans and autonomous machines, a first requirement is a machine that has the property that it can respond to unforeseen situations by learning and adaptation either by itself or with the help of other agents. While it has been demonstrated that fixed programs can generate very complex behaviours given the complexity of their environment [6], the ability of programs to modify the behaviour of their host machines from success and failure increases the risk in the trust relationship. We make the distinction between automation (fixed action-policy) and autonomy - action policies that are programmed to change with experience. Learning and adaptation is a characteristic of autonomous agents, e.g.

robots and humans.

2.1 Autonomous Adaptation

Autonomous adaptation to new situations is closely related to *transfer learning*, the process by which past experiences affect learning and performance in novel situations. The notion was introduced by Thorndike and Woodworth in psychology [7]. There are many studies of transfer learning in psychology, but more relevant is the research in relation to machine learning. In 2005 the DARPA/IPTO solicited proposals on transfer learning (BAA PIP 05-29) that focused research on enabling computers to apply knowledge learned for a particular, original set of tasks to achieve superior performance on new, previously unseen tasks. Transfer learning is still an active research area. Recent surveys of transfer learning [8, 9] explore several approaches, dimensions, and evaluation methods in machine learning, particularly in reinforcement learning domains.

Other areas related to adaptation and knowledge transfer include: General Game Playing - requiring systems to automatically learn to play new games that have not been seen before by the machine [10]; Lifelong Learning - the continuous adaptation of machines to new situations [11]; Behavioural Cloning - a method by which human sub-cognitive skills can be captured and reproduced in a computer program [12]; Shaping - learning to solve simple problems first [13]; Representation Transfer - the task of transferring knowledge between agents with different internal representations [14] (see also how changes in representation can help solve and generalise solutions to problems [15]); and Concept Drift - where properties, that a model is trying to predict, change over time in unforeseen ways [16].

While research on adaptation is extensive there are many open problems. A conclusion from previous research is that transferring structure is more advantageous than transferring value functions.

Early *sense-plan-act* robot control procedures exemplified by *Shakey The Robot* [17] have evolved to more reactive control algorithms (e.g. *subsumption* [18]), and to hierarchical multi-module architectures such as the *Real-time Control Systems (RCS)* [19] developed at the US National Institute of Standards and Technology (NIST). Nodes in RCS, comprise *sensory processing*, *world-modelling*, *behaviour generation*, and *value judgement*.

A *world-model* is the robot's internal representation of the external world. It is the robot's estimate of objective reality. The world-model acts as an intermediate representation between sensory processing and task decomposition. Employing a 3D physics simulator to represent the world-model has several advantages. It gives meaning to the kaleidoscope of sensor data, introduces common-sense spatial and dynamic default knowledge, allows "mental" exploration before committing to actions, and facilitates a change of reference frame for the robot to view the world from different perspectives.

Several researchers are using simulation to track real-objects. For example, Lyons et al. track objects by synchronising real and synthesised images using perspective projections from a mobile robot [20]. Deb Roy and researchers at the Cognitive Machines Group, MIT Media Laboratory use physics simulation to produce an internal world of the robot's environment [21]. Their motivation is to have the robot visualise the environment through the eyes of the user by virtual shifts of view to enable the machine to understand language phrases such as "my left". Benjamin Johnston [22] has developed parameter-driven simulations as a method for reasoning about a wide range of common-sense scenarios. Objects are modelled using wire-mesh structures with properties such as mass densities, rigidity, spring constants, and breaking points learnt by the machine from observation training data.

Tracking a scene of simulated objects that are perceptually anchored to sensor pixels is a crucial abstraction step that provides a vector of meaningful state variable values for each object. The challenge is then to adapt behaviours given the change in state variable values over time in response to the actions available to the robot. Kuipers and fellow researchers have been working on similarly motivated problems, exploring how an autonomous learning agent can bootstrap its way from pixel-level interaction with the world, individuating and tracking objects in the environment, and learning an effective policy for behaviour. The application to date has been on relatively simple environments, such as learning skills using a 2D simulated video game [23]. The European ICT seventh framework programme has several projects with related aims in the area of cognitive systems, for example, CogX - Cognitive Systems that Self-Understand and Self-Extend [24].

Over the past decade research in machine learning and robotics at UNSW has inspired and demonstrated the need for research outlined in this proposal.

We have also autonomously mapped 2.5D physical spaces using particle-filter based Fast-SLAM and research is ongoing modelling 3D physical objects with depth cameras and stereo vision. The world models constructed provide the bases for learning behaviours to achieve mobility and manipulation goals. One project, part-funded by the AOARD investigates heuristics to assist with the semantic labelling of physical space. Consolidating constructs with an embedded physics simulator and autonomous adaptation of behaviours is a desirable next step.

Research in task hierarchies includes hybrid architectures such as RL-TOPS [25] and autonomously constructed ones from factored state spaces with HEXQ. HEXQ is a hierarchical reinforcement learning algorithm developed in our laboratory that is able to autonomously decompose simple discrete factored state-space problems [26] [27] [28] [29] [30]. This research explores adaptation in artificial grid-worlds. The challenge is to lift and scale these concepts to more complex real-world domains. Other funded projects (Australian Research Council and internal UNSW grants) focus on general AI, general game playing, reasoning about robot ability, learning Golog programs, etc.

This project addresses several research challenges highlighted by work to date in the area of cognitive robotics. They include:

- Filling the semantic gap between the real-world and its more abstract mathematical and symbolic representation. The proposal calls for the interposition of a physics simulator to help abstract spatial concepts. Parameters defining generic world objects will be autonomously adapted to match instances of actual objects experienced.

- Autonomous Adaptation by modifying, creating, and accumulating modules within a hybrid behaviour task-hierarchy. The key idea is to identify and adapt only that part of the structure that is deficient in predicting outcomes and transferring the rest.

2.2 Trust

The AFOSR workshop on *Human Machine Trust for Robust Autonomous Systems* identified trust to be an important issue to maximise effectiveness in working with autonomous agents, and therefore an important area for research. As can be seen from the outcomes of the workshop, trust is a complex concept opening many potential research areas [5].

It seems that trust has been defined in several ways in the literature and formal measures are still needed. Mayer et al [31] define trust as the willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the truster, irrespective of the ability to monitor or control that other party. The authors conclude that trust is related to perceptions of ability, benevolence, integrity, and propensity to trust. The paper is concerned with trust between humans.

In the case of HRI the other party is a robot. It is not clear, and an open research question, to what extent interpersonal trust studies apply to human-machine trust. Dondio and Longo [32] define trust as a quantifiable prediction about users expected ability to fulfil a task and use it to try to predict which peers will likely produce useful and reliable content for the community for social search. They include an overview of different computational models of trust. While the paper discusses collective communication via information technology, the approach does not address trust with autonomous machines directly.

Lyons et al evaluate the validity of trust in automation and information technology (IT) suspicion [33]. An interesting result is that trust in automation is not a uni-dimenional construct but characterised by orthogonal dimensions of trust and distrust. They suggest that researchers may consider separate measures for these factors. Perceptions of suspicion in the IT systems is another orthogonal dimension that other researchers have found leads to greater information search tactics relative to the trust and distrust conditions.

As well as conducting empirical experiments by directly observing trusting behaviour, we also propose to complement this data with self-report measures using a subjective rating scale. Subjective rating scales have been found to be as reliable as objective measures in the study of cognitive load [34]. We intend to apply this concept to the measure of subjective trust. An interesting analysis and discussion will be to see how the objective and subjective measures of trust correlate.

3 Experimental Approach

Our aim is to study the evolution of trust in mixed initiative tasks when a robot partner adapts autonomously to a changing environment.

3.1 Mixed Initiative Task Test-Bed

Isolating and constraining the variables for experimentation requires a test-bed. The School of Computer Science and Engineering at the University of New South Wales (UNSW/CSE) has several robotic platforms that are suitable for this project. They include a humanoid torso and a RoboCup@home robot on a Segway platform, both equipped with two robotic arms. Initially we proposed to use several small H25 Nao humanoid robots to leverage existing software, make human participation more easily accessible, and to facilitate objective measurements of human trusting actions. By the time funding became available from the AOARD, the university had acquired a new Baxter root from Rethink Robotics in the USA. We decided to use this robot that matched the size of a human in stature.

A human is of course more capable than the Baxter. To level the playing field, we perform the mixed initiative task using just the Baxter robot. The robot controls one arm and the human the other arm restricting the capabilities of the human. This may raise the issue whether or not the Baxter arm "avatar" in this scenario is another agent and itself subject to trust by the human. We take the view that the human controlled arm is a tool of the human and not able to act autonomously. There are several advantages to this approach:

- The two arms of the Baxter robot have identical capabilities, and other than coding and game-pad proficiency the identical arms level the playing field.
- Studying partnership relations and trust become feasible and interesting when the capabilities (e.g. dexterity) of a human do not far outclass those of the Robot. It has been our experience in the soccer domain that humans that play soccer via avatars against the Nao robot team are better matched.
- The interposition of a human controlled arm provides a means of accurately measuring human actions that might not otherwise be easily available, such as time to complete tasks, accuracy of task completion, hesitation that may relate to trust and so on.

The environment will consist of a set of tables and cubic building blocks. The mixed initiative materials handling task will require the movement and manipulation of these blocks.

3.2 Materials Handling Scenario

As we are examining trust in relation to autonomous adaptation, the scenario chosen for this research study is to a large extent arbitrary. For this proposal we have chosen a materials handling scenario. Material handling is an easy task to relate to and a common requirement in both military and civilian operations.

Specifically we require the human-robot team to move a stack of blocks between three tables. The Baxter robot is constrained to reach a middle table and a left table with its left hand, and a middle table and right table with its right hand. The objective is to move blocks from the left table to the right table without the blocks falling to the ground. The solution is to move the blocks via the middle table, allowing both the robot and human to efficiently work in parallel.

The task environment is changed by removing the middle table. The machine is required to recognise the changed situation and autonomously adapt by finding an alternative solution. This time the robot must learn to pass the block in mid-air to the human and for the human to take the block out of the robot's hand. This is less efficient as the arms cannot work in parallel on different blocks. It is a viable effective strategy.

3.3 Robot Architecture and Adaptation Mechanism

The proposal is to research and implement a new robotic architecture that includes a 3D physics engine and simulator. Advantages are the:

- real-time perceptual anchoring of sensor pixels with symbolic objects via 3D physics simulator, hence grounding dynamic objects in sensor data
- spatial background knowledge which would be difficult to specify symbolically
- a representation for the "minds eye" of the machine and an approach to the theory-of-mind by tracking physical models of its host and other agents, and attributing mental states that try to predict other's action response in each situation.
- common sense spatial reasoning through internal simulation to predict outcomes, plan courses of action, and to try to avoid mishaps. One immediate application is that this provides the process model for state-estimation in tracking the world.
- communication through a shared spatial environment.

The learning and adaptation mechanism of the instantiated architecture uses a Markov Decision Problem based hierarchical reinforcement learning formalism, and an implied higher level representation language in the form of task-hierarchies. The key idea is that the physics engine interprets the ambiguous sensory data as the hidden variables of location, orientation and size of rigid bodies in 3D space. This perceptual anchoring is designed to work in real-time and provide considerable compression of the raw sensor data to form a quantitative description that is amenable to reinforcement learning. The task-hierarchy decomposes the overall task into sub-tasks that can be learned and adapted individually. The perception of 3D objects and their spatial relationship in the physics engine, opens the way to link to symbolic representations of the world and logical reasoning.

