Technical Report No: 2003/07

Beliefs, desires and intentions in a hybrid coached agent architecture

Robert Logie

Jon G Hall

Kevin G. Waugh

2003

Department of Computing
Faculty of Mathematics and Computing
The Open University
Walton Hall,
Milton Keynes
MK7 6AA
United Kingdom

http://computing.open.ac.uk

Beliefs, desires and intentions in a hybrid coached agent architecture

Robert Logie[1], Jon G. Hall[2], and Kevin G. Waugh[2]

[1] Osaka Gakuin University, Faculty of Informatics
2-36-1 Minami Kishibe, Suita Shi Osaka, 564-8511 Japan
rob@utc.osaka-gu.ac.jp
[2] The Open University, Department of Mathematics and Computing, Walton Hall, Milton Keynes MK7 6BJ England

Abstract. This paper introduces a proposed research project, Patchworld, which is intended to investigate norm guided emergent behaviour in heterogeneous agent systems. Patchworld has two novel features; localised agent coaching and a *flat* hybrid agent architectural model
Patchworld aims to address a number of problem areas, most notably those of adaptive behaviour, learning transfer, truly decentralised systems and peer level agent architecture. Modal logic is used to provide a common thread through these problem areas and between the agent's architectural modules. Patchworld agents will look forwards, attempting to achieve their goals, using normal and non-normal modal logic. They will also look backwards, attempting to understand and improve their behaviour, using a logic with deontic modalities.
This paper has two purposes, firstly to describe the proposed research project and, secondly, to indicate where and how modal logic will be used in the proposed agent's operation. Other aspects mentioned above will be addressed in future work.

1 Introduction.

Patchworld is a proposed research project intended to investigate norm guided emergent behaviour in a heterogeneous agent system. Patchworld has two novel features; localised agent coaching, providing for a fully decentralised system and a *flat* hybrid agent architectural model

Patchworld aims to address a number of problem areas, most notably those of adaptive behaviour, learning transfer, truly decentralised systems and peer level agent architecture. Modal logic is used to provide a common thread through these problem areas and between the agent's architectural modules. Patchworld agents will look forwards, attempting to achieve their goals, using normal and non-normal modal logic. They will also look backwards, attempting to understand and improve their behaviour, using a logic with deontic modalities.

This paper is in five sections, this introductory section continues to briefly introduce the areas addressed, describes some related work and indicates what this project may contribute. The second section outlines agents and agent systems, in the context of this work, and describes some features of the proposed system. The third section covers the use of so-called *attitudes* in describing agent systems and indicates how these may be represented using a modal logic. Section four describes the proposed agent architecture in more detail indicating how the novel features, localised coaching and a flat architecture, will operate. The summary outlines what the research is intended to achieve and what future work is planned.

1.1 Coaching

a number of *humanistic* terms are used as a means of abstraction in agent systems research. It is common to speak of agents having *beliefs*, *desires* and *intentions* (these are important in the context of this paper and are discussed in sections 3.1 and 3.2). Such abstractions make it easier to describe and reason about an agent's actions. Similar abstractions are applied to interactions between agents, it is common to talk of agents *trusting* or *negotiating* with other agents. Many types of agent relationships have been studied but the coaching or advising relationship has received less attention than others[1]. Coaching agents have been, for the most part, used in adversarial domains such as RoboCup where coaches have had the privilege of being able to see all of the agent's world. Coaches in Patchworld will operate using local knowledge only. This knowledge will be augmented by that of other agents in its vicinity and it is envisaged that coaches will become experts on *local* matters.

1.2 Agent architectures

Agents can be broadly partitioned into two classes, reactive and cognitive. Reactive agents simply respond to cues, triggers and stimuli in their environment and do so by executing pre-programmed reactive plans – set sequences of actions – aimed at achieving particular goals. Cognitive agents may maintain an internal model of their environment and use this to formulate plans aimed, again, at achieving goals. Jennings [2] notes that a major selling point of purely reactive agent systems is that overall behaviour emerges from interactions between component behaviours. This is view championed by Rodney Brooks, a strong critic of symbolic approaches to agency. The conceptually simplicity of reactive agents masks a number of difficulties, notably those of designing agents in such a way that they can take account of non local information and in such a way as to be able to improve their agent level performance over time. Jennings further notes that agents using a large number of behaviours can quickly become too complex to understand.

