Alternative model-building for the study of socially interactive robots

Meurig Beynon, Antony Harfield, Sunny Chang
*Department of Computer Science, University of Warwick, UK
{wmb,ant,csvmu}@dcs.warwick.ac.uk

Abstract

In this discussion paper, we consider the potential merits of applying an alternative approach to model-building (Empirical Modelling, also known as EM – see http://www.dcs.warwick.ac.uk/modelling) in studying social aspects of human-robot interaction (HRI). The first section of the paper considers issues in modelling for HRI. The second introduces EM principles, outlining their potential application to modelling for HRI and its implications. The final section examines the prospects for applying EM to HRI from a practical perspective with reference to a simple case study and to existing models.

Introduction

The difficulty of dealing effectively with issues relating to social intelligence in the design of robots is widely recognised. In discussing this challenge, Fong et al. (2002) identify two approaches to the design of socially intelligent agents, the "biological" and the "functional". The biological approach aims to draw on understanding of animals and their behaviour to design robots which exhibit similar properties to their biological counterparts. The functional approach only takes the functionality of such robots into account and is not concerned with the mechanisms by which this is achieved. Traditional AI generally takes a functional approach. The biological approach is favoured by those interested in the social sciences and biology.

Whatever the orientation of the robot design, there are major technical and conceptual issues to be addressed in developing robots that are socially responsive. It is implausible that such issues can be resolved by implementing behaviours that are comprehensively pre-specified by abstract analysis of the operating context. Dautenhahn (2004) proposes that the social personality of a robot should grow through a socialisation process similar to that observed in animals such as dogs. Adams et al. (2000) sees robotics as offering "a unique tool for testing models drawn from developmental psychology and cognitive science". His approach to building sophisticated robots incrementally, using the concept of a subsumption architecture (Adams et al., 2000; Brooks, 1991), indicates a possible way in which such a socialisation process might be supported. However, as Dautenhahn (1995) has observed, the role of 'the social factor' in the development of intelligence has been little explored in the 'sciences of the artificial', and we cannot necessarily expect that techniques for building intelligent robots will deal with social aspects. Adaptation to the social environment is likely to be a much more subtle process than adaptation to a physical context, and demands a more intimate interaction between human and automated activities than has been achieved hitherto.

This paper examines the prospects for deploying Empirical Modelling (EM) in HRI research. EM is an unconventional approach to modelling that reflects a fundamental shift in emphasis in the science of computing. In certain respects, this shift echoes Brooks's outlook. EM favours the construction of physical artefacts that in some sense 'embody knowledge', rather than abstract representations based on formal languages. It also promotes an evolutionary and incremental approach: models are initially very simple, but can eventually attain a high level of sophistication. For the present, EM research is not specifically concerned with how learning or other forms of adaptation might take place automatically. The focus of interest is rather on maintaining an intimate connection between the developing model and the modeller's perception and understanding, which grow in parallel throughout the model-building. The model development is sharply distinguished from other approaches by its emphasis on incrementally layering 'perceptions of relations' rather than 'functional behaviours'. In this way, the primary focus is enhancing the robot's capacity to respond to its current situation rather than on extending its current repertoire of behaviours.

1 Issues in modelling for Human Robot Interaction

Traditional techniques for modelling have problematic aspects in the context of robotics. Closed-world models of robot behaviour may appear to give useful insights in the abstract, but the vagaries of the physical world lead to serious discrepancies between real and virtual behaviours. It is such considerations that prompt Brooks to advocate '[using] the world as its own model' (Brooks, 1991). There is little doubt that problems of this nature will always be an issue, but EM is such a radical alternative to traditional modelling approaches (Beynon, 1999, 2003) that there is hope that it can offer new remedies or palliatives.

Our objective is to develop modelling techniques that can be used in direct and live conjunction with researches on actual robots in a laboratory. The aspiration is to make models that are sufficiently subtle to address social aspects of the interaction to some degree. There are many ways in which empirical study of HRI in the laboratory can potentially be combined with experiments in a virtual environment. We might wish to use virtual models to explore experimental scenarios for which empirical data has been derived, or to connect the behaviour of agents in the physical environment directly to that of their avatars in the virtual world. Possible goals might be formulating new hypotheses, making a virtual record of significant interactions observed in the laboratory, or identifying new patterns of robot behaviour to be programmed. For these purposes, models need to be sufficiently authentic that they can guide the programming of robots. Ideally, we would like to be able to direct the modelling activity freely in a variety of different ways, corresponding to different modes of observing the HRI, mixing modes of observation and experiment in real and virtual worlds.