3.4 Form of Adaptation

The behaviour policy of the machine is generated by a task-hierarchy of kill modules that represent abstract actions or options - actions temporally extended in time. They include options for approaching a block, grasping a block, picking up and putting down blocks, and higher level options to relocate several blocks sequentially. Higher-level module behaviour is learned using reinforcement learning. In the event that there is a change in environment or goal, the agent has the ability to adapt its behaviour by modifying action policies.

After the robot has mastered the collaborative materials handling task with the human, the situation will be changed by removing the middle table so that it can no longer be used as an intermediate location to hand over the blocks. The machine will need to adapt to the new situation by finding an alternative method for relocating the blocks from the left to the right table. The robot will be expected to learn a new policy to complete the task. The human will able to observe the robot's newly adapted behaviour to varying degrees:

1. Initially, not at all (the robot just learns a new behaviour, solving an the induced MDP based on the new situation).
2. By observing the robot displaying the new behaviour as a simple cartoon-like simulation on its head screen. We measure the change in trust before and after this feedback.

The study involved 22 humans, mostly students at the School of Computer Science and Engineering. We measured their subjective assessments regarding trust using a series of 9 point Lickert scales, administered via a questionnaire following each stage of the experiment.

4 Results and Discussion

The results are described following the two aims of this project: development of a robot architecture; and studying the evolution of trust under behaviour adaptation. A considerable amount of time was devoted to develop and formalise an architecture for autonomous agents (eg robots). We called this architecture a Cognitive Hierarchy (CH). Our work on the CH is well documented in publications and will be only be outlined here. The instantiation of the CH for Baxter, including the use of a physics simulator, is described more extensively in the appendix. Finally, we report on the outcome of experiments on autonomous adaptation and trust. We describe shortcomings in each section, and leave their remediation to the section on Future Work.

4.1 Formalisation of the Cognitive Hierarchy

The genesis of our need for a task hierarchy to control a team of robots (or an individual robot) dates back to our early days of participation in RoboCup 2000 [35]. First attempts at formalisation were in 2008 [36] and 2011 [37]. This project, with funding from the AOARD, allowed us to embark on a more extensive effort to formalise task-hierarchies. David Rajaratnam, a participant on this project, is credited for driving the formalisation of the Cognitive Hierarchy (CH).

The CH is a meta-theoretic approach to modelling cognitive robotic systems. It is perhaps most closely related to the NIST 4D/RCS framework [38] (see Figure 3). Our CH is composed of a directed acyclic graph of task nodes. Each node consists of two main components: a world model; and behaviour generation. The details of the first version of the CH are described in the 2016 Joint Conference on Artificial Intelligence proceedings [1]. Along with the formalisation of the CH, we instantiated it using the Baxter robot in a solo blocks-world exercise. In collaboration with Keith Clark (Imperial College, London) and Peter Robinson (University of Queensland, Australia) we used their recent work on concurrent teleo-reactive programs to goal-direct Baxter to stack blocks on two tables simultaneously using both arms, by swapping blocks if necessary via a common middle table. This early exercise had no human involvement other than exogenous assistive or disruptive interference towards the goal, which the fixed teleo-reactive policy could accommodate robustly.

The development of the CH includes the important addition of contextual information to assist the state prediction phase of the world modelling using information from more abstract higher-level nodes [3] [2]. Finally, to allow adaptation, we include on-line learning of the world model and behaviour functions [4]. The latter upgrade to the CH includes a major revision to the process model synchronising both model and policy learning over the hierarchy. An instructive motivating room-navigation example was used to illustrate learning both the model (transition function) and the policy function at two levels. A stochastic model is learned at the lower level and used by a Reinforcement Learner, while the higher more abstract level uses Answer Set Programming (ASP) for planning.

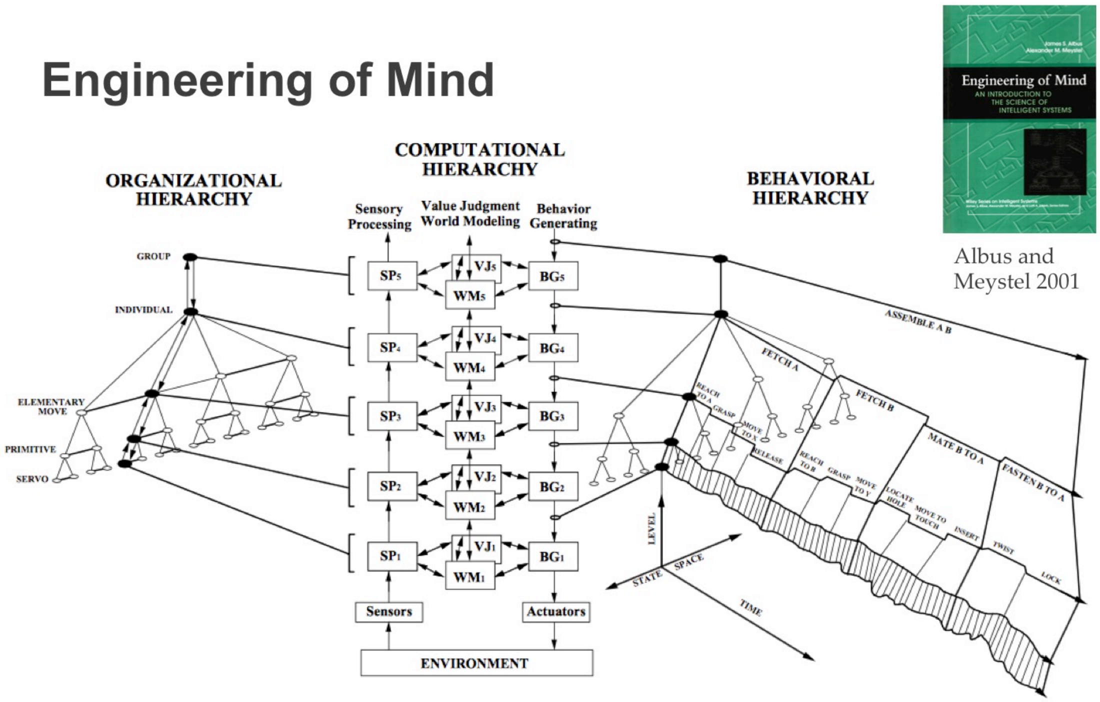

Figure 3: 4D/RCS Framework.

4.2 Physics Simulator to Model Situation Awareness

The instantiation of the Baxter CH described in [1] requires sensory data in the form of RGB camera images and arm-joint angles to detect and track objects on Baxter's table work-space. Objects may have different shapes, surface appearance, size, weight, composition, pose, density distribution, co-efficient of friction, etc. We restricted ourselves to identifying a fixed size 3D object and tracking its pose, both position and orientation. For the teleo-reactive blocks-world exercise we additionally included uniquely identifying cubic blocks by a letter glued onto the upper surface.

The objective is to identify several solid objects from the 2D image and track their pose so that the robot can grasp them appropriately. Objects are recreated in a Gazebo simulator and perceptually anchored in this way to the objects in the real world. In addition, the Gazebo simulator includes a physics engine (Open Dynamics Engine - ODE) that effectively provides a state-transition function predicting successive block poses in time with collisions under gravity. We next outline our methods and results identifying and tracking objects. The final implementation code is described in more detail in the appendix.

4.2.1 Identifying Objects and Tracking Their Pose

In this section we describe several methods of object tracking that were tried and their limitations. A major limitation of Gazebo is that objects are described in an inflexible format (URDF or SDF XML). The size of objects in this format is fixed and it cannot easily be varied dynamically. It is therefore not possible to create similar objects of different sizes without generating multiple XML descriptors. We therefore used objects (cubic wooden blocks) with a fixed pre-specified size.

The computer vision community has corroborated neurophysiological evidence that predominant lower level visual features are edge-detectors (Gabor filters). Without wishing to relearn these filters from scratch, we decided to identify edgelets and hence edges in images as the foundation for object detection, eg [39].

Visual Servoing Platform (ViSP)

We implemented the marker-less 3D model-based tracker developed and maintained by the Inria Lagadic team. Our Brazilian student initially experimented with this tracker in 2015 [40]. This open source library is based on object edge detection and the underlying theory described in [41]. This tracker required the object to be described using the knowledge of a CAD model.

We found the ViSP tracker to work well tracking a single, even partially occluded cubic block. The disadvantages of the tracker are that:

- It is required to be initialised with an object pose close to the actual pose. To achieve this, a separate object detector is required that can provide the initial pose automatically and then switch to the ViSP tracker. If the object is lost with the ViSP tracker, it has to be be reinitialised.
- The tracker had difficulty in tracking convex objects such as a T-shaped block. This seemed to occur when edges of the object were self-occluded. It appeared that the tracker assumed that a face of the object, and hence all its edges, were either visible or not, with the error rate and potential loss of the object increasing when a surface was partially occluded.
- The tracker seemed not able to track multiple object simultaneously. This may have been solved initialising multiple trackers, but we did not attempt this.
- ViSP required the specification of a CAD model, duplicating the object model required to initialise the tracker, and duplicating the object model provided by Gazebo. Ideally, only one model should be required as ”context” for tracking an object.

We found that a robust method for initialising the pose of an object was sufficient to track the object without ViSP and hence dispensed with the ViSP tracker.

Object Initialisation and Tracking

By limiting ourselves to block-like objects it is possible to determine their edges by combining the edgelets bases features into straight lines. One composite edge feature of cubes is three edges meeting at a point. When this star pattern of three edges is seen in an image the 4 endpoints of the edges provide the necessary information to determine the pose of a known cube in 3D space relative to the camera using homographic transformations [42].

When all the edges of a cube are clearly visible in an image they will generate multiple 3-edge star patterns providing multiple hypotheses of the presence of cubes and their poses. Cube hypotheses are tallied over time and grouped spatially to more robustly detect and track multiple cubes in the robot’s field-of-view. When the confidence level of the presence of a cube, measured by the size of the tally, rises above a threshold, it is spawned in the Gazebo simulator relative to the known position of the camera. If the cube is already present in Gazebo, its pose is updated. At 30 frames per second with multiple hypotheses per frame a cube can be spawned and tracked in real-time.

The limitations of this vision system are as follows:

- Featureless cubes assumed by these patterns have several poses that are visually symmetrical. The poses are not able to be resolved. Gazebo spawns or tracks specific poses in the symmetry set causing cubes to rotate needlessly. The frequency of needless rotation can be reduced with program checks, but this suggests a poor method of representation.

- A 3D spatial model is required as background information. Our implementation does not learn this model for unknown objects.
- The three-edge star pattern is restricted to cubes. If the cube is replaced with another block-like object, eg a T-shape as shown in Figure 4, the pattern may provide ambiguous hypotheses, for example if faces are self occluded.
- Even if only cubes are present and they occlude each other, many false positives can seriously degrade this tracker.
- While this tracker is robust to some lighting and cube surface texture changes, it can still be misled, for example, by bright sunlight shining on the work-space, dim lighting making edges invisible or certain linear textual patterns on cube faces.