1.3 Norms and norm guided systems

Normative systems, systems in which certain acts or behaviours are permitted, proscribed or obliged, have received a lot of attention in AI research. One of the reasons for this is that norms are seen as being able to provide an advantage to society at reasonable cost to individuals[3]. There has been a lot of effort aimed at trying to understand now norms can be incorporated into agent systems and there are still gaps in this understanding[4]. Boella and Damiano have recently investigated flexible norm compliance by filtering norms through an agent's existing goals and comparing the utility of norm compliance with the agent's current intentions[3].

Patchworld attempts to contribute to this understanding by considering norms in terms of safety and liveness properties and by applying a deontic logic based normative analysis after agents have discovered a method of satisfying a system norm. A system norm is, for example, a rule stating that there should be no fires. If an agent accidentally starts a fire then the *no fires* norm will compel it, or other agents, to find some way of putting the fire out.

2 Agents and agent systems

Despite the enormous interest in agents and agent systems research there is no widely accepted definition of what an agent is[5][6]. It will be useful to define an agent and an agent system in the context of this paper before proceeding.

Definition 1. *An* agent *is a bounded, rational autonomous, persistent, self interested and self motivated entity situated in an environment which it can at least partially perceive and at least partially manipulate.*

Definition 2. *An* agent system *is an environment populated by a number of agents which can potentially interact with each other either directly or indirectly via some environmental mechanism.*

There is nothing contentious in either of these definitions so they are offered without further justification, the reader is encouraged to refer to [7] or [5] for further background. Taken together, these definitions encapsulate what Wooldridge[5] terms *a weak notion of agency* such as used in agent based software engineering. These definitions have been deliberately left broad to allow for evolving agents and an evolving system. There is no a priori requirement for interaction as, for example, in the definition proposed in [8, page 40], the potential for such interaction is extant and mechanisms may be direct, as in an agent to agent conversation, or indirect where one agent obtains data from tokens placed in the environment by other agents.

2.1 Agents and agency

Wooldridge[5] uses the terms *agent* and *agency* interchangeably, this is justified if the agents under consideration are intelligent agents, since not all agents are intelligent we need a definition of agency. Minsky[9] uses the steering wheel of a car as an example, a driver can consider the steering wheel as an agency for changing a vehicles direction. The steering wheel *appears* to know what to do in response to the drivers commands and the driver has no real need to know what the steering wheel does on his or her behalf. When there is a problem and the driver needs to know what is wrong then it's better to think of the steering wheel as a non intelligent agent that simply performs a given task, there is little point in asking the steering wheel for a problem description. As another example, consider a coaching agent in a soccer team; without an understanding of what the game requires it would be very difficult for that coach to act deliberately so as to improve the teams performance.

Definition 3. *An* agency *is an agent or a number of agents that possess knowledge of a particular task. The agency does not necessarily possess the abilities required to carry out this task but these abilities must exist and must be available to the agency.*

A soccer coach may be the least physically able member of a team and, consequently, unable to score the goal that the team requires. However the coaches knowledge and communication abilities combined with the other team members physical abilities may well score the goal that the team needs.

2.2 The agent environment

An agent's environment provides context and support for its abilities. Being able to hear has no great advantages in an environment that does not support sound. The environment provides a collection of objects that agents (situated in that environment) can perceive and manipulate. This collection includes all of the agents in an environment so any approach used to model agents should also be able to model their environment. Russell and Norvig[7] list five principal characteristics which distinguish environments; determinism, accessibility, episodic or non episodic, discrete or continuous and static or dynamic.