In the context of modelling for HRI, we identify the following issues as particularly significant:

- having an approach to model development that is incremental, admits highly flexible adaptation through human intervention (because social conventions and physical interactions are so subtle and difficult to circumscribe), and is holistic (because, for instance, social conventions about personal space (Hall, 1966) and lower-level concerns such as navigation are inseparably linked).
- developing models that have explanatory power, so as to be able to trace the effects of robot action to their origins, attribute responses to stimuli appropriately and account for the fact that the robot does more than can be specified and represented in propositional terms.
- interrelating human and machine perspectives intimately so as to be able to exploit the qualities of both humans and robots, as is required to program robots to achieve the high degree of attentiveness to context that is demanded in social situations without compromising their capability to act obliviously with superhuman efficiency where appropriate.

Various kinds of relation have a significant impact upon social interaction. These include:

- Spatial relations - An agent's physical location and the surrounding space are likely to affect the behaviour of the agent. Actions in small confined spaces are usually different from those in large open spaces.
- Temporal relations - Time plays a significant role in human behaviour. When time is at a premium humans are likely to perform tasks differently from when they have plenty of time.
- Status relations - The status of human agents affects their interaction and expectations. Interaction with those with whom we are familiar differs from interaction with strangers. Interaction within the working environment, families and cultural contexts is likewise differentiated according to the status of the agents with whom we are interacting.

Taking account of such relations in interaction is something that humans typically learn from experience. On this basis, a most important characteristic in modelling for HRI is a capacity to accommodate learning in a dynamic fashion. This has particular relevance for the prospects of applying EM to HRI because EM proceeds by modelling relations as they are encountered in experience.

2 The Empirical Modelling Approach

The Empirical Modelling approach to HRI will be sketched with reference to the role played by the primary concepts – agents, observables and dependencies – and to the general characteristics of the development of a model as a *construal*.

2.1 Agents and Observables

Empirical Modelling (EM) approaches the construction of a model of a concurrent system from the perspective of an external observer trying to make sense of what is being observed (Beynon et al., 1990). If the task is to make a virtual representation of a physical system, the principles of EM can be seen as similar to identifying the situation within the context of familiar 'scientific' theory, complemented – where there is no such theory to hand – by the application of the 'scientific method'. In this context, the modeller identifies what they perceive to be the principal agents responsible for state change, and develops hypotheses about how their interaction is mediated by observables. This section will introduce EM as it might apply to the scenario of studying the social behaviour of robots, without particular concern for the technical challenges this might present for EM tools and other relevant technologies in their current state of development. Specific models that indicate the present state of EM development in key aspects will be discussed in the next section.

In the HRI laboratory, the most prominent agents are the robots and the humans who interact with them. Within the scope of the EM model, other elements of the situation are typically also construed as agents. For instance, an item of furniture, a door or a pet might be an agent in so far as its presence and current status influences the state change that takes place. If, moreover, there is some abstract activity in process, such as might be informally described as 'the robot is going to collect up the empty wine glasses', this too would naturally be represented by an agent of some kind. Relevant issues to be noted are:

- the concept of an observable is intended to embrace not only what the external observer can directly apprehend, but what the agents within the system are deemed to directly apprehend. For instance, the 'observables' relevant to the robot might include information from its distance sensor along a particular direction, and information about the status of the current task in hand.

- it is generally necessary to take account of the transient nature of observables, so as to reflect the presence or absence of agents in the situation. For instance, when the task of collecting empty wine glasses is accomplished or aborted, the related observables are no longer meaningful.

Because the model-building activity serves an explanatory function, it is appropriate to characterise an EM model as a 'construal' (cf. the extended discussion of this term in (Gooding, 1990)). Note that, in arriving at a construal, the external observer has to project agency that is human-like on to the non-human agents in the situation. For instance, to explain the behaviour of an automatic door, the modeller may postulate an observable by which the door 'perceives' itself as open, and consider the door to be responsible for manipulating its aperture accordingly.

2.2 Dependencies

Agents and observables are complemented by additional features of the situation that are most distinctive of EM – dependencies. A dependency is a relation between changes to observables that pertains in the view of an agent within the system. In effect, there are latent relationships between those things that an agent is deemed to observe, that are 'perceived' by the agent to be linked in change in an indivisible manner. This indivisibility is in general 'relative to the agent', and its status depends upon the nature of the construal. For instance, in some contexts, the activity of 'collecting an empty wine glass' might be viewed by the external observer as an atomic operation that indivisibly reduces the count of empty wine glasses so far accounted for. Where the robot is concerned, on the other hand, such an operation would necessarily involve a highly complex and intricate sequence of sensing and activation steps.

By their nature, the key concepts of EM are defined by experience. What is deemed to be an agent, an observable or a dependency is at all times subject to change through observation and experiment on the part of the modeller (cf. the way in which varieties of agency are seen to be socially constructed in (Dautenhahn, 1998)). The through-and-through empirical nature of these constituents is reflected in the character of the construal itself, which is conceived and developed quite differently from a traditional computer model.

