Intuition and Observation in the Design of Multi Agent Systems

Scott Moss

Centre for Policy Modelling
Manchester Metropolitan University Business School
Manchester M1 3GH, United Kingdom
+44 (0)161 247 3886

s.moss@mmu.ac.uk

ABSTRACT

Both formal analysis in the sense of proving theorems about the properties of agent and mechanism design and the use of formalisms as representation languages have been central elements in the foundation of multi agent systems research. The choice and frequently the development of formalisms for the specification and description of multi agent systems has been guided by intuition regarding the importance and nature of such concepts as belief and intention. An alternative to this foundational approach is a representational approach developed by modellers of observed social systems who design agents and mechanisms to capture observed behaviour and modes of social interaction. While the foundational approach has had an important influence on the research agenda of agent based social simulation, the representational techniques of agent based social simulation modellers have had no discernable influence on formalistic approaches to software engineering for multi agent systems. The purpose of this paper is to define a means of making available the lessons of real social systems to adopting formal approaches to MAS design. The means employed turns on the development of a canonical model capturing features of an observed social system in a way that relates explicitly to concepts such as belief, desire, intention, commitment, norms, obligation and responsibility. As a result, it is possible to define these concepts with minimal ambiguity either as an alternative to the use of formalisms as representation languages or as a bridge to such formalisms.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: coherehnce and coordination; multi agent systems

General Terms

Management, Design, Theory, Verification.

Keywords

Validation, Agent Based Social Simulation.

1. INTRODUCTION

The standard approach to agent and mechanism design for multi agent systems is heavily influenced by issues of verification or quasi-verification and hardly at all by validation.

By quasi-verification I mean the common practice of describing design in terms of a mathematical or logical formalism when the design itself is too rich to allow for formal proofs of its properties. This is the consequence of a tension between validation issues – the intention to demonstrate that software with the stated design characteristics will do something useful – and verification issues. The virtue of quasi-verification is that the design aspects are stated unambiguously even if there are no proofs of the properties of the design. The downside is that, by restricting the design features so that they can be expressed in relation to a particular formalism, the scalability and scope of software systems with the specified design characteristics may be severely restricted.

Wooldridge [27], for example, argues with regard to the use of formal logics for agent design that "giving anything like a complete account of the relationships between an agent's mental states is extremely difficult…. In attempting to develop formal theories of such notions [as beliefs, desires and the like], we are forced to rely very much on our intuitions about them. As a consequence, such theories are hard to validate (or invalidate) in the way that good scientific theories should be. Fortunately , we have powerful tools available to help in our investigation. Mathematical logic allows us to represent our theories in a transparent, readable form…." He goes on to state that formal proofs generate predictions of the theory so that we can "see whether or not their consequences make sense." However, in the final chapter of the book from which the above quotations are taken, Wooldridge summarises the difficulties of verification either axiomatically or semantically. In the particular case of BDI models, "there is no clear relation between the BDI logic and the concrete computational models used to implement agents [and] it is not clear how such a model could be derived."

Because no formalism has any sort of objective precedence over any other, the choice of formalism for purposes of verification is chosen on the basis of a pre-theoretic belief that a formalism with particular properties is appropriate to the type of program to be verified. The development of BDI logics from Bratman's [4] argument that beliefs, desires and intentions should be formally consistent to the inclusion of commitment by Cohen and Levesque [6] to the Rao-Georgeff [23] specification of the BDI architecture all rely on formal specifications of intuitive claims about the nature of goals and planning by humans. Further issues such as norms, trust, interests, commitment, obligation and responsibility are frequently expressed in terms of BDI and similar (*e.g.*, deontic) formalisms. [9]

The verification problem is by no means restricted to the use of BDI or other formal logics. For example, Jennings, *et al* [11], in reviewing the literature on automated negotiation by agents note that formal properties of game theory are proved only for highly specialised strategies and require costless or no computation to find solutions acceptable to all negotiators

(Nash equilibria). Heuristic approaches are, of course, just that – they have no formal basis. Argument based approaches using formal logics to resolve contradictory statements suffer the same limitations as those identified by Wooldridge.