These five elements will be perceived differently by agents (and, possibly, differently by each agent) situated in an environment and by an observer with full access. Events that may be linked by an external observer as a simple causal conjunction may not be obvious to agents in the environment. Any method used to model an agent system must be flexible enough to be capable of representing the system from the point of view of an observer as well as from the point of view of the least capable agent within the system. In addition to these physical characteristics an agent environment will have a social aspect. Social aspects of agent systems vary greatly and will be addressed later when work commences on a test system.

Definition 4. *An* agent environment *is a physical or virtual area containing a collection of objects that agents situated in that environment are able to at least partially perceive and manipulate. The environment will have a set of rules governing, as necessary, the physical properties of objects, constraining the behaviour of agents or requiring certain behaviour from agents.*

2.3 Patchworld

the idea for Patchworld grew from a discussions about deontic logic, safety and liveness properties[10] and StarLogo (see [11]) simulations. System liveness is embodied in simple, minimally skilled agents. Safety properties are distributed randomly throughout the world in *safety patches*. These patches contain system norms, indicating what the system ought or ought not to do, and some simple skills. As agents move around they will collect safety patches randomly and leave records of their internal state in a trail of *state patches*. The state trail allows other agents to indirectly acquire information or behaviours from their peers. It is anticipated that over time the agents will evolve into agents suited to dealing with certain aspects of system behaviour leading to the emergence of a stable system capable of dealing with disturbances. In further discussions the idea of a coaching agent was introduced. The coach is intended to assist agents or groups of agents to refine any behaviour patterns that they may have discovered and to facilitate the propagation of *good* behaviour patterns throughout the world.

Patchworld agents Emergent behaviour is something normally associated with reactive agents and learning normally associated with cognitive agents. Both are required here. The attraction of a completely new architecture hides a number of pitfalls[6, page 235], it was decided to pick the required features from existing architectures and combine them to form a *hybrid* agent combining both reactive and cognitive elements. There is a body of literature on hybrid architectures covering areas such as process management agents[12], hybrid control of robots using an interface agent [13], and symbolic/reactive hybrid robot control [14]. The only novel aspects of the proposed architecture is the ability of the coach interface to provide low level assistance to the basic agent module by analysing data from the history module. The coaching interface will occasionally provide low level assistance to the reactive core when it is unable to select an appropriate action. In most cases operation will be biased towards the reactive core with the coach interface providing occasional assistance. Dedicated coaching agents will be biased towards being controlled by the coach interface and will have a greater allocation of system resources for storing and analysing historical data.

Coaching agents When two agents communicate there may be difficulties in them being able to relate to each other. Each agent will have its own representation of the local environment and they may use different representations or names for the same actions or objects. It may be obvious to an observer that the two agents are in an identical situation but not to the agents. Agents may co-operate to achieve something by chance and be unaware of the key events in their recent histories that have got them to the point where they are co-operating. At a moment in the future either of these agents may encounter a coach and give that coach their historical data, the coaching agent will be able to analyse the data from an observer's point of view and condense the important points from the behaviour history so as to produce a plan or behaviour for the joint action.

Coaching agents are common in other environments, RoboCup provides many instances of coaches (see, for example, [15]) but these generally have global knowledge or a global view of the environment. Patchworld coaches are situated in the environment along with other agents and work with local knowledge and local perceptions. These coaches can, however, draw on the knowledge and perceptions of other agents and apply greater resources to reasoning with and about it. This gives Patchworld two levels of reasoning, weak in general agents and strong in coaches.

3 Goal driven agents and agents with attitude

The role of *attitudes* such as belief, desire and intention in agent design has long been recognised [16][17][18]. This section introduces these attitudes firstly from a philosophical point of view and secondly from a more agent centred perspective.