In the first place, there is no notion of a static or comprehensive functional specification of the modeller's construal. The construal itself takes the form of a physical artefact, or set of artefacts, to represent a current situation and understanding on the part of the modeller; it embodies the patterns of agency, dependency and observation that are deemed to pertain in the situation. When a system has been - for certain practical purposes - comprehensively identified and understood, there will be a single unifying artefact that captures all the observables within the modeller's construal and represents the viewpoint and insight of the external observer. In so far as these observables have specific current values, the artefact itself will serve to represent the current state of the system to which it refers (cf. the way that a spreadsheet records the current status of a financial account). The atomic changes of state that are possible in this state will be represented by possible redefinitions to which appropriate observables are subject, whose impact is in general to change the values of several observables simultaneously, and perhaps change the pattern of dependencies itself. In the HRI laboratory scenario, such an atomic change might typically reflect an 'infinitesimal' movement or sensory update on the part of the robot, or a primitive action on the part of a human agent, such as pressing the television remote control. Note that - because of the dependencies - a single action on the part of an agent may update several observables simultaneously (as when pressing the remote switches the television on). There is also the possibility for independent changes of state to occur simultaneously (as when the robot moves, and the human agent presses the remote control at the same time). The modeller can make use of such a construal to trace characteristic system behaviours, though the effect is quite unlike the exercising of statically pre-specified behaviours in a closed-world that is commonplace in conventional computer programming. Suppose for example that the robot is programmed to collect the empty wine glasses, but that at some point during this collection process one of the wine glasses is accidentally smashed into pieces. It then becomes necessary to adapt the parameters of the collection activity to take account of the new situation - something which the modeller should be able to cope with dynamically when exercising a good construal of the situation, but would have had to have been within the explicit scope of a programmed behaviour.

2.3 Developing a construal

As the above discussion highlights, the development of an EM construal is concerned with something less specific than representing any particular set of functionalities. For any complex reactive system, the goal of developing a single unifying artefact to reflect the modeller's comprehensive understanding is a pipe dream. The quality of a construal is contingent upon the degree of familiarity and understanding that the modeller has built up through observation and experiment, typically over an extended period of interaction. The true potential and limitations of EM in concurrent systems modelling are best appreciated by viewing the construal not in some purported final perfected form, but as it evolves in conjunction with the development of the modeller's understanding. In applications such as HRI modelling, it is plausible that this development should ideally accompany the construction of the real environment from its inception, so that the model grows in subtlety and scope in counterpoint with the understanding of the laboratory technicians and experimenters. To conclude this brief overview of EM principles, it will be helpful to outline informally how such an incremental process of construal might take place.

Throughout the development process, the representation of the construal has two aspects: the physical artefact as it is realised on a computer, or more precisely using an appropriate suite of computer-based peripherals (cf. the distinction between a musical instrument and an orchestra), and documentation in the form of a textual description of the agents, observables and dependencies and their interrelationship within the modeller's construal. As will be illustrated below, in our implementation framework, these two ingredients of the construal are respectively associated with a set of *definitive scripts*, and a set of *LSD accounts* of the agents, to be referred to as 'scripts' and 'accounts' in what follows. An LSD account classifies the observables deemed to shape the behaviour of an agent, with reference to how it perceives, acts upon and responds to its environment. To put these ingredients in context, it is quite plausible that, in the HRI scenario, we might have a good grasp of the determinants of the robot behaviour in the abstract, and reasonable models for its behaviour in certain idealised scenarios (e.g. robot motion where the floor is level and the coefficient of friction is uniform, and the lighting conditions are favourable). We may also have reliable knowledge of the characteristics of the physical environment where issues such as the location of furniture and the operation of doors and light switches are concerned. Such information provides the basis for developing several ingredients that contribute to a useful construal. These might include:

- scripts to represent the principal features of the environment in which the robots and human agents interact.
- an account of a robot's behaviour with reference to the observables that are intrinsically associated with it (such as the current status of its sensors, its location and velocity), together with the external observables to which it responds.
- a script to represent a test environment within which idealised motion of a robot can be inves-

tigated experimentally, and interactively adapted through intervention by the modeller.

In this scenario, many more difficult issues remain to be addressed, such as understanding the relationship between what the robot sensors record (e.g. the distance from the nearest object in some direction) and how this needs to be interpreted in context (as in 'the robot is approaching the table on which there is a wine glass'): these will typically require extensive empirical investigation.