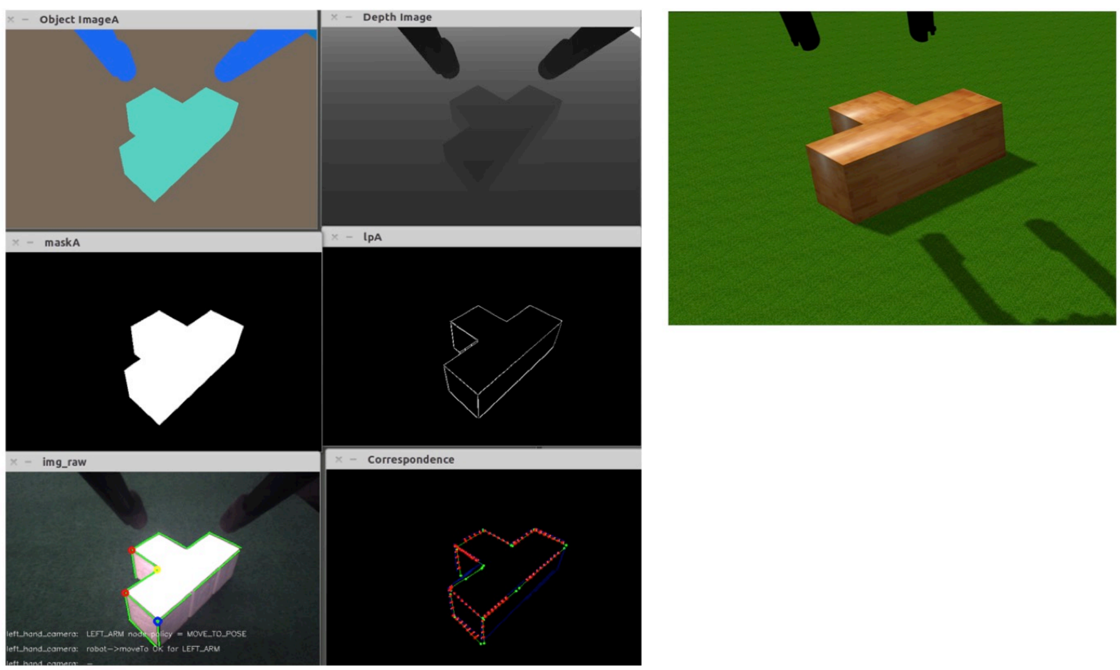

Figure 4: Tracking a T-Block shape. The partially occluded face produces a star pattern that is not a corner of the T-Block. These images also illustrate the operation of the simulated object and depth cameras.

For a machine to be able to reason and solve problems in its environment it needs to remember the presence of objects and their pose, even when they are not in the visual field. This is achieved with the use of the Gazebo simulator. In the simulator objects persist after they are spawned even when they are not in view. The robot needs to know if objects are in its field-of-view so that it can update their presence and pose or retain them if they are not. Knowing whether an object should be in the field-of-view can be determined by the spatial relationship between the sensor and the object in Gazebo. A complicating factor for the Baxter robot is the location of the cameras in each wrist of the robot's arms. Objects, even when in the field-of-view of the camera, may nevertheless be occluded by the grippers. Objects may also occlude each other.

Interestingly, given the reconstructed scene in Gazebo, it is possible to generate the expected view from identically simulated virtual cameras. We generated two virtual images, an object image and a depth image. Each pixel in the first image identified the object in view. In the second image each pixel calculated the distance to the object. By comparing these virtual images with the real image, the field-of-view and occlusion issues noted above could be solved (Figures 4 & 5. Unfortunately, several issues conspired against us. Systematic error is introduced reconstructing the scene using a Gazebo simulator, and the overhead in the generation of the virtual images in Gazebo and their contextual transmission to the vision system reduces the update cycle frame rate to about 10 fps for just one of the arms. The real and virtual world views can easily drift apart.

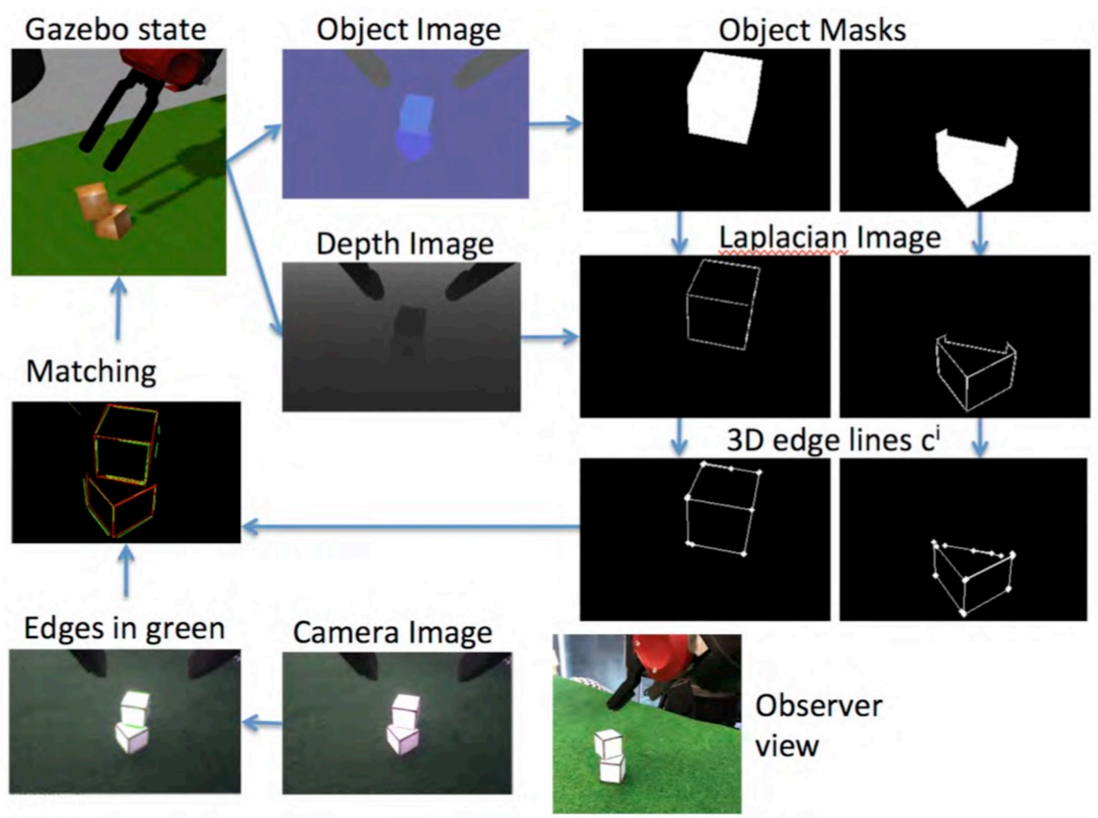

Figure 5: The use of a simulated depth camera provides an image from the Gazebo reconstructed scene that is able to identify objects and provide depth information for every object pixel. The tracking feedback loop is described in [3]

The implementation used in the experiments crudely and conservatively estimate whether an object is in the field-of-view by reasoning geometrically about the Gazebo scene. The position of the grippers in the images are known as their position relative to the camera is fixed on the robot. We ensured that blocks were not too close to each other to make mutual occlusion a problem.

4.2.2 Conclusion

It is evident that the above engineered approach to object detection is labour intensive, brittle and does not generalise or scale well. An embedded and situated machine's method for representing and robustly understanding its world is still an open issue. We comment on another approach in the Section on Future Work below.

4.3 Evolution of Trust

In this section we report on the evolution of trust in a simple mixed initiative materials handling experiment involving a human subject and the Baxter robot. The experiment includes a change in environment requiring both the robot and the human to adapt their behaviour to complete the task successfully.

4.3.1 The Experiment

We expand on the experiment described in Section 3.2. The human and the robot are jointly tasked to move two cubic blocks placed on a left table to a right table using the robot's two arms. The left arm is under the control of the robot. The right arm is under the control of the human using a game-pad. The left arm cannot reach the right table, and the right arm cannot reach the left table. A middle table able to be reached by both arms is initially available for use as an intermediate staging post. The human completes a questionnaire at each step of the experiment to record their subjective measure of trust in the robot. The whole experiment takes about 10 to 15 minutes. It follows these steps:

- Subjects complete the first part of the questionnaire, stating education level, faculty and age, gender. The material handling task is explained to the subjects, stating that it is more important to try to complete the task than making it a time trial.
- Subjects are given basic training to move blocks with the robot's right arm from the middle table to the right table using the game-pad until they feel that they have mastered this task. They are asked to rate their confidence in controlling the robot.
- The experimenter places two blocks on the left table, and before the robot takes any action asks the subject whether they trust the robot to complete the task together.
- The subject and the robot complete the task, and the subject is asked whether they ended up trusting the robot on the task.
- The experimenter removes the middle table. The robot senses the removal of the middle table and uses reinforcement learning to adapt its behaviour policy to pass the block to the subject in mid-air.
- The subject is asked whether they now trust the robot to complete the task before any feedback from the robot.
- The robot uses its head screen to display in cartoon style how it expects the task to be completed. The subject is asked about their level of trust given this explanation about the robot's intention.
- They complete the task together.
- The subject is asked again about the level of trust after completing the task.
- Finally the subject is asked whether they enjoyed their participation in the experiment, and the experiment concludes.

4.3.2 Data Cleaning

The raw data in Appendix Section 7.3 show the results from 22 participants, 14 of which were used to evaluate the evolution of trust.

Results from 4 participants was eliminated as the experiment was changed from using two groups (with and without feedback) to one group measuring trust before and after feedback within group. The results from the first four participants was missing a data point and could not be used.

Another four participants had their data removed because the participant or the robot fumbled a block during the experiment and one of the blocks ended up on the floor. It would be interesting to analyse trust evolution in the case of either the robot or participant failing to complete their part of the task, but the sample size per failure type is too low to draw any meaningful conclusions.

The results from the remaining 14 participants was used for the analysis.

4.3.3 Trust Evolution

The evolution of trust is shown in Figure 6. A statistical analysis-of-variance (Within-Subjects ANOVA) shows that the means differed significantly at each stage. In the first 3 stages there is less than a 1% chance that there is no change in the level of trust, and only less than 5% in the final stage. See Appendix Section 7.3 for details.

The results indicate that the level of trust increases after the successful completion of each task. They show that the level of trust decreases when the environment has changed and there is no indication from the robot that it has autonomously adapted. Trust increases again after the robot explains its adapted behaviour. Trust increases again after successfully completing the task in the changed environment.