3.1 The BDI model

Bratman[17] notes that desires and intentions are *pro-attitudes*, that is attitudes which have a motivational role in agent behaviour. (Belief and knowledge are considered as *information attitudes*[5].) Believing that turning on an air conditioner will make me feel cooler will not motivate me to turn it on. This belief in conjunction with a desire to feel cooler may produce an intention to turn the air conditioner on. Desires and intentions function differently, I am writing this on a portable computer and can adopt the intention of turning the air conditioner without worrying about the consequences for my work. If I am writing on a non-portable machine then my beliefs about the capacity of the electrical circuit that my computer and the air conditioner share in conjunction with my desire to complete my work may prevent me from adopting an intention to turn the air conditioner on. It would be irrational to simply turn the air conditioner on knowing that the combined load may trip a circuit breaker preventing me from fulfilling both of these desires. Intentions are conduct controlling pro-attitudes whereas desires are potential influencers of action. An intention involves a commitment to action that is absent from a desire[3].

An intention may be *future directed*, I feel comfortable just now but knowledge gained from the weather panel of the morning newspaper may lead me to adopt an intention to turn the air conditioner on at 11am. I may not need to reconsider this so the intention has a characteristic *stability* or *inertia*. At 10am I change a library file and need to rebuild an entire application which requires that I switch to a more powerful non-portable computer thus introducing the breaker problem. I must now reconsider my intention to turn the air conditioner on at 11am. Given a change in circumstances my earlier future directed intention play a part in my subsequent reasoning. If there is no need for me to reconsider and

[3] Bratman terms this relation between intention and action the *volitional dimension* of commitment.

the time approaches 11am I must now decide how to switch the air conditioner on, there is a remote controller - somewhere. I need the remote controller to turn the air conditioner on so I adopt an intention of finding it. My earlier intention has influenced my reasoning and led to a new intention[4].

The simple examples above use a belief-desire model, the intention to turn an air conditioner on can be reduced to the belief that the air conditioner will make me feel cooler and a desire to feel cooler. This model has both descriptive and normative aspects, it attempts to structure a common-sense approach to action and it attempts to articulate a practical rationality. Is this simple model adequate? If one accepts the existence of a predominant desire then this model is adequate. My having a prominent desire to become cooler means that I desire this more than performing any other option that I deem incompatible. My intention to turn the air conditioner on cannot be identified with such a predominant desire because it does not admit either a volitional or reasoning centred commitment.

The tactic of reducing intentions to beliefs and desires is, it would appear, inadequate. Bratman goes on to consider intentions in the context of a boundedly rational agent. Bounded rationality introduces a requirement for *practical* as opposed to omniscient reasoning. This leads to intentions being considered as partial plans which can play a role in future reasoning. Intentions are both inputs to and outputs from an agent's reasoning. As inputs they can pose problems - where is the remote controller. They can also pose constraints - my desire to complete my work using a non-portable computer constrains my ability to use an air conditioner. Intention has two facets, one deals with intentional action and the other with co-ordinating plans, recognising these facets forces us to consider intention as a distinct element of agency.

The section heading mentioned *goals* but so far has only dealt with beliefs, desires and intentions, how do goals fit in? Cohen and Levesque (see [19, page 301]) don't formalise the concept of a *goal*, instead they consider only the consequences of goals. This leads to intention being considered as a persistent goal. Goals, in turn, are a subset of an agent's desires, the set of desires that the agent has intentions of fulfilling.

This puts us in a position to list some further definitions:

Definition 5. *A pro-attitude is something that plays a motivational role in an agent's behaviour.*

Definition 6. *A belief is a fact that an agent holds to be true in its present state and which may be true in the future.*

Definition 7. *A desire is a pro-attitude which can, potentially, influence an agent's behaviour.*

Definition 8. *An intention is a pro-attitude which controls an agent's behaviour and to which the agent has some degree of commitment.*

[4] This is termed the *reasoning centred dimension* of commitment.