It would, of course, be completely misleading to suggest that verification never happens. In the planning literature, for example, Pollack's [22] classic 1990 paper on plans as mental models uses Allen's [1] interval based temporal logic only as a *representation language*. But just a few years later, we get to Grosz's and Kraus' papers on SharedPlans, *e.g.* [10], report proofs of relevant theorems relating belief to capability. Even these impressive results and demonstrations of progress are applied to relatively simple problems. In the cited Grosz-Kraus paper, for example, the test problem concerns two individuals planning a dinner party with 14 tasks. In seeking to address larger scale problems with many more agents and tasks – 60 agents and 50 tasks – these authors [24] turned to simulation modelling with no apparent formalism as representation language.

In general, full verification is applicable only to multi agent systems that are highly restricted in terms of the complexity of their agent and mechanism designs. Quasi-verification – the use of formalisms as representation or specification languages – is naturally less restrictive than axiomatic or semantic verification but clearly has not supported implementations of large scale multi agent systems. Simulation models are in effect larger (though by no means very large) implementations of multi agent systems but these lack the clarity and precision of the verified and quasi-verified systems.

An important question facing the agents research community turns on the respective roles of formal and simulation analysis. There is already a substantial gulf between the tidy multi agent systems amenable to verification and quasi-verification on the one hand and, on the other, the messy multi agent systems used for simulation-based analysis of agent and mechanism design.

The purpose of this paper is to investigate whether and how simulation analysis can complement the use of formalisms by providing a clear target and replacing intuition with a more objective and perhaps effective means of formulating agent and mechanism designs than intuition as described by Wooldridge.

2. OBSERVATION AND INTUITION

The intuition driving formalist specification of agent theory must, at some level, be guided by observation and experience. To rely on an entirely pretheoretic intuition to determine the concerns of the theory – beliefs, desires, intensions, trust, and so on – but, at the same time, to insist on as much rigour as possible in the agent design is curious. Presumably, the reason for relying on these and other aspects of mental states in designing agents is that they support an analogy with successful human behaviour. Indeed, examples from the MAS planning literature [10, 22] clearly take human capabilities as appropriate analogies for essential agent capabilities. If individual human characteristics are thought to be a good guide to agent design, then the characteristics and adaptability of social interaction should be thought to be an effective guide to mechanism design. And, rather than to rely on introspection and armchair theorising, agent and mechanism design might better be based on sound observation of human behaviour and social interaction in environments that are known to capture important aspects of agents' societies.

The description of social observation by means of multi agent systems is one of the roles of agent based social simulation (ABSS). ABSS encompasses two separate approaches: the foundational and the representational. Foundational ABSS is exemplified by the work of Castelfranchi and Conte, *e.g.* [7] who employ quasi-verification to explore possible foundations for a new social theory that would inform both agent and mechanism design and the analysis of real social systems. Representational ABSS is the use of multi agent systems to describe observed social systems or aspects thereof and to capture the sometimes conflicting perceptions of the social system by stakeholders and other domain experts. While foundational ABSS is a well established and respected area of work within the computer science end of the MAS research community, representational ABSS has had less influence. In a sense, this paper is a manifesto for the greater recognition of the role of representational ABSS in software design since representational ABSS can inform, systematise and strengthen the intuition that is anyway required to formulate the theories expressed as logical or mathematical (*e.g.*, game theoretic) formalisms in agent and mechanism design.

The process by means of which representational ABSS can be used to inform agent and mechanism design will have to be based on well validated social models. Clearly, the more ways in which such models are validated, the more confidence we can have that they are accurate representations of the individual behaviour and social processes that will inform our intuition as software designers. One form of validation can involve the comparison of statistical signatures of the software system and the target social system. Some early work on the use of statistical signatures to distinguish between the goodness of representation of different models was due to economists at the Santa Fe Institute who developed an artificial stock market to identify key features of actual behaviour that lead to observed clustering of price and volume volatility. A recent example of this work is [14]. A further development in the validation of representative ABSS systems is being explored in relation to policy analysis for the consequences of climate change and sustainable resource (particularly water) management. By involving stakeholders actively in the modelling process, the agent designs are validated as good descriptions of specific target social entities – individuals or (say) organisations. Similarly, the mechanism design is validated on the basis of its accuracy as description of social interactions among stakeholders.

From the point of view of the software engineer, representational social simulation models of an individual social process can certainly inform intuition or suggest new approaches to agent and mechanism design. However, it is difficult in any detailed case to distinguish between properties of the system that have some special function in making that particular system effective and robust and properties that would support effective action and interaction more generally. Indeed, it might be useful to identify what it is that makes some special arrangements useful and, to inform the development of truly adaptive systems, how those special arrangements emerged in their particular social context.