By its nature, an EM construal can accommodate partial understanding and support the modeller in gaining further insight. Though there is not typically one unifying script to represent the entire system comprehensively from an objective external observer's perspective, there will be a collection of subscripts associated with those scenarios for which the modeller has sufficiently detailed understanding. As explained in the above discussion, the behaviours that can be exercised using these scripts are open for the modeller to explore and extend in an experimental fashion. What is more, the behavioural interpretation of the construal can be modified by the modeller 'in-the-stream-of-thought'. This is in sharp contrast to modifying the behaviour of a conventional program, which entails terminating execution, changing the specification and attempting to reconstruct what – taking the changed specification into account – can only be an approximation to the original situation of use. It is also conceptually easy to exercise scripts representing independent aspects of the same situation in combination, as is appropriate where understanding of a situation is too partial to support a conventional implementation of behaviour, but significant behaviours can be explored subject to intervention by the modeller. Taking in conjunction, scripts and accounts also serve as a powerful way of communicating understanding between experimenters.

3 Practical Aspects of Empirical Modelling

This section illustrates how EM techniques can be applied in practice. The scenarios considered relate to interactions between humans and robots that might arise in a house environment. They help to indicate how EM might be used to support the development of a robot that exhibits some degree of social awareness. Our illustrative examples draw upon pre-existing EM models of a house environment, and of various activities that give insight into the potential for effective modelling of human and robot interaction.

3.1 Agent-oriented modelling

Though the term is widely and frequently used, the Artificial Intelligence (AI) community has great difficulty in agreeing on a definition for 'agent'. As Wooldridge and Jennings (1994) point out: "This need not necessarily be a problem: after all, if many people are successfully developing interesting and useful applications, then it hardly matters that they do not agree on potentially trivial terminological details.". This point of view is strongly endorsed by EM, where the implementation and interpretation of a specific pattern of activity that is conceptually associated with one and the same agent evolves with the model. In a typical pattern of model evolution, a pattern of behaviour that is initially carried out by a human agent can be progressively more comprehensively executed automatically, so that eventually it can be exercised without – or more precisely, in what seem to be the only realistic circumstances, without – the intervention of the human agent. What adds particular force to Wooldridge's observation in this context is that it is not appropriate in EM to conceive the evolution of a model in terms of a discrete succession of progressively more expressive models, each with its own distinctive functionality. In so far as it makes sense to speak of the identity of an EM model, it is more appropriate to think of this identity as unchanging throughout its development, in the same spirit in which we say that 'the child is father to the man'.

By way of illustration, consider the situation where a robot has to negotiate a corridor in which there is a person walking towards it. This situation is encountered by millions of people everyday as they walk down corridors, paths and streets. Because avoiding someone while walking is something we do with relative ease, it is easy to take it for granted. However, the factors affecting this behaviour are quite complex and reproducing this behaviour in a model is a non-trivial task. In applying EM in this context, it is initially appropriate to think about the robot's actions with reference to how a human agent with the same capacity to sense and react to its environment as the robot might respond. As the modeller's understanding of the issues involved matures, it will become possible to automate the more routine aspects of this response. For instance, the forward motion of the robot along the corridor could be automated, and only its lateral movement could be under the control of the human developer. Typically, successful negotiation of the corridor may be automatable subject to reasonable assumptions about the behaviour of the approaching person, or 'opponent'. There may be no satisfactory strategy if the opponent is malicious and sets out to obstruct the robot's passage. Even where the opponent is benign, there may still be exceptional circumstances in which the familiar parallel side-stepping behaviour arises, when the robot's forward motion may need to be suspended. To overcome this problem, which arises at a rather advanced stage in the modelling, it is in general necessary to combine automation of routine aspects of the robot behaviour with mechanisms for open-ended human intervention when singular scenarios arise. Only when these singular scenarios are understood in sufficient detail does full automation become possible. In the transition from an initial model in which the state change for collision avoidance is predominantly supplied by the modeller to a final model in which this state change can be carried out autonomously by a programmed agent, the nature of the agent effecting the state change evolves in ways that are liable to subvert any but the weakest notion of agency. This is in keeping with the observation by Lind (2000) that, in agent-oriented software engineering, "the conceptual integrity that is achieved by viewing every intentional entity in the system as an agent leads to a much clearer system design".

Our illustrative example can be further elaborated with reference to specific practical tools that support EM. To enable the developer to act in the role of the robot, it is first helpful to give an LSD account of the robot's relationship to its environment (cf. section 2.3). This involves classifying the observables that affect the behaviour of the robot as an agent. Projecting ourselves into the role of the agent, there are some observations that the agent can make about the environment – these determine the observables that are *oracles* to the agent. We might assume, for instance, that the robot agent has sufficient 'visual' capability to be able to identify other agents or static objects, to locate the positions of the other agents that are within the field of vision, and to determine in which direction the other agents are moving (the *state* observables of these agents). We can further suppose that the robot agent has conditionally control over certain observables (its *handles*), and that there are certain dependencies between observables that can be deemed to apply in the view of the agent (its *derivates*). It is then possible to describe simple strategies that a robot might employ with reference to the LSD classification of observables. For instance, one simple avoidance strategy is: *if an agent is in the direction that one is walking then take a step sideways*. This might be captured in an LSD account as shown in Figure 1.