3.2 The logic of the BDI model.

BDI components are usually considered against a background of branching time and possible worlds - a standard approach for models incorporating logical modalities. For some examples see[20][21][5]. Rao and Georgeff build on earlier work using possible worlds frameworks and consider intentions as being on par with beliefs and desires (this *elevation* of importance is on line with Bratman's consideration). This treatment allows management of different commitment strategies broadening their model's ability to deal with different agents. Rao and Georgeff also consider where agents have control over the outcome of their actions and where chance plays a part (this is covered in detail by Horty[20]).

Possible worlds semantics traditionally treats each world as a set of propositions and models beliefs as a *belief accessibility relation* linking certain worlds. A proposition is believed if and only if it is true in all belief accessible worlds[22]. Each belief accessible world leads, in turn, to a tree of further worlds. Note that belief accessible worlds are a subset of all accessible worlds. Intentions can be represented, similarly, by a set if *intention accessible* worlds. Recall that an intention involves some degree of commitment, this allows us to characterise the set of intention accessible worlds as a set of worlds that the agent has committed itself to attempting to realise.

Rao and Georgeff[16] describe these in three accessibility relations, one for beliefs, one for desires (which may be applied to goals) and one for intentions. These relations deal with two *types* of attitude, informational and motivational. The informational attitudes can be characterised by a KD45 system and the motivational attitudes by a KD system[23]. To be able to deal with both an agent must use a combined modal logic.

4 The Patchworld agent.

This section describes the architecture of the proposed agent then loosely describes interactions between agents and within the agent. The latter descriptions are aimed at providing a sketch of the logical foundation of Patchworld's operation.

4.1 General description.

Patchworld agents will start life as simple, minimally skilled reactive agents. as time progresses they will pick up new behaviour patches specifying simple behaviours and system norms. Over time and with the assistance of dedicated coaching agents these agents will improve their skills so as to become adapted to their environment and to be able to satisfy the system's safety and liveness properties. It is anticipated that they will be able to react to unusual or emergency situations so as to contain or remove problems. Reacting to emergencies is something that needs to be done quickly, ideally without wasting time deliberation over possible actions whilst the emergency worsens or causes damage. Patchworld agents need a reactive component to behave in such a manner.

By improving their skills agents are learning what they can do with their abilities and when to use them effectively. This can be by way of experimenting, exchanging information with other agents or by being coached in a particular skill which has been developed by other agents. Reactive agents are usually associated with emergent system level behaviour and not with learning and improving individual behaviour. They are, essentially, entities that perceive and react, the rules for reaction may be complicated but there is no deliberative step in the process. How can they learn? A neural net could be used in the agent's core but the requirement for training rather than coaching precludes this approach. Genetic algorithms are another option, evolution is by fitness for purpose and transferring evolved behaviours would simply involve lifting an improved reactive core from one agent and fitting it to another. This is a dangerous process as a behaviour evolved in one agent with a given set of other behaviours may not work in another agent with a different set of behaviours. There is no explanation or symbolic representation of the behaviour that can be used to analyse side effects or dependencies. Clearly some other approach is needed, something that can help the agent evolve and, at the same time, generate behaviours that are explainable and amenable to analysis. An additional cognitive module is proposed, this will act as a miniature on-board coach providing a low volume stream of analysis, suggestions and modifications for the reactive core. When the agent encounters a dedicated coaching agent the mini coach can communicate and, together, apply greater resources to analysing the agents behaviour history and current state so as to tune the reactive core for better performance.

The proposed Patchworld agent will be a hybrid with reactive and cognitive parts, When the agent is operating normally it will be the reactive part that is exercising control. When the reactive module has finished its cycle the cognitive module may get some time to analyse any new data that has become available. If the agent encounters difficulties, new events or new objects then the cognitive part will take over and attempt to decide on an appropriate action. If the cognitive module takes more than a cycle to complete its work then it may pre-empt the reactive module during the next cycle. This will bias the agent towards cognitive operation in difficult or new situations. Class A/B audio amplifiers provide a nice analogy, at low output levels their operation is biased towards class A mode and at higher output levels they shift to operation in class B mode.

4.2 The proposed architecture.

The architecture is illustrated in figure 1 and its main components briefly described below.