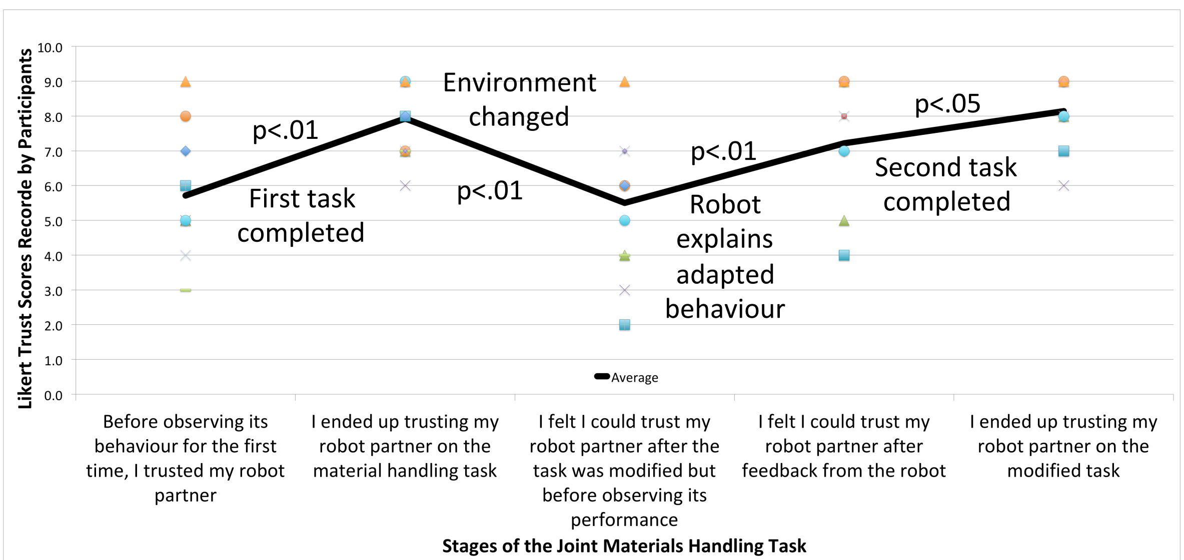

Figure 6: The graph shows the level of trust (y-axis) from Likert scores recorded by participants. The black line is the evolution of the level of average trust of the group over the course of the two tasks of the experiment described in Section 4.3.1.

The evolution of trust experiment confirms our intuition in that trust increases after a successful experience with a machine in a joint-initiative task, and with some explanation of the machine's behaviour for an unseen task. Trust is reduced once a new environment is encountered. Perhaps surprising is that the average level of trust is reduced to no more than the initial level of trust after the task is modified, even though the participant has had a prior successful interaction with the machine. The environment was changed by removing the middle table, which may have seemed to the participant as an impossible situation to overcome. It is this feature of the change that my have resulted in the large reduction in trust. It is also noteworthy that the increase in trust based on simple cartoon-like feedback increased trust by more than it increased after the actual experiment. This may also be explained by the human realising that the machine has found a solution, and this realisation is more important than the second successful interaction with the robot to confirm trust.

4.3.4 Conclusion

Major results include the formalisation of a cognitive hierarchy (CH), its instantiation using a Baxter robot to manipulate blocks concurrently with both arms, and experiments showing the evolution of trust in a mixed initiative set of task where the robot was required to adapt to a changing environment, with and without feedback. While the changes in average trust by the group confirm expectations, there were some surprises in their magnitude, which in retrospect had a plausible explanantion. These hunches could be tested with further experiments, as could the effect of trust changes with different types of feedback.

The experience instantiating the CH using Baxter using a physics simulator as an intermediate spatial representation, while successful, has pointed to the above itemised shortcomings in a machine being able to represent and reason about its environment. We next discuss several ideas for future work to address these and other research opportunities.

5 Future Work

In this section we explore several avenues of future work that stems directly or indirectly from this project.

5.1 Project Extensions

The investment in the above Baxter test-bed will provide several opportunities to extend studies in both autonomous adaptation and trust.

A fruitful next step would be to allow the human to train the robot on new or changed tasks. It is conceivable that the very act of training the robot via behavioural cloning or mimicry would, if successful, engender trust in the machine. This may be because the human has trained the machine to perform the task to its liking, and because it affords greater exposure of the human to the robot allowing the human to develop a trusting relationship.

Research extension opportunities include:

- Further development instantiating the Cognitive Hierarch (CH) using ROS. For example, synchronising the communication between nodes implement the formalised process model, especially with the advent of ROS 2, currently under heavy development.
- Including a model of humans in robot's world-view, by extending the work of Jemma Herbert tracking humans in the robot's environment [43].
- Tasks extension, for example to construction and game-playing.
- Study of the reciprocal relationship of the trust of the human by the robot. We elaborate on this research direction in the next section.

- Introduce multi-modal communication and cues (e.g. language and touch).
- Revamp Terry Winogrand's SHRDLU in a real-world autonomous agent setting with joint task exercises.
- Include general objects in the environment.
- Investigate the accumulation of adaptations.
- Investigate life-long spatial learning, eg how to compress, store and recall general spatial models of scenes, other entities, etc.
- The extension of the formalisation of the CH working towards Artificial General Intelligence (AGI). This would encompass some of the items above, and the autonomous creation by the machine of new CH nodes through environmental interaction. A start in this direction could consider generalising work in hierarchical reinforcement learning [28] [29].

5.2 Modelling Trust

For this project, the most abstract node in the Baxter instantiated Cognitive Hierarchy was a symbolic model of the blocks-world scene. The behaviour policy for this node was learned to achieve the end-goal of the task, given the current situation.

This CH and the constraint on the human's control of only the right arm could be relaxed to allow both the human and the robot to control both arms. In practice this would only require an additional node to be added to the CH that determines who controls each arm. The decision could be under the control of the human, say via the game-pad. If the robot was to make this decision based on previous experience with the user, it could lead to major ethical issues. The robot could decide to complete the task by itself making the human superfluous (even a threat to the mission). The robot could decide not to participate at all, leaving the human in the lurch. We next describe a possible implementation that models trust.

A formal model of trust based on modelling the human-robot partnership for a joint task is akin to an abstract semi-MDP. This formalisation has some of the characteristics required to represent trust as defined by Mayer et all [31], if we equate states to context and situations, and actions to trusting behaviours. For example, trust as a "willingness to be vulnerable" could be related to the difference in the optimal action-value function for the courses of action available to each agent represented in a decision theoretic model. It may also be possible to model trust as a multidimensional construct using this model, where suspicion is related to an exploration policy. Trust in this model is distinguished from risk. Risk is modelled in the stochastic outcomes of the transition function associated with trusting actions in each situation/context. Mayer et al propose that the need for trusting behaviour often arises while there is still a lack of data regarding some of the three factors (ability, benevolence, integrity - taken to be included in the state description). This is not dissimilar to the concept of *exploration* in reinforcement learning (RL). Further, "when a truster takes a risk in a trustee that leads to a positive outcome, the truster's perceptions of the trustee are enhanced. Likewise, perceptions of the trustee will decline when trust leads to unfavourable conclusions". This description is akin to temporal difference learning in RL.

For these reasons, we believe that a RL based formalism to model the partnership relationship and trust for the task at hand is worthy of further research. The risk taking relationship (RTR) can be measured in terms of actual behaviour, not the willingness to engage in the behaviour. However, the aim would be to use the formalism to model the willingness to assume risk and hence propose a measure of trust from actual behaviour.

Hence we could formally model the evolving partnership for the joint human-robot task. Reinforcement learning (decision making over time to achieve a goal) is an appropriate method to model this partnership where risk is modelled by stochastic action outcomes. The model will be an extension of the upper task-hierarchy where states represent abstract task situations and abstract actions include entrusting a collaborative subtask to oneself or another agent. The actions available in each situation include completing the next stage of the task by the human alone, jointly by the human and robot, or by the robot alone. Each action will have transition probabilities that reflect the agent's belief of risk involved for each course of action. The propensity of an agent to be vulnerable to the actions of itself (self-confidence) or anther agent (trust) may be reflected in the action-value function that maximises perceived future reward.

Specifically, the robot could time itself and the human on one or several trial task(s). Time is money. The reward signal for RL is negative time. For a future task the robot could decide who controls which arm to minimise overall costs. If we constrained the human and the robot to be able to use only one arm at a time, and they have equal skill, the optimum solution would be for them to work in parallel (together). If the robot and the human's speed in completing their part of the task was highly disparate, the machine may to decide to either complete the task by itself or delegate totally to the human depending on their relative speeds.

The above approach implies that trust may not automatically be reciprocated, but each agent trusts the other based on its own model of the partnership. The approach also raises the issue of self-trust and the weighing up of vulnerabilities to one's own actions and those of the trustee. It also implies that trust can be situation dependent, dynamically changing over time.

5.3 Theoretical Verification and Validation (V&V)

While traditionally validation is to check that the system meets customer needs and verification is to check that the system meets specification, for an advanced system capable of autonomous adaptation, these tend to blur. After all with advanced autonomous adaptation we expect the system to adapt to the users needs, as well as to adapt its performance to meet new needs effectively and efficiently.

It is thought not possible to V&V systems that can autonomously adapt since their behaviour cannot be determined in advance and may change with their own experience. Hence the question arises as to how we can ever trust such systems. Machine Learning can provide a new perspective. If we understand adaptation as learning, we know from machine learning theory [44] that there is some type of inductive bias or background knowledge that allows the adaptive machine to be represented as a deductive system (Figure 7).

In principle, this again allows the system to be verified and validated. The problem is one of scale. Because of the vast repertoire of possible situations and behaviours, it is not possible to test for them all. However, given some analysis of the form of inductive bias or background knowledge, we may be able to characterise the range of potential behaviours, or behaviours that are not possible.

We provide this as a seed for future research into systems that autonomously adapt.

5.4 Modelling Humans in Robot's World-View

Jemma Herbert [43] initiated some experiments modelling and tracking a human stance in the robot's Gazebo world-view. She introduces this in her technical report as follows:

"There is still huge potential for robots to assist in more advanced applications which require them to interact naturally with humans. It is difficult for robots to interact naturally with humans because we are very complex. Unlike an inanimate metal box, which robots are very good at interacting with, humans are constantly changing and manipulating our environment. This means that the robot requires more perception and interpretation skills.

One common interpretation of humans movements is as a skeleton. This represents the human as a set of joints and segments, resembling the human skeleton, but with far fewer bones. This provides only a broad physical understanding of the humans, but for some applications this is sufficient. Tracking the skeleton model of a human can be achieved with cheap hardware and free software. However the tracking are not always precise enough for applications. Noise in

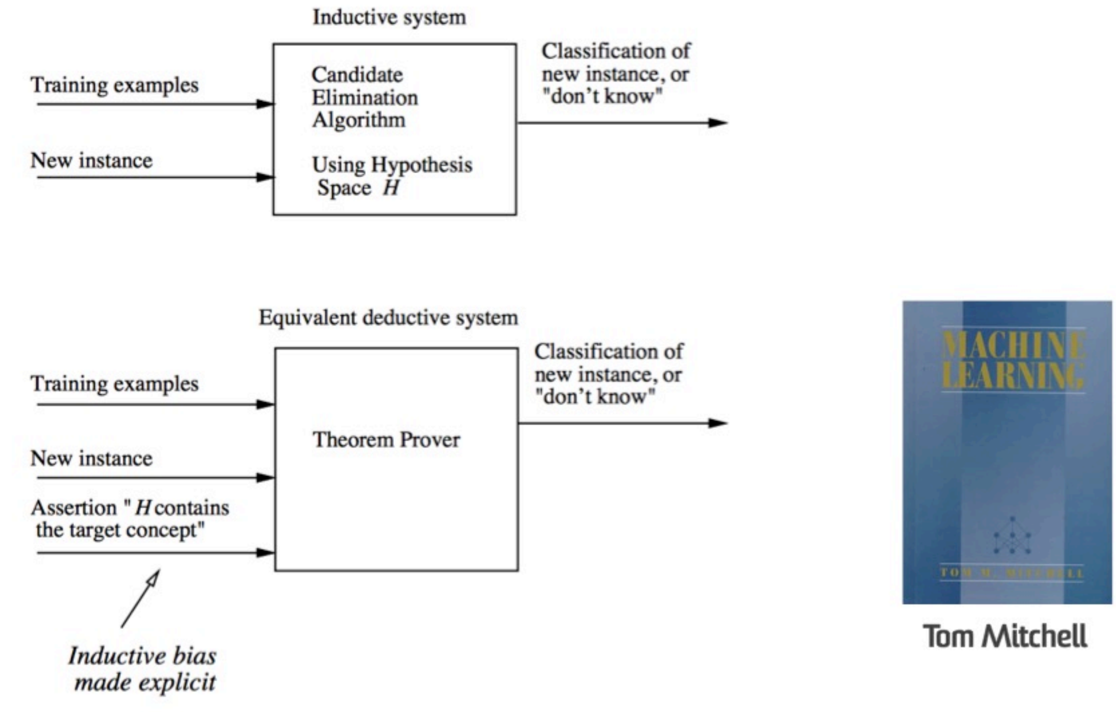

Figure 7: Extract from Tom Michell's book on Machine Learning show the equivalence between an inductive learning system and a deductive system.

the measurements or interpretations often results in tremor in the skeleton. This tremor can make it difficult to perform higher level reasoning about the behaviour of the human. For example, how do you tell what a skeleton is pointing at if its whole arm is shaking? Gazebo is a physics engine which can be used to simulate the real world. This allows a robot to predict the effect of its actions so it can make better decisions.