As discussed in section 2.3, there are two aspects

```
agent SimpleAvoidingAgent {
   states
   //observables belonging to the agent
      position_x, position_y,
      direction, potential_collision
   handles
   //observables that the agent controls
      position_x, position_y, direction
   oracles
   //external observables the agent responds to
      opponent_position_x, opponent_position_y,
      opponent_direction
   derivates
   //dependency between observables
      potential_collision =
         (position_x == opponent_position_x) &&
         (direction != opponent_direction)
   protocol
   //trigger actions
      potential_collision -> position_x++,
      ! potential_collision ->
         position_y = position_y + direction
}
```

Figure 1: A simple example of an LSD account. The derivate *potential_collision* highlights the situation where a collision may occur and the protocol specifies a change in *position_x* aimed at avoiding a collision.

to the development of a construal in EM: the construction of a physical artefact on the computer, and the associated documentation of the modeller's construal. The physical artefact is a source of experience for the modeller that metaphorically represents perceptions of the environment by a whole range of agents. Figure 2 for example, is a snapshot from an EM model of collision avoidance developed by Warwick student Chris Martin in his final year project in 2003-4 (see Figure 2). The geometric elements of the figure are lines and circles that represent the paths traced by two agents, their fields of vision and current locations and headings. The perspective associated with the model is that of an external observer monitoring the behaviour of two people passing each other in a corridor, as if viewed from above. Our EM tools are such that this model could in principle be run in a distributed fashion in conjunction with variants of the model that represent the corridor interaction from the perspectives of the agents themselves. This allows the modeller to investigate through experiment how the roles of agents can be played by human agents or automated.

Martin's model embodies a construal of collision avoidance more sophisticated than that documented in Figure 1. The model was developed to explore how human agents manage collision avoidance, and hence involves a richer construal of visual activity, taking account of the idea that it is only possible to look in one direction at once, and that the eye is only sensi-

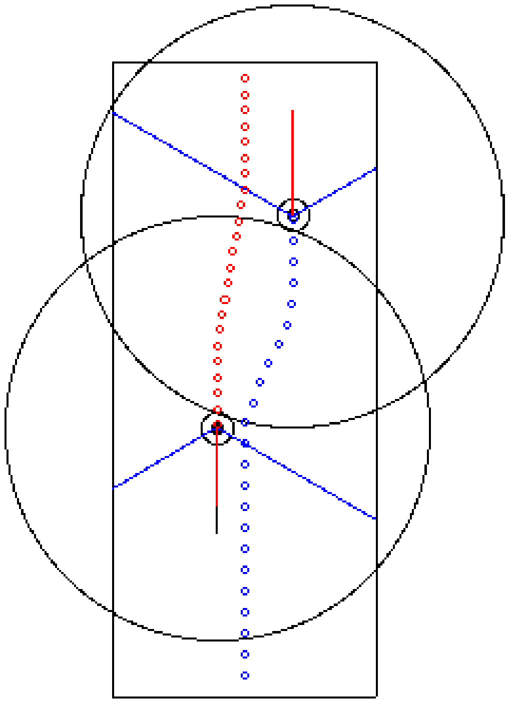

Figure 2: Two agents successfully avoiding a collision in a corridor.

tive within 80 degrees of the direction of looking. Because the modeller's construal is itself to some degree tacit in interaction with the model (cf. Gooding's observation that a construal must be viewed in conjunction with its associated body of ostensive practices (Gooding, 1990)), it is difficult to appreciate Martin's model fully without being able to consult him as the model-builder, or to have a dialogue with him about his interaction with the model. An LSD account is a useful adjunct to the computer model that helps to expose the most prominent meaningful interactions with the model. In practice, there is typically much interesting and significant interaction with and within a model that cannot be explicitly captured in an LSD account. For instance, the collision avoidance strategies used in the most advanced variants of Martin's model were never explicitly described by an LSD account, and involve spatial and temporal considerations that are too subtle to be conveniently specified in an abstract protocol in isolation from the model.

The above discussion illuminates the context for the development of EM artefacts and LSD accounts in HRI. Model construction and the elaboration of LSD accounts are symbiotic processes that do not follow a preconceived pattern, but are mutually supportive. Models and accounts can relate to many different perspectives on agency and modes of observation and construal. Artefact and documentation develop together, and serve complementary purposes both private to the modeller and in relation to the communication of construals.