– BAM Basic Agent Module : This is responsible for low level agent functions moving, recognising whats a patch and what isnt, picking up and installing patches.

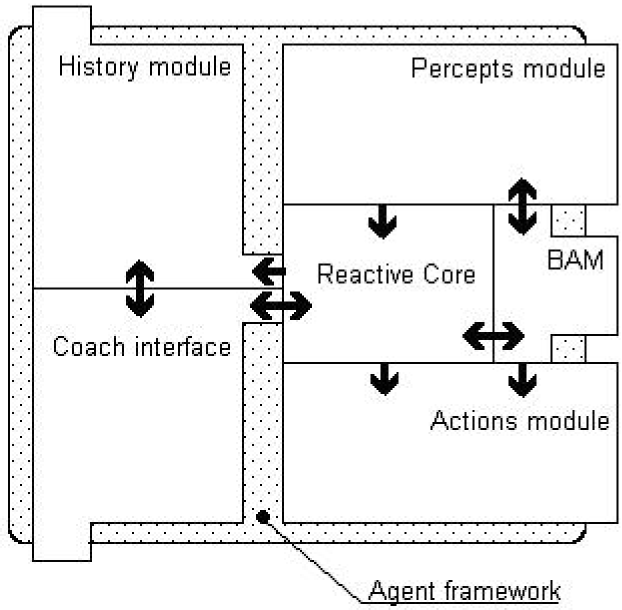

Fig. 1. The Patchworld agent architecture.

- Agent framework : This module carries out background tasks and manages communications between other modules. The main function of the agent framework is resource management, it is configured by the BAM and will provide resources to agent modules as instructed.
- Reactive core : This module is responsible for the agents general operation and how it acts in response to environmental data.
- Percepts module : This contains sensors for a subset of world elements and can furnish other modules with environment data.
- Actions Module : The actions module provides an interface which allows the agent to manipulate its environment.
- History module : This module provides general agents with a small store for maintaining a history of recent events. Coaching agents will have a much larger history module providing storage for a database of behaviours and sample histories. This will allow the coach to acquire sufficient data to allow analysis and refinement of behaviours.
- Coach interface : In non coaching agents this module has two functions, it provides low level assistance when the reactive core has difficulty in making a decision and it manages transactions with coaching agents updating the reactive core as necessary.

4.3 Interaction between and within agents.

Two normal of modal logic systems have been mentioned, KD45 and KD, have already been mentioned. These are generally accepted as being adequate for informational and motivational attitudes respectively. Recent work by Dastani and van der Torre[24] has suggested using non-normal modal logic to model an agent's desires. Goals form a logically closed set whereas desires do not. When an agent lacks knowledge of its world then numerous future worlds may be accessible to it[16]. It is intended that agents will use weak non normal logics for dealing with their limited local knowledge. Coaching agents will attempt to minimise uncertainty by pooling perceptions and data from a number of agents and will use a stronger, normal logic to make better use of the greater knowledge available to them.

The agent's reactive core will operate according to sets of rules (possibly complicated and with a temporal element) generated by the coach interface using the agent's local history module and by interaction with dedicated coaching agents. These rules will be updated when necessary. A coach may provide better alternatives or deem certain behaviours no longer useful. The agent may generate alternatives internally in response to the reactive core being unable to cope with combinations of events. The coach interface will use an analysis method working on modal principles to generate rules which are expected to achieve certain results. Where there is uncertainty or knowledge is lacking then these logical principles will be weaker than where there is greater certainty.

The idea of a flat architecture was mentioned in section 1 and section 4.1 outlined some general details about interaction between agent modules. In normal operation both the reactive and cognitive modules are operation, the reactive module will, generally, be responsible for selection appropriate actions and the cognitive module will carry out simple analysis of event histories. If the reactive module has trouble in deciding on an action then it will enlist the help of the cognitive module. Similarly, if the cognitive module improves on a reactive plan, perhaps by removing redundant steps, then it will present the updated plan to the reactive module. If the cognitive module has pre-empted the reactive module and a new problem appears in the environment then the reactive module will be able to appropriate control so as to attend to it. Unlike, for example, the subsumption architecture[25] or TouringMachines[26] there is no hierarchy of modules and both *sides* operate as equal partners using the agent's resources by mutual agreement.