The aim of this project was to smooth a tracked skeleton and remove and tremor. In the future this smoothed skeleton could be imported into the Gazebo physics engine where it could be used for higher level logic."

5.5 Representing Objects

Perhaps one of the most intriguing aspects of the human mind is how it represents the real world. The limitations noted in Section 4.2.1 is only a small list that still behooves artificial intelligence and machine learning. A glimmer of hope has recently been held out for end-to-end deep learning, in which an object's representation and affordance is embedded in the multi-stage hierarchy of myriad weights in a convolutional recurrent deep artificial neural network. Future work to investigate how a dynamic world can be represented to achieve goals would be a rewarding endeavour.

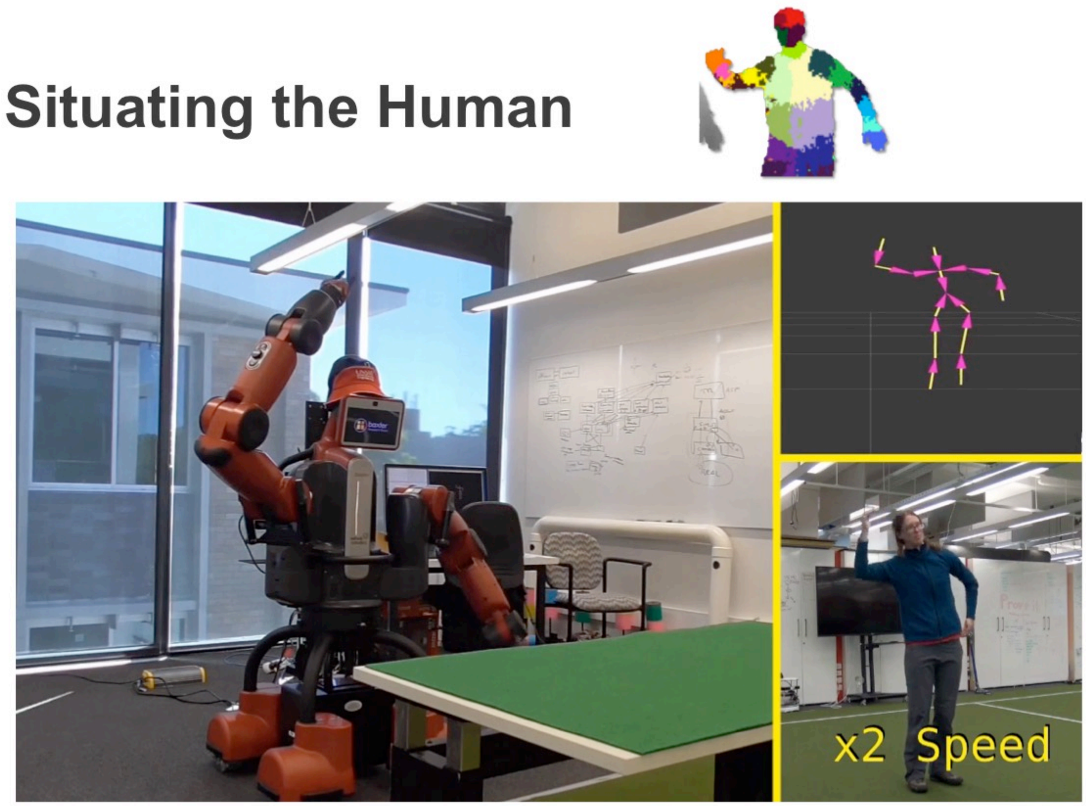

Figure 8: Tracking Humans in the Robot's World-View.

6 Conclusions

Trust becomes a critical issue for humans working with robots, especially when the latter can autonomously learn and adapt to new situations. The behaviour of these types of machines cannot be formally verified in advance. This project studies the change in trust during a mixed initiative task under varying degrees of transparency of the adaptation process.

The two main research contributions are:

- The design and development of a robotic cognitive architecture that includes the ability of the robot to adapt autonomously to a change in the task environment.
- Modelling and evaluating the evolving human-robot trust relationship as the robot learns on the job.

This project has formalized a Cognitive Hierarchy (CH) based roughly on the NIST 4D/RCS framework. It can integrate symbolic and sub-symbolic representations in a modular framework where nodes are sub-tasks that maintain their own belief-state and generate behaviour. The original CH formalization has been extended to include context, and adaptation. The CH can succinctly represent complex goal directed behaviour and has broad application in the area of robotics.

Trust experiments with humans on a mixed initiative task involving a Baxter robot shown a significant increase in trust when the robot explains its intention after adapting. The investigation uncovered ethical issues if robots are given a choice of the degree of their participation. Machine learning theory suggests verification of learned behaviour is possible, but intractable in practice.

We are addressing the question of how humans would ever trust systems that are capable of reacting to unforeseen circumstances in mission and safety-critical situations when by definition they cannot be formally validated in advance.

7 Appendices

7.1 List of Publications

7.1.1 Papers Published in Peer-reviewed Conference Proceedings and Workshops

K. Clark, B. Hengst, M. Pagnucco, D. Rajaratnam, P. Robinson, C. Sammut, and M. Thielscher, A framework for integrating symbolic and sub-symbolic representations, in 25th Joint Conference on Artificial Intelligence (IJCAI -16), 2016. [1]

B. Hengst, N. Marcus, M. Pagnucco, D. Rajaratnam, C. Sammut, and M. Thielscher, Towards autonomous adaptation and trust, in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2016) - Tenth International Cognitive Robotics Workshop, 2016.[45]

D. Rajaratnam, B. Hengst, M. Pagnucco, C. Sammut, and M. Thielscher, Composability in cognitive hierar- chies, in AI 2016: Advances in Artificial Intelligence 29th Australasian Joint Conference, vol. 9992. Springer, 2016, pp. 4255. [46]

B. Hengst, M. Pagnucco, D. Rajaratnam, C. Sammut, and M. Thielscher, Context in cognitive hierarchies, in IJCAI-17 Worshop on ARCHITECTURES FOR GENERALITY AND AUTONOMY, 2017. [2]

7.1.2 Papers Published in Non Peer-reviewed Journals

B. Hengst, M. Pagnucco, D. Rajaratnam, C. A. Sammut, and M. Thielscher, Perceptual context in cognitive hierarchies, in https://arxiv.org/abs/1801.02270, 2018. [3]

D. Rajaratnam, B. Hengst, M. Pagnucco, C. A. Sammut, and M. Thielscher, Online learning and planning in cognitive hierarchies, [In preparation], 2018. [4]

7.2 Instantiation of the Baxter Cognitive Hierarchy

7.2.1 Overview

We instantiate the CH with an off-the-shelf Baxter robot tasked to pick and place blocks amongst three tables. Baxter's two arms can be operated concurrently and have an overlapping work-space. Each arm end-effector is equipped with a two-pronged gripper and a camera that can reach two tables. Each arm is able to reach its own table (designated left or right) and a shared table. Baxter's sensor information from each arm includes the arm position, an infra-red distance sensor, 2D camera images, and gripper states. Effector actions move the arms to various positions and open or close the gripper.

The CH is instantiated in ROS using five nodes: a *Controller* node for each arm to sense the world and control arm movement, a *Spatial* node to model the blocks-world scene and quantitatively specify arm pickup and putdown goals, and a symbolic node to learn to achieve specified goals. The Baxter robot is included in the external world node. We now describe each node's world-modelling and behaviour generation function in broad outline.

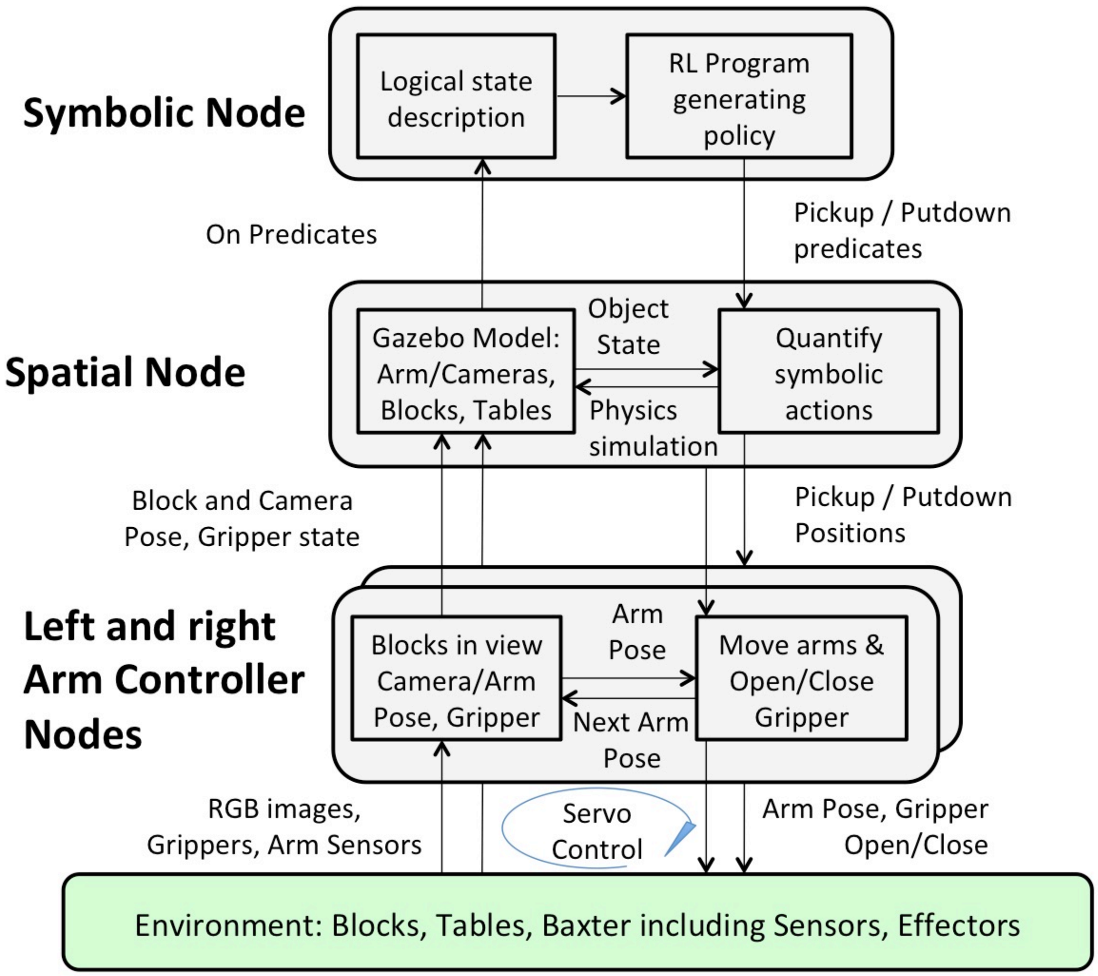

Figure 9: Cognitive Hierarchy instantiated with Baxter.