The first objective in applying EM to HRI would be to better understand how human capabilities and behaviours and robot capabilities and behaviours can be most effectively concurrently elaborated and integrated. As has been illustrated, EM can help us to explore the factors that are significant in determining human behaviour in relation to such tasks as collision avoidance. It can also enable us to construct idealised prototype behaviours that are expressed in terms of high-level abstract observables that serve as useful reference models for devising and analysing robot behaviour. A more ambitious goal involves demonstrating that EM can be used in programming robots. A key aspect of this might involve implementing the `SimpleAvoidingAgent` model with reference to a more primitive and explicit account of the vision capability of an actual robot, through progressively elaborating its states, oracles, handles and protocol. It is in this connection that the usefulness of models and accounts that are intimately related and synchronised is most evident.

It is through developing and experimenting with models based on such construals that the modeller will be able to recognise and address more subtle features of problems of HRI. For instance, by playing out the role of a robot agent in collision avoidance, the modeller will be able to highlight the impact of spatial, temporal and status relations in the interaction. If the person walking towards you is elderly or infirm then it is appropriate to move out of their way so that they are inconvenienced as little as possible. If time is critical (as when there is a fire in the building) then observing social distances will be less of a priority than getting to the fire exit as quickly as possible. Our prior experience suggests that, provided our underlying construals of the more prosaic aspects of avoidance behaviour have been developed with due regard for EM principles and concepts, it will be possible to adapt models to reflect more sophisticated behavioural issues in social interaction. A key factor in this is the well-conceived application of modelling with dependency.

3.2 Modelling using dependency

Dependency is one of the main concepts underlying model-building using EM. Dependencies reflect relationships between characteristics and perceptions of objects that are always maintained. Dependency arises commonly in mechanical systems, where a change to one component directly affects another component in a predictable and indivisible manner. There is no context in which the state of one component is not synchronised with that of a related component.

Dependency maintenance is one of the central characteristics of the software tools that we have developed for EM. Our primary modelling tool supplies notations within which scripts of definitions can be formulated to express dependencies between the many different kinds of observables that determine the various aspects of the state of an EM artefact (see, for instance, the discussion of modelling situated, explicit, mental and internal aspects of state in (Beynon et al., 2001)). The simple illustrative example used in this section makes use of elements from one such model, originally developed by the third author in her final year project in modelling an intelligent house environment. An important feature of EM, to be elaborated in the next section, is the scope it offers for models to be re-used for different purposes, and for relatively complex models to be built up incrementally through assembling and combining simpler components.

Dependency plays a key role in all forms of human-robot interaction. With reference to each agent, there is a dependency between what is observed and what is inferred. With reference to an agent in its environment, there is a dependency between what exists and what is observed. In EM, models of environments are built up from observables and dependency. In modelling a house, for instance, the position of a lamp on a table is dependent on the position of the table: if a person moves the table then the lamp also moves, but not *vice versa*. The illumination of the room is dependent on the position of the lamp and also the position of other objects in the room. If a person or robot is obstructing the lamp then it will affect the illumination of the room with potentially undesirable effects. A socially sensitive robot will need to take account of these dependencies.

By way of illustration, consider the dependency involved in a living room, where there are likely to be people watching television. Clearly, it would be undesirable for a robot to obstruct someone while they are watching television. As in modelling a potential collision in the corridor (cf. the derivate in Figure 1), we can represent a potential obstruction by devising a system of dependencies. If we work with a 2D model of the living room such as is depicted in Fig-

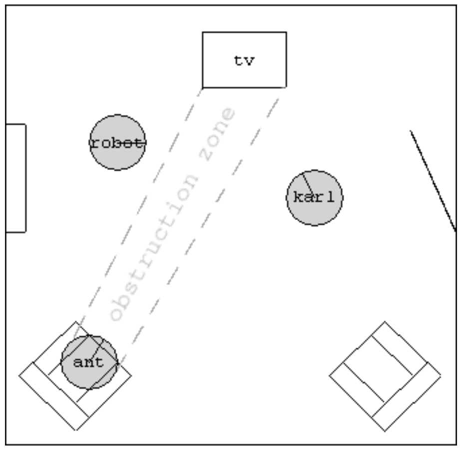

Figure 3: The living room model. Whether the robot causes an obstruction is dependent on the position of the people and the television.

ure 3 then we can identify certain areas of the room where the presence of a robot agent will cause an obstruction. Using dependency, these areas can be defined in terms of the position of other agents and the television, so that they change dynamically as agents move around. Other issues might also effect whether there is an obstruction. If the television is switched off then the robot can be fairly sure that it is not being watched. The obstruction is then dependent on: the robot being inside the area between the people and the television, and the television being switched on.