The coach interface module will allow agents, either singly or multiply, to communicate directly with dedicated coach agents. The coach will use its greater resources to analyse the collective event histories of the agents and may be able to provide improved plans or sequences of actions. Is so then the coach agent will distribute these plans to the connected agents for incorporation into their reactive cores.

Agents will be unable to communicate directly with each other so in cases where co-operation between agents is required a triggering mechanism will generate a probability of the other agent co-operating. If an agent deems this probability sufficient then it will start the activity requiring co-operation unilaterally.

4.4 Analysis of event histories

Patchworld coach interface modules will use a backwards looking possible worlds representation of event histories (collected by several agents in the case of dedicated coaches) and apply a variant of standard deontic logic to this in an attempt to identify roles by assigning responsibilities to agents to ensure that they *see to it* that certain things are brought about. Horty[20] has presented an account of deontic logic in the context of agency. agency in this account is represented by Chellas's *stit* semantics representing agent's abilities to *see to it* that something is brought about It is envisaged that by the coach interface module's viewing events from the point of view of agents being able to *see to it that* something is done then roles will be identified and assigned.

5 Summary and future work

Much remains to be done. The ideas on using a weak non-standard logic in the presence of uncertainty and a stronger logic as uncertainty reduces need to be clarified and given a sound basis. An agent knowing what it doesn't know, characterised by 5 - the negative introspection axiom, can lead to problems for resource bounded agents. what an agent doesn't know is a potentially unbounded set. A means of reducing or constraining this uncertainty would be useful so a means of Using logics of varying strength depending on the amount of uncertainty present is a novel feature. Dastani and van der Torre's work[24] on non-standard logics for desires suggests that this may be a useful approach. When an agent is unsure about it's environment then this lack of knowledge may lead to multiple belief accessible worlds[16]. The introduction of, perhaps, a Bayesian reasoning module may be able to guide the agent's operation and assist in indicating an appropriate strength of logic to use.

Halpern notes that one of the more important aspects of probabilistic reasoning about is reasoning about dependencies, indepences and causality. Bayesian networks are a powerful and well understood tool for representing dependencies and independencies[27]. A Bayesian reasoning system in the coach interface module may be able to provide the required low level assistance

van der Torre et al.[28] note that deontic logic formalises reasoning *about* norms and not with norms. Deontic logic can tell us what ought to be done but can't give any assistance when it comes to stating which agent takes on a given responsibility. The abstraction provided by deontic analysis *after* agents have discovered how to do something may help in deciding how to allocate responsibilities for certain tasks. It is envisaged that further work, when a test system is built, will provide insights into this problem at least in the context of a system of agents with tightly constrained abilities.

Test systems being considered are evolving industrial or business processes. Such processes can be described normatively and constrained in their world by the placement of raw materials or process elements. Agents would evolve to manage these materials and processes.