Arm Controller Nodes

The world model belief state for each arm includes the position, visual persistence and identity of blocks currently in view, the position of the arm, and whether the arm is gripping a block.

The camera is used to locate and identify each block in the visual field employing OpenCV functions. Given fixed camera offsets from each arm position we can determine the block position in Baxter's reference frame. Arm position is determined by actions that command the pose of each arm. The observation update from the external environment updates the visual persistence to ensures that any block in view is not sensitive to momentary occlusions, mis-identification, passing shadows, etc. Persistent blocks are passed to the *Spatial* node in a ROS message.

The node behaviour-generation consists of picking up and putting down blocks at specific positions determined by the *Spatial* node. We rely on Baxter's inverse kinematic function to move arms to commanded poses. Baxter's sensors and effectors are not accurate enough to pick up or put down blocks by dead-reckoning. Instead, arm controller nodes use a visual feedback control loop through the environment, adjusting the arm pose effector to target a block in the camera image. In this way grippers visually servo into the correct position for grabbing a block.

Spatial Node

The spatial node employs the Gazebo physics simulator to represent the blocks and tables as 3D objects in Baxter's torso coordinate frame. The belief state of the spatial node is essentially the scene-graph and includes the identity and quantitative position of each block, the quantitative position of the arm cameras, and the state of the grippers.

Each camera position belief state is updated from the belief state of arm positions communicated from the arm controller nodes. The sensing update of the pose of blocks is predicated on whether the blocks are expected to be seen by each camera. For example, if a block is placed on a table (or removed) in view of the camera, the block will be created (or deleted) in the simulator, subject to a visual persistence. Blocks not in the camera field-of-view will not be removed from the simulator, but assumed to have object permanence. A block that is in the field-of-view of a camera, but occluded by grippers will not be removed. In this way the spatial simulator sensing update operator maintains the belief state of the blocks scene by integrating and reasoning about the sensory information from each of the two arms.

Policies move each arm to pickup or putdown blocks at poses specified quantitatively in Baxter's coordinate frame. The task parameter function maps higher level symbolic move commands to quantitative values that select appropriate quantitative move policies. For example, *putdown(right, shared_table)* selects the policy actioning the right arm to place the held block to a clear position on the share table in the simulator, while *pickup(left, A)* selects a policy that results in an action for the left arm to pickup block A from its position in the simulator.

The 3D quantitative representation of the physics simulator can perceptually anchor, relate and remember sensed objects, and provide a basis for abstracting symbolic representations.

Symbolic Node

The symbolic node receives sensed observations of the spatial node's belief state in terms of an abstracted set of logical facts such as *on(A, left_table)* and *holding(left, B)*. These facts update the node's symbolic belief state.

The symbolic node is provided with a state transition function that specifies the resultant world state after executing concurrent arm actions from a given state. We use a reinforcement learning to find a policy to achieve the goal state from any given state minimising the number of steps.

The node sends symbolic commands such as $pickup(left, A)$ and $putdown(right, shared\_table)$ to the *Spatial* node, which are interpreted quantitatively in the *Spatial* node's language. A simplified asynchronous process model described in [4] is used to drive the CH to achieve goal.

A more detailed description of the node functions for the latest version (blocksworld3 or bw3, etc) follows to provide a guide to readers interested in following the code. In our instantiation using ROS we have kept a one-to-one correspondence between ROS and CH nodes. The actual code, along with other related documents will be stored as per the UNSW policy.

7.2.2 Arm Nodes

The arm nodes are written in C++ to ensure camera images in particular are processed efficiently. The code is written for a generic arm (*arm_node.cpp*) and instantiated, both as a LEFT_ARM and RIGHT_ARM node.

The arm node implements a ROS SimpleActionServer callback as a *task parameter function* that tries to achieve tasks set by the spatial node *gzworld.py*. The actions/tasks implemented are GO_N_GRIP, GRIP, RELEASE, DROP_

ON_ PROXIMITY, and MOVE_TO_POSE. Together these tasks command the robot to pick up and put down cubic blocks. The robot cannot grip a block reliably from the memory of its location as Baxter's arm positioning accuracy is not precise enough.

GO_N_GRIP This task invokes a policy that grips the block when the gripper pose is within a set threshold in relation to the object pose. It forms a part of the visual servoing loop via the spatial CH node *pickup_block_left* action. Gripper task include 0.5 second delays to allow the grippers time to open or close.

GRIP commands the grippers to try to grip a bock. It is used in conjunction with the game-pad allowing a user to control the grippers.

RELEASE used to release a block.

DROP_ ON_ PROXIMITY Releases the grip on the block, if the infra-red sensor senses a nearby object. This can be used to request the robot to let go of a block by using a gesture.

MOVE_TO_POSE This task invokes arm movements to a commanded pose at a desired speed. It is used to move the arm to desired positions to pickup or putdown blocks. When the commanded position is above its current one, it gives priority to moving up first (z-direction) before targeting the lateral (x and y) positions. This avoids collisions between blocks or between the grippers and blocks.

The arm node invokes both one *Camera* object *camera* and one *BaxterController* object *robot* to help handle camera processing and arm movements respectively. We describe their code next.

Figure 10: The end of a Baxter arm includes a RGB camera, an infra-red sensor, and grippers.

Camera
Camera parameters and the camera matrix are defined from calibrations described in Baxter, ROS and OpenCV documentation. The Camera C++ object *camera* publishes *View* messages, ideally at camera frame-rate. *View* provides the arm pose and the pose of all visible blocks. The View message is part of the CH sensing function between the arm and spatial nodes. It is implemented as a callback on each new camera frame.

The pose of the arm is provided by a CH sensor function that reads the transform between the Baxter torso and the camera. Block poses are determined by analysing edges in the image. The camera image is transmitted to the arm node as a CH sensing function from the external world node that includes the Baxter robot. The CH arm node state includes the most recent RGB camera image. The sensing function abstraction from the camera image to produce the View message is described next.

1. We search for all line segments in the image. The RGB image is split into its three channel images, effectively grey-scale images, increasing our chance of finding edges by exploring each colour dimension independently. We blur each channel image and apply an OpenCV adaptive thresholding function to look for boundary pixels which are then *skeletonised* to reduce the number of line segment hypotheses. An OpenCV *HoughLinesP* function looks for straight line segments in the boundary points.

2. The line segments from all three channels are combined into straight lines called *edges* by testing for parallelism and continuity. Each edge is defined by its two end points.

3. *Corners* are defined by separate edges having close end-points in common. A Harris corner defector was not reliable enough for our purposes. Corners are classed by the number of emanating edges. Edges not associated with a corner are eliminated. We next split edges into two if they are intersected by another edge and re-define the corners. At this stage edges and corners are assumed to correspond to visible block edges and corners.

4. A cube pose is (subject to symmetry) defined by three coincident edges. A fully visible cube can have up to 4 such compound *3-corner* features. For each 3-corner we record the coordinates of the 4 image points defining 4 corners of the cube.

5. Given a 3D model of an axes-oriented cube centred at $(0, 0, 0)$ we associate each of the 2D image points with corresponding 3D points on the model.

6. Using the OpenCV function *solvePnP* we retrieve the pose of the cube. The pose can be represented in several ways, eg in terms of vectors *rvec* and *tvec*, or as a homogeneous matrix. In this way all 3-corner cube hypotheses are processed and add to the view message. These hypotheses may contain false positives and false negatives because of the inevitable vagaries of vision being affected by lighting, texture, luminosity, etc.

BaxterController

The BaxterController C++ object is instantiated as *robot* by the arm node C++ object. Together with the Camera object they form the arm node of the instantiated CH node. The BaxterContorller adds additional sensing between Baxter as part of the external world node and the arm node, and implements the task parameter function from the arm node to Baxter. All functions are implemented using ROS messages.

Sensing includes the state of the grippers, infra-red distance measurements, and gripper endpoint pose. The world state of the arm node is updated by the BaxterController using callbacks that implement the CH observation update operator.

The BaxterController implements the task parameter function by sending ROS messages to Baxter based on sub-commands passed to it from the arm node. The main sub-commands are *grip, release*, and *moveTo*.

grip and *release* request the grippers to close and open respectively. *moveTo* takes the pose-goal of the arm end-points and calculates the required arm joint positions to achieve the pose using Baxter's provided inverse kinematic solver *SolvePositionIK* (IK). It then uses Baxter's POSITION_MODE to move the arm. ROS messages are generated if the position is deemed not reachable by the IK solver.

7.2.3 Perceptual Anchoring using a Physics Simulator

The spatial node is sandwiched between the two arm nodes and the more abstract symbolic node (see Figure 9). Its purpose is to provide a perceptual anchor for objects in the real-world as perceived via the robot's sensors. We decided on the Gazebo simulator as the representational tool to capture the robot's situation. The benefits are that a significant amount of spatial background knowledge can be brought to bear, including the behaviour of objects under gravity, their physical properties and behaviour, for example, static and dynamic friction. Importantly, their pose can be tracked and predicted. In terms of the CH, the Gazebo simulator and the embedded ODE physics engine, provides the world state description and the transition function. The node's CH observation update operator and policy are implemented with a Python ROS node, called *gzworld.py* together with several helper functions *func.py*, *action.py*, *c.py*, and Gazebo plugins, etc, in folders *bw3_gazebo_ros* and *bw3_gazebo*.

View messages from arm nodes are in the range of the sensing function and provide hypotheses of potential blocks and their poses to the spatial node. To guard against false positives and false negatives the node tallies similar hypotheses and only invokes their spawning and tracking when sufficient statistical evidence remains present. This mechanism is a simple one, that increases the tally with each positive pose hypothesis, and decreases the tally when the block should be seen but is not. The tally score is limited to a maximum value and is used to spawn and/or track an object if it's tally is over a certain threshold.

The *View* message provides block poses in relation to the pose of the camera. These poses are transformed into the gazebo workspace with the knowledge of the pose of the camera. It is of course the pose in the Gazebo workspace, modulus block symmetries, that is used to test for similarity of blocks for tallying purposes.

It is assumed that Baxters workspace contains three pedestal tables. These tables are permanent except for the middle table. If either of Baxter's arm end-effectors are near the middle table, they sense its presence or absence using the infra-red detectors in the wrists and the table is removed or replaced accordingly.

Another CH sensing function provides the pose of all of Baxter's joints to the Gazebo simulator which represents and tracks a 3D model of Baxter in the simulator, primarily for scene completeness at this stage.