The way in which these dependencies can be directly modelled using EM models is further illustrated in Figure 4, which comprises some key definitions drawn from the underlying model depicted in Figure 3.

When model building with dependency, we can explore the effects of altering observables which may have meaning in the environment. For instance, different people might have different sensitivities about how much space is unoccupied in the visual field around a television. This would mean that the possible obstruction areas would differ according to who was watching. The dependency in the model would make it possible to adapt the model without making any changes to our models of the living room environment or the robot.

The use of dependency in EM is much more significant than the mere introduction of an additional programming construct to a traditional programming language. Appropriately used, dependency serves to support the development of models that stand in a very different relation to interaction in the external

```
tv_is_on = true
tv_position = {16,28}
tv_left = tv_position - {3,0}
tv_right = tv_position + {3,0}

chair1_position = {5,5}
chair1_occupied = true

chair2_position = {27,5}
chair2_occupied = false

robot_position = {10,20}

robot_is_obstructing_chair1 =
        tv_is_on &&
        chair1_occupied &&
        insidetriangle( robot_position,
        chair1_position, tv_left, tv_right)

robot_is_obstructing =
        robot_is_obstructing_chair1 ||
        robot_is_obstruction_chair2
```

Figure 4: An extract of definitive script showing that an obstruction is dependent on the positions and status of other agents in the model.

world from traditional programs. The notion of 'construal' is categorically different in character from the idea of a program that is based on a functional specification and optimised to meet its specific functional objectives. This has significant implications for the way in which EM artefacts can be developed and combined.

3.3 Evolving the model

In conventional software development methods it is common for a specification to be formalised before any design or implementation has begun. EM in contrast is of its essence concerned with development that is incremental, at least where the modeller's conception of the artefact is concerned. That is to say, even if the modeller incorporates some pre-existing EM artefact in the development, as has been quite common in practice, the comprehensive understanding of the artefact that may be required to exploit it fully in the new context normally involves a corroborative process of interaction with the artefact and the external world that is similar in nature if not in scale to the interactions involved in its original construction. This corroborative activity is not an all-or-nothing exercise in comprehension such as is typically demanded of the programmer confronted with a formal specification for a requirement or a design, but an active process of becoming familiar through changing the state of the EM artefact in an experimental fashion. This is because a construal only serves its function subject to the engagement of the human interpreter, whether or not the interpreter was also responsible for its development.

In building an EM artefact from scratch, the model-builder takes experimental interaction a step further than simply experiment-for-confirmation. The model-building is exploratory: it is an exploration in the creation of a model, where there is a place for blind experiment-for-discovery. The model-building can begin with little knowledge of what a final model might embody. It is the job of the modeller to develop understanding through exploration of the model; at all times acquiring knowledge and insight in constant connection with the model. This activity of model-building establishes an intimate relation between the artefact itself and the mental model of the modeller, as expressed in terms of expectations about possible states of the artefact, and reliable patterns of interaction with it.

The EM environment goes some way to providing the exploratory power needed to bring the model into close alignment with the modeller's construal of a situation. The interactive nature of the environment enables the modeller to incrementally build artefacts and observe their effects on-the-fly. Some characteristic features of EM can be described with reference to illustrative examples.

Consider a possible development of the living room environment discussed previously. Suppose that we introduce more agents, including one intending to move from one side of the living room to the other – perhaps to reach the cocktail cabinet on the far side of the room. The agent will have to observe the avoidance zones in the living room by exploiting dependency, and also avoid oncoming agents that may be moving across the room. One way of building a model to represent this situation is to combine the living room model and the corridor model, and explore the effects of this conjunction. The result of combining two small models with relatively simple actions is a model with a more complex behaviour.

By evolving a model in this way, incrementally building new artefacts and combining them with existing artefacts, is becomes possible to observe new phenomena and gain insight into more complex behaviours.

The use of dependencies also enables other forms of direct extension of models. Since the EM environment provides a notation for 3D graphics, the modeller might consider extending the 2D model into a 3D model of the living room. This involves writing dependencies to link the positions of objects in the 3D model to their point locations in the 2D model. This kind of model extension can be developed on-the-fly in a exploratory manner.

It is important to note that EM models never reach a "final" state where the implementation is complete: they are continually refined and exercised in new ways through interaction on the part of many different agents (e.g. in the role of designer, observer, or interaction partner). That modelling social interaction should have this open-ended nature is completely plausible. As we do not fully understand the nature of social conventions (Gilbert, 1995) – even our own – it is unlikely that we will ever want to finalise (or completely formalise) a behavioural model.