References

1. Riley, P., Veloso, M., Kaminka, G.: An empirical study of coaching. In: Distributed Autonomous Robotic Systems 6. (to appear), Springer-Verlag (2002)
2. Jennings, N., Sycara, K., Wooldridge, M.: A roadmap of agent research and development. Autonomous Agents and Multi-Agent Systems **1** (1998) 7–38 June 2002 Printed.
3. Boella, G., Damiano, R.: An architecture for normative reactive agents. In Kuwabara, K., Lee, J., eds.: Intelligent Agents and Multi-Agent Systems, PRIMA 2002, Springer-Verlag (2002) 1–17
4. Lopez, F., Luck, M., d'inverno, M.: Constraining autonomy through norms. In: International Conference on Autonomous Agents. Volume 2., Bologna, Italy, ACM (2002) 674–681 October01.
5. Wooldridge, M., Jennings, N.R.: Intelligent agents: Theory and practice. Knowledge Engineering Review **10** (1995) 115–152
6. Wooldridge, M.: An introduction to Multiagent Systems. Wiley (2002)
7. Russell, S.J., Norvig, P.: Artificial Intelligence: A modern approach. Prentice Hall (1995)
8. d'Inverno, M. Luck, M.: Understanding agent systems. In: Understanding Agent Systems, Springer (2001)
9. Minsky, M.: The Society of Mind. Simon and Schuster (1987)
10. Lamport, L.: A simple approach to specifying concurrent systems. Communications of the ACM **32** (1989) 32–45
11. Resnick, M.: Turtles, termites and traffic jams. MIT Press (1994)
12. Debenham, J.: A multi-agent architecture for process management accommodates unexpected performance. In: ACM symposium on Applied computing, Como (2000)
13. Strippgen, S., Peters, K.: The other way round! - collaborative communication with agents. In: ACM Agents99. (1999) 108–115
14. Oliveira, E.: Robots as responsible agents. In: IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation. Volume 3. (1997) 2275–2279
15. Riley, P., Veloso, M.: Recognizing probabilistic opponent movement models. In Birk, A., Coradeschi, S., Tadokoro, S., eds.: RoboCup-2001: The Fifth RoboCup Competitions and Conferences. Springer Verlag, Berlin (2002) (extended abstract).
16. Rao, A.S., Georgeff, M.P.: Modeling rational agents within a bdi-architecture. In Fikes, A.J., Sandewall, E., eds.: proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning, KR91, Morgan Kaufmann (1991)
17. Bratman, M.E., Israel, D., Pollack, M.: Plans and resource-bounded practical reasoning. In Cummins, R., Pollock, J.L., eds.: Philosophy and AI: Essays at the Interface. The MIT Press, Cambridge, Massachusetts (1991) 1–22
18. Georgeff, M., Ingrand, F.: Decision making in an embedded reasoning system. In: Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI (1989)
19. Ferber, J.: Multi agent systems, an introduction to distributed artificial intelligence. Addison Wesley (1999)
20. Horty, J.: Agency and deontic logic. OUP (2001)
21. Halpern, J., Moses, Y.: A guide to completeness and complexity for modal logics of knowledge and belief. Artificial Intelligence (1992) 319–379

22. Halpern, J.Y., Moses, Y.: A guide to the modal logics of knowledge and belief: Preliminary draft. In: Proc. of the 9th International Joint Conference on Articial Intelligence, Morgan Kaufmann (1985)
23. Bennett, B., Dixon, C., Fisher, M., Franconi, E., Horrocks, I., Hustadt, U., de Rijke, M.: Combinations of modal logics. Artificial Intelligence Review **17** (2002) 1–20
24. Dastani, M., van der Torre, L.: Specifying the merging of desires into goals. In: EurAsia-ICT 2002: Information and Communication Technology: First EurAsian Conference, Springer-Verlag (2002) 824–831
25. Brooks, R.: A robust layered control system for a mobile robot. Robotics and Automation, IEEE Journal of **2, iss.1** (1986)
26. Ferguson, I.A.: TouringMachines: An Architecture for Dynamic, Rational, Mobile Agents. PhD thesis, Cambridge, UK (1992)
27. Halpern, J.: A logical approach to reasoning about uncertainty: a tutorial. In Arrazola, X., Korta, K., Pelletier, F., eds.: Discourse, Interaction and Communication. Kluwer (1997)
28. van der Torre, L., Ramos, P., Fiedeiro, J., Tan, Y.H.: The role of diagnosis and decision theory in normative reasoning. In Ch. Meyer, J.J., Schobbens, P.Y., eds.: Formal Models of Agents. Volume 1760 of Lecture Notes in Artificial Intelligence. Springer (1999) 216–239