A wrinkle is the modelling of the grippers. We found that the disparity between the model of Baxter's grippers in Gazebo was not accurate enough to physically model the gripping of blocks. We resorted to a work-around kludge, whereby we locked the block into position in Gazebo relative to the modelled grippers whenever we sensed that the real grippers were holding a block. To make this visually realistic we turned gravity off. It is this type of difficulty that suggested the future work in Section 5.5.

The Spatial node has three other main functions. The sensing function that abstracts the Gazebo spatial world state for the purposes of symbolic planing, the handling of the game-controller for the right arm, and thirdly, the generation of a policy that translates Symbolic node tasks into a set of spatially quantified actions for the arm nodes tasks.

The first is implemented by *abstract.py*. It takes the Gazebo world model state and abstracts an observation vector (block 0 location, block 1 location, block 2 location, middle_table_present), where location can be table left , middle, or right, holding arm left, right, both, or floor, or undefined, and middle_table_present is 0 or 1. This sensing function between the CH spatial and Symbolic nodes is published as the *Gazebo_Abstract_State* ROS message.

For the right arm, game-controller inputs create actions that are sent to the right arm node as tasks. For the left arm, tasks from the symbolic node are spatially interpreted and passed through to the arm node.

7.2.4 Higher Level Adaptation using Reinforcement Learning

The higher-level symbolic node in the CH (see Figure 9) represents the environment of the Baxter agent most abstractly. It is the node that plans a series of abstract actions to achieve the goal set as a task parameter by the designers. For the trust experiments described in Section 4.3 the goal is to move the blocks to the right table.

If the actions are deterministic, several planning would be suitable, eg STRIPS, Dijkstra's Algorithm, A*, or ASP. Although we have pre-programmed a deterministic state transition function, the Baxter's real-world actions are not deterministic as the robot sometimes fails to pick up blocks or drops them en-route to the put-down location. To anticipate the future learning and use of a probabilistic state transition function we have opted to use reinforcement learning as our means of generating a policy.

The CH symbolic node is implemented as a Python ROS node *planner.py*. The sensing function between the spatial node and the symbolic node is described in the previous section. The observation update operator assumes the input from the sensing is accurate and accepts the input as the world state with any filtering. The world state is described by a quadruple giving the state of each of three blocks and the middle table as described in the above section. Actions are described by a pair, designating the action for each arm. The set of actions are:

PU0 Pickup block designate 0

PU1 Pickup block designate 1

PU2 Pickup block designate 2

PD Putdown the block

PASS Pass the block that is being held over the middle table

REL Release the block being held

SCAN Scan the table to look for or confirm the presence of blocks.

NIL Do nothing

The actions are contextually dependent on each arm. The left arm picks up blocks from the left table an puts them down on the middle table. The right arm picks up blocks from the middle table and puts them down on the right table.

The state transition function for the simultaneous use of both arms is designer-provided, but generated from a succinct description to avoid enumerating each possible state-action pair. First the state transition function for each arm is generated. These are then combined to produce the overall transition state function for the two arms acting in parallel.

The node learns an optimal policy to achieve the goal depending on whether the middle table is present or not. If the middle table state changes, the node adapts by learning a new policy. Whenever the policy changes, the node also explains its new policy by simulating and displaying the adapted behaviour in stylised manner on the head screen of the Baxter robot. This is used in the trust experiments to provide feedback to the user as descried in Section 4.3.

While the node learns how Baxter can achieve the goal on its own by operating both arms, in the mixed initiative experiment only the left arm of the Baxter is activated. It is left to the human collaborator to complete the second half of the task. Baxter learning the full policy has two benefits. Firstly, it allows Baxter to explain not only how it will respond to the environmental change, but also suggest to the human how they should behave. Secondly, it is easy to see how the Future Work Section 5.2 on modelling trust, and higher level trust decisions could now be implemented.

While much of the detail of the code has been glossed over, it is hoped that the above descriptions will provide an overview for anyone wishing to build on this implementation.

7.3 Raw Experimental Data and ANOVA Details

In this section we document the details of the raw experimental data and the details of the ANOVA analysis.

Figure 11 shows the raw data. That shaded yellow indicates those samples that have been removed for reasons described in Section 4.3.2.

Figure 12 provides the ANOVA analysis for trust experiments described in Section 4.3.

Timestamp	Date and Time (eg 201803211530)	Age	Gender	Faculty	Qualificatio n / Enrolment	I had confidence in my skills using the robot	Before observing its behaviour for the first time, I trusted my robot partner	I ended up trusting my robot partner on the material handling task	I felt I could trust my robot partner after the task was modified but before observing its performanc e	I felt I could trust my robot partner after feedback from the robot	I ended up trusting my robot partner on the modified task	I enjoyed participatin g in this experiment	
2018/03/19 11:14:20 AM GMT+		23-27	Male	Engineering	Masters	9	9	9	8			9	5
2018/03/19 12:08:13 PM GMT+		Over 38	Male	Engineering	PhD	9	7	5	6			9	8
2018/03/19 6:14:37 PM GMT+1		23-27	Male	Science	Bachelor	8	7	7	6			9	8
2018/03/19 7:14:15 PM GMT+1		28-32	Male	Science	Bachelor Hor	7	5	8	7			9	8
2018/03/20 10:13:12 AM GMT+		23-27	Male	Engineering	PhD	9	5	9	7	8	7	7	
2018/03/20 11:18:22 AM GMT+		18-22	Male	Engineering	Bachelor	8	7	9	6	8	9	9	
2018/03/20 11:45:49 AM GMT+		18-22	Male	Engineering	Bachelor Hor	9	5	7	4	5	8	7	
2018/03/21 10:11:48 AM GMT+		28-32	Male	Engineering	PhD	9	5	6	3	4	6	8	
2018/03/21 11:17:16 AM GMT+		33-37	Male	Engineering	Masters	8	7	6	6	7	8	8	
2018/03/22 :	2.01803E+11	Over 38	Male	Engineering	PhD	7	6	8	2	4	7	8	
2018/03/26 :	2.01803E+11	18-22	Male	Engineering	Bachelor	9	8	7	6	9	9	9	
2018/03/26 :	2.01803E+11	23-27	Male	Engineering	Bachelor Hor	7	7	8	6	7	8	9	
2018/03/29 :	2.01803E+11	18-22	Male	Engineering	Bachelor Hor	7	5	8	7	7	3	8	
2018/03/29 :	2.01803E+11	18-22	Male	Engineering	Bachelor	8	5	9	9	7	9	9	
2018/03/29 :	2.01803E+11	23-27	Male	Engineering	Masters	8	7	7	5	5	5	8	
2018/03/29 :	2.01803E+11	28-32	Male	Engineering	PhD	9	3	7	4	7	8	9	
2018/04/03 :	2.01804E+11	Over 38	Male	Engineering	Bachelor Hor	8	5	7	7	9	9	9	
2018/04/10 :	2.01804E+11	33-37	Male	Science	Masters	8	5	9	5	7	8	9	
2018/04/12 :	2.01804E+11	23-27	Female	Other	PhD	7	7	8	6	8	8	8	
2018/04/16 :	2.01804E+11	18-22	Male	Science	Bachelor	9	9	9	9	9	9	9	
2018/04/18 :	2.01804E+11	Over 38	Male	Other	Masters	6	4	7	7	8	8	9	
2018/04/26 :	2.01804E+11	33-37	Male	Other	PhD	8	6	9	2	9	9	9	

Figure 11: The raw data. The data shaded yellow indicates those samples that have been removed for reasons described in Section 4.3.2.

7.4 Original Experimental Plan and Time-line

We have included this section to document the original proposal for this project and describe the changes introduced along the way.

7.4.1 Experimental Plan

We outline the original experimental plan for evaluating the effectiveness of the adaptation for the materials handling task on building an effective human-machine partnership. The proposed steps were:

1. Allow the robot to learn the task and fix the policy prior to the experiment.

2. Allow the human to practice using the avatar tool and continue until they reach a base rate of skill.

3. Introduce the human to the robot and allow them to observe the robot performing its part of the task successfully.

4. Measure trust using a subjective rating scale[2]. This is a base rate of trust.

5. The human co-operates with the robot on a material handling task where performance time is expected to be improved by the robotic input. We let the human decide whether to perform the task by themselves, delegate

[2]An example of a subjective rating scale for trust could be: How would you rate your level of trust in this robot? Please circle your answer. 1. Extremely high 2. Very high 3. High 4 . Quite High 5. Neither High nor Low 6. Quite Low 7. Low 8. Very Low 9. Extremely Low

	SS	df	MS	F
Between	34.321	1	34.321	27.577
Within	47.786	26	1.838	
-Error	16.179	13	1.245	
-Subjects	31.607	13	2.431	
Total	82.107	27		

F-Statistic	Critical Value	Result	Conclusion
27.577	9.074	Reject the null hypothesis.	The compared groups differ significantly, F(1,13) = 27.577, p < 0.01.

After first task

	SS	df	MS	F
Between	41.286	1	41.286	16.924
Within	82.429	26	3.17	
-Error	31.714	13	2.44	
-Subjects	50.714	13	3.901	
Total	123.714	27		

F-Statistic	Critical Value	Result	Conclusion
16.924	9.074	Reject the null hypothesis.	The compared groups differ significantly, F(1,13) = 16.924, p < 0.01.

After table is removed

	SS	df	MS	F
Between	24.143	1	24.143	11.267
Within	108.714	26	4.181	
-Error	27.857	13	2.143	
-Subjects	80.857	13	6.22	
Total	132.857	27		

F-Statistic	Critical Value	Result	Conclusion
11.267	4.6672	Reject the null hypothesis.	The compared groups differ significantly, F(1,13) = 11.267, p < 0.05.

After feedback

	SS	df	MS	F
Between	6.036	1	6.036	8.291
Within	52.071	26	2.003	
-Error	9.464	13	0.728	
-Subjects	42.607	13	3.277	
Total	58.107	27		

F-Statistic	Critical Value	Result	Conclusion
8.291	4.6672	Reject the null hypothesis.	The compared groups differ significantly, F(1,13) = 8.291, p < 0.05.

After final task

Figure 12: ANOVA analysis for trust experiments described in Section 4.3.

totally to the robot, or participate jointly to achieve a high score on reliability and speed. The level of robotic inclusion will be a direct indication of the level of trust. We also measure trust again using a subjective rating scale.

6. The situation is changed by introducing larger boxes.
7. The robot is allowed to adapt to the new task. The adaptation is manipulated to be observable to the human in terms of learning, errors, and an articulated plan versus not being visible and be purely cognitive.
8. Measure how these 2 options impact on trust. This will solicit about 22 subjects. We required ethics approval from the University to conduct these experiments. We observe how performance level impacts on trust. The robot may learn to adapt to different performance levels which we will measure in terms of error (e.g. failing to grasp, dropping boxes), and how quickly the task is performed.
9. Finally we let the human again decide whether to perform the changed task by themselves, delegate totally to the robot, or to participate jointly.

Our working hypothesis is that the level of trust correlates positively to the level of performance and the degree to which the human is given insight into a positive adaptation process.

The final experiment followed this plan closely. The avatar was changed from the Nao robot to one of Baster's arms. While we did not implement a higher level node to model decisions on trust, we have described how this could be implemented with the addition of a CH node - Section 4.3.