It is natural for readers unfamiliar with EM thinking and practice to question whether our discussion of applying EM principles to HRI engages sufficiently with the hard problems that are the primary focus of the call for papers. The modest content and conservative themes that are represented in our illustrative examples may suggest a lack of ambition that is out of keeping with our pretensions to an 'alternative model-building' approach. Whilst it is true that our research on applying EM principles to HRI is as yet in its earliest stages, and that far more investment is required to evaluate its true potential, we are optimistic about the prospects of fruitful results in the long term. The same cultural influences that associate computation so strongly with specification and optimisation also often lead us to think of difficulty primarily in terms of problems that can be explicitly framed and whose solution we hope to address by dedicated directed effort that is informed – and in some respects limited – by specific goals. In this way, we come to attach great value to targeted specific techniques and solutions that take us beyond the commonplace territory of a problem domain, whether or not they can be integrated with other solutions of a similar nature, or usefully related to the more mundane regions of the problem space. This is not a concept of difficulty that is well-suited to interpreting our aspirations for EM.

To put the ambitions of EM in perspective, it is useful to contrast *having powerful algorithms to solve specific technical problems in a domain*, and *having a powerful construal of the key phenomena in a domain*. Gaining the latter is invariably a matter of acquiring a large body of experience – even when this experience is guided (as in an established science) by an advanced and comprehensive theory. Since EM is primarily concerned with using the computer to support the development of construals, rather than to implement sophisticated algorithms, it is unsurprising that EM has found broad application to many fields, but has yet to contribute conspicuous specific applications to any one. Similar considerations apply at a different level of abstraction when considering the relationship between ingenious solutions to specific problems in HRI and ways of thinking about the domain that can promote a general understanding and an integration of what may appear to be separate concerns.

The above discussion informs our orientation towards applying EM to problem-solving in HRI. Hard problems often come into being because our solutions to the easier problems are too tightly constrained. This is frequently the result of making the simplifications in our models that are necessary to generate solutions that are sufficiently efficient in execution or ingenious in conception to attract attention. Any interaction involving a human will inevitably involve a complex model of activity. Exploratory model-building provides a method with which we can start our model-building on a small scale and incrementally extend the model to ever increasing complexity. In this context, the hard problem is to integrate the solutions to relatively easy problems without losing conceptual control. This is intimately connected with what this paper highlights as perhaps the most significant hard problem in modelling: giving dynamic support for situated learning.

Acknowledgements

We are much indebted to Kerstin Dautenhahn and Chrystopher Nehaniv for introducing us to their research on socially interactive robots, and to Steve Russ and Chris Martin for their work on applying EM to crowd behaviour.

References

B. Adams, C. Breazeal, R. Brooks, and B. Scassellati. Humanoid robots: A new kind of tool. *IEEE Intelligent Systems*, 15(4):25–31, 2000.

W.M. Beynon. Empirical Modelling and the foundations of AI. *Computation for Metaphors, Analogy and Agents, Lecture Notes in Artificial Intelligence,* 1562:322–364, 1999.

W.M. Beynon. Radical Empiricism, Empirical Modelling and the nature of knowing. *In Proceedings of the WM 2003 Workshop on Knowledge Management and Philosophy, Luzern, April 3-4*, 2003.

W.M. Beynon, M.T. Norris, R.A. Orr, and M.D. Slade. Definitive specification of concurrent systems. *Proc UKIT'90 IEE Conference Publications,* 316:52–57, 1990.

W.M. Beynon, C. Roe, A.T. Ward, and K.T.A. Wong. Interactive situation models for cognitive aspects of user-artefact interaction. *In Proceedings of Cognitive Technology: Instruments of Mind (CT2001)*, pages 356–372, 2001.

R. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

K. Dautenhahn. Getting to know each other—artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems*, 16:333–356, 1995.

K. Dautenhahn. The art of designing socially intelligent agents: science, fiction and the human in the loop. *Applied Artificial Intelligence Journal*, 12(7-8):573–617, 1998.

K. Dautenhahn. Robots we like to live with?! - a developmental perspective on a personalized, life-long robot companion. *in Proc IEEE Ro-man 2004, Kurashiki, Okayama, Japan (invited paper)*, 2004.

T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42, 2002.

N. Gilbert. Emergence in social simulation. *in N. Gilbert, and R. Conte (eds), Artificial Societies, London: UCL Press*, pages 144–156, 1995.

D. Gooding. *Experiment and the Making of Meaning*. Kluwer, 1990.

E.T. Hall. *The Hidden Dimension*. Doubleday, 1966.

J. Lind. Issues in agent-oriented software engineering. In *Agent-Oriented Software Engineering*, pages 45–58, 2000.

M. Wooldridge and N.R. Jennings. Intelligent agents: Theory and practice. *Knowledge Engineering Review*, 10(2):115–152, 1994.