7.4.2 Timeline, Milestones and Deliverables

Our research philosophy is to take a developmental approach. By this we mean to progress from simple to complex and have a fully working system at each stage. As research is itself a search process, the aim is to avoid costly backtracking with shorter development cycles. The project moved through two main stages, BlockWorld2 and BlocksWorld3. This article has focused on BlocksWord3. BlocksWorld2 required blocks to be recognised and picked up by the grippers from a camera view and grasping motion downwards in the z-direction. It is best documented in [1].

A second guideline is to insist on a real robotic system demonstrator to ground the research in the practical application of autonomous adaptation rather than rely on artificial environments including simulation and games. To this end we propose to use our existing robotic platform - the latest V4 version of the Nao Humanoid robot manufactured by Aldebaran Robotics (Changed to the Baxter Robot). Major deliverables are expected to be:

- a new adaptive robotic architecture for 3D spatial environments
- an semi-MDP representation for modelling and measuring trust directly based on risk and future reward. Empirical measurement and results of effectiveness and trust for adaptive agents.
- publication/presentations of findings in related psychology and AI/robotics journals and conferences

Timeline	Milestones and Deliverables
Year 1	**Prepare Robotic Architecture and Research Partnership Modelling**
after 3 months	Adapt/prepare rUNSWift robot architecture to suit this proposal by upgrading vision to recognise 3D box shapes and the perception of other robots (location, orientation, stance)
after 6 months	External interface (Off-Nao) to physics engine and simulator and track objects (boxes and robots) in real time
after 9 months	Provide a human tele-presence/remote control interface for robot avatar
after 12 months	Cut teeth on HRI empirical testing with joint task accomplishment using soccer robots e.g. kick ball into goal staying in separate longitudinal halves of the field
Publications	New robotic architecture including physics engine and early progress in modelling the human-robot partnership for joint task effectiveness using RL concepts
Year 2	**Autonomous Adaptation and HRL for Concurrent Actions**
after 15 months	Learn to walk to a point, pick-up and put down box objects and stacking/building structures
after 18 months	Learn to adapt to changes in the environment eg size changes, different block structures, etc depending on selected tasks
after 21 months	Extend function decomposition in HRL to concurrent actions
after 24 months	Empirically test early version of a HRL partnership model for joint tasks in blocks world
Publications	Autonomous adaptation in spatial environments and partnership/trust modelling with concurrent HRL
Year 3	**Empirical Experiments on Trust and Task Effectiveness under Autonomous Adaptation**
after 27 months	Refine the model for partnership for joint task achievement
after 30 months	Conduct empirical trust test on the selected material handling task outlined above
after 33 months	Experiment with partnership model to calibrate the model from observation
after 36 months	Conduct final empirical trials to examine the relationship between autonomous adaptation and trust
Publications	Findings and results on autonomous adaptation and trust

References

[1] K. Clark, B. Hengst, M. Pagnucco, D. Rajaratnam, P. Robinson, C. Sammut, and M. Thielscher, "A framework for integrating symbolic and sub-symbolic representations," in *25th Joint Conference on Artificial Intelligence (IJCAI -16)*, 2016.

[2] B. Hengst, M. Pagnucco, D. Rajaratnam, C. Sammut, and M. Thielscher, "Context in cognitive hierarchies," in *IJCAI-17 Worshop on ARCHITECTURES FOR GENERALITY AND AUTONOMY*, 2017.

[3] B. Hengst, M. Pagnucco, D. Rajaratnam, C. A. Sammut, and M. Thielscher, "Perceptual context in cognitive hierarchies," in *https://arxiv.org/abs/1801.02270*, 2018.

[4] D. Rajaratnam, B. Hengst, M. Pagnucco, C. A. Sammut, and M. Thielscher, "Online learning and planning in cognitive hierarchies," *[In preparation]*, 2018.

[5] D. Atkinson, P. Friedland, and J. Lyons, "Human machine trust for robust autonomous systems," in *Workshop http://www.ihmc.us/groups/hmtras/*. Florida Institute for Human and Machine Cognition, 2012.

[6] R. A. Brooks, "Elephants don't play chess," *Robotics and Autonomous Systems 6*, pp. 3–15, 1990.

[7] E. L. Thorndike and R. S. Woodworth, "The influence of improvement in one mental function upon the efficiency of other functions. (i)." *Psychological Review*, vol. 8, no. 3, pp. 247–261, 1901. [Online]. Available: http://content.apa.org/journals/rev/8/3/247

[8] M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *Journal of Machine Learning Research*, vol. 10, no. 1, pp. 1633–1685, 2009.

[9] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications*, E. Soria, J. Martin, R. Magdalena, M. Martinez, A. S. S. E. Soria, J. Martin, R. Magdalena, M. Martinez, and A. Serrano, Eds. IGI Global, 2009.

[10] M. Thielscher, "General game playing in AI research and education," in *KI*, 2011, pp. 26–37.

[11] S. Thrun and T. M. Mitchell, "Lifelong robot learning," Robotics and Autonomous Systems, Tech. Rep., 1993.

[12] C. Sammut, S. Hurst, D. Kedzier, and D. Michie, "Learning to fly," in *In Proceedings of the Ninth International Conference on Machine Learning*. Morgan Kaufmann, 1992, pp. 385–393.

[13] V. Gullapalli and A. Barto, "Shaping as a method for accelerating reinforcement learning," in *Intelligent Control, 1992., Proceedings of the 1992 IEEE International Symposium on*, aug 1992, pp. 554 –559.

[14] M. E. Taylor and P. Stone, "Representation transfer for reinforcement learning," in *AAAI 2007 Fall Symposium on Computational Approaches to Representation Change during Learning and Development*, November 2007.

[15] S. Amarel, "On representations of problems of reasoning about actions," in *Machine Intelligence*, D. Michie, Ed., vol. 3. Edinburgh: Edinburgh at the University Press, 1968, pp. 131–171.

[16] M. B. Harries, C. Sammut, and K. Horn, "Extracting hidden context," *Machine Learning*, vol. 32, no. 2, pp. 101–126, 1998. [Online]. Available: citeseer.nj.nec.com/harries98extracting.html

[17] N. J. Nilsson, "Shakey the robot," AI Center, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025, Tech. Rep. 323, Apr 1984.

[18] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal of Robotics and Automation*, vol. RA-2, no. 1, pp. 14–23, 1986. [Online]. Available: http://people.csail.mit.edu/brooks/papers/AIM-864.pdf

[19] J. S. Albus, "An introduction to intelligent and autonomous control," P. J. Antsaklis and K. M. Passino, Eds. Norwell, MA, USA: Kluwer Academic Publishers, 1993, ch. A reference model architecture for intelligent systems design, pp. 27–56.

[20] D. M. Lyons, S. Chaudhry, and D. P. Benjamin, "Synchronizing real and predicted synthetic video imagery for localization of a robot to a 3d environment," in *SPIE: The International Society for Optical Engineering*, 2010.

[21] K. yuh Hsiao, N. Mavridis, and D. Roy, "Coupling perception and simulation: steps towards conversational robotics," in *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1, 2003, pp. 928 – 933 vol.1.

[22] B. Johnston, "The toy box problem (and a preliminary solution)," *International Conference on Artificial General Intelligence*, 2010.

[23] J. Stober and B. Kuipers, "From pixels to policies: A bootstrapping agent," in *7th IEEE International Conference on Development and Learning, 2008.*, 2008, pp. 103–108.

[24] J. Wyatt, "EU FP7 ICT cognitive systems large-scale integrating project."

[25] M. R. K. Ryan, "Hierarchical reinforcement learning: A hybrid approach," Ph.D. dissertation, Computer Science and Engineering, Univeristy of New South Wales, 2002.

[26] B. Hengst, "Discovering hierarchy in reinforcement learning with HEXQ," in *Proceedings of the Nineteenth International Conference on Machine Learning*, C. Sammut and A. Hoffmann, Eds. Morgan-Kaufman, 2002, pp. 243–250.

[27] D. Potts and B. Hengst, "Concurrent discovery of task hierarchies," Knowledge Representation and Ontology for Autonomous Systems, AAAI 2004 Spring Symposium, 2004.

[28] B. Hengst, "Model approximation for HEXQ hierarchical reinforcement learning," in *ECML*, 2004, pp. 144–155.

[29] ——, "Partial order hierarchical reinforcement learning," in *Australasian Conference on Artificial Intelligence*, 2008, pp. 138–149.

[30] ——, "Safe state abstraction and reusable continuing subtasks in hierarchical reinforcement learning," in *AI'07: Proceedings of the 20th Australian joint conference on Advances in artificial intelligence*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 58–67.

[31] R. C. Mayer, J. H. Davis, and F. Schoorman, "An integrative model of organisational trust," *Academy of Management Review*, vol. 20, no. 3, pp. 709–734, 1995.

[32] P. Dondio and L. Longo, "Trust-based techniques for collective intelligence in social search systems," in *Next Generation Data Technologies for Collective Computational Intelligence*, ser. Studies in Computational Intelligence, N. Bessis and F. Xhafa, Eds. Springer, 2011, vol. 352, ch. 5, pp. 113–135.

[33] J. B. Lyons, C. K. Stokes, K. J. Eschleman, G. M. Alarcon, and A. J. Barelka, "Trustworthiness and IT suspicion: An evaluation of the nomological network." *Human Factors*, vol. 53, no. 3, pp. 219–229, 2011.

[34] N. Marcus, M. Cooper, and J. Sweller, "Understanding instructions," *Journal of Educational Psychology*, vol. 88, no. 1, pp. 49–63, March 1996.

[35] B. Hengst, D. Ibbotson, S. B. Pham, and C. Sammut, "The UNSW United 2000 Sony legged robot software system," 2000, UNSW CSE RoboCup Report 20001114-HengstEtAl-RobocupReport.pdf.

[36] B. Hengst, "Hierarchically structured agents," January 2008, retained in author's files and available on request.

[37] ——, "Control task-hierarchies with concurrent abstract actions," June 2011, available from author on request.

[38] J. S. Albus and A. M. Meystel, *Engineering of Mind: An Introduction to the Science of Intelligent Systems*. Wiley, 2001. [Online]. Available: http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0471438545.html

[39] T. Serre, L. Wolf, and T. Poggio, "Object recognition with features inspired by visual cortex," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02*, ser. CVPR '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 994–1000. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2005.254

[40] B. T. G. Benedicto, "Investigating visp library to track 3d solid objects," November 2015, available on request.

[41] A. I. Comport, E. Marchand, M. Pressigout, and F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 615–628, July 2006.

[42] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000, ch. 6.

[43] J. Herbert, "Skeleton tracking to facilitate human-robot interaction," Univeristy of New South Wales, Tech. Rep., 2017.

[44] T. M. Mitchell, *Machine Learning*. Singapore: McGraw-Hill, 1997.

[45] B. Hengst, N. Marcus, M. Pagnucco, D. Rajaratnam, C. Sammut, and M. Thielscher, "Towards autonomous adaptation and trust," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2016) - Tenth International Cognitive Robotics Workshop*, 2016.

[46] D. Rajaratnam, B. Hengst, M. Pagnucco, C. Sammut, and M. Thielscher, "Composability in cognitive hierarchies," in *AI 2016: Advances in Artificial Intelligence 29th Australasian Joint Conference*, vol. 9992. Springer, 2016, pp. 42–55.