To this end, it has been found useful in the social simulation literature to devise "canonical models" that capture relatively abstract representations of features of real social systems. Sometimes the models are used to identify phenomena that have *not* been captured in more verbal and intuitive analyses of the target social system [8]. In other cases, the models are used to identify subsumption relationships among apparently quite different models.[19] Both of these roles of canonical models usefully inform intuition in the design of agents, their modes of interaction and the norms that constrain both. An important feature of both uses of canonical models is that they support the analysis of the conditions in which the social simulation models are applicable. By analogy, the use of such canonical models will support the analysis of the conditions in which different agent and mechanism designs are appropriately incorporated into multi agent software systems.

3. Canonical models: an example

Negotiation is a classic and difficult problem in multi agent system research. Some negotiations can take place via mediators [26] but others are both multilateral and direct. In fact, most negotiations are direct with only auctions and some unique and difficult negotiations taking place through intermediaries. In the case of the unique and difficult negotiations – the Middle Eastern and Northern Ireland peace processes, for example – the essential infrastructure for the mediation is difficult to establish. In cases where difficult negotiations are undertaken repeatedly, as in labour contract negotiations between stable unions and managements, there is in many countries some recourse available to independent mediators with a corresponding infrastructure to provide the mediation. But such mediation infrastructure – apart from auctions – is by no means the norm. For this reason, the example to be developed here will concern only direct negotiation.

In some negotiating environments, the actions eventually taken by one party will have no effect on the state of the environment or therefore on the actions or abilities to satisfy the goals of the other parties to the negotiation. Datamining by agents is an obvious example since the acquisition of information by one agent does not *ipso facto* reduce the ability of other agents to acquire the same information. It is less clear that, for example, supply chain negotiations share this property. The sale of inputs to the manufacturer by one supplier will surely influence the ability of another agent to supply the same inputs. Changes in the production targets of one agent in the supply chain will also influence the goals and scope for action of other agents both upstream and downstream. The example developed here concerns negotiations in which actions and goals cannot be independent.

The environment for this model is based on a detailed qualitative and hydrological study of the stakeholders concerned with water supply, use and management in the Limberg basin of the River Meuse. Considered as a social feature, water is pure conflict. It is used for a wide variety of purposes and, without incurring substantial costs and maintaining substantial infrastructures, water cannot be reused. The water management issues in the Limberg basin relate not only to the usual water quality and quantity of supply issues, but also to river navigation, flood control, environmental protection, ground water extraction for agricultural use and other issues. Flood control can take several forms: dykes to contain the water (until it gets further downstream) and the use of floodplains. Floodplains prevent, or require the evacuation from, housing and they also support a different type of flora and fauna. Agriculture can exist in the floodplains but bear the risk of occasional losses from flooding. One scheme for the river is to deepen it to support a larger volume of shipping. The river is deepened by extracting gravel from the riverbed and the sale of the gravel reimburses the gravel extractors. The stakeholders are several ministries of the national government of the Netherlands, the Limberg provincial government, citizens' groups, farmers' groups, NGOs including a range of "greens", the gravel extraction companies and navigation companies.

The goals of these stakeholders are not fixed and constant. The private gravel extraction and navigation companies are of course interested in profits and returns on their investments. The provincial government favours the establishment of floodplains over dykes because of concerns that a dyke failure is calamitous in terms of loss of life and property while floodplains involve much less risk. On the other hand, the capital cost of dykes is very much less than the cost of establishing floodplains and does not involve the relocation of existing communities. The provincial government favours floodplains and the central government favours dykes. However, after each of two major floods in the 1990s, the importance of flood control was heightened for all stakeholders though the concerns became less intense with the passage of time. At least in the Limberg region, there is evidence from stakeholders that the relative importance of different goals is determined for each stakeholder to some extent by recent experience.

The different stakeholders in the Limberg region do not share the same beliefs regarding either the current state of their environment or the actions available to them or the consequences of those actions. Frequently, they do not consider the same issues which makes it difficult in negotiation for each stakeholder to understand the interests determining the positions of the other stakeholders.

All of this takes place in the context of an environment in which there are unpredictable clusters of extreme events – principally peak discharges of water down the Meuse from the Rhine as well as its own catchment.

The above account of the issues in the Limberg basin has been derived from domain experts in the FIRMA project: Freshwater Integrated Resource Management with Agents.[1] The purpose of that project is to develop tools for policy analysis using agent based social simulation modelling. One aspect of this work involves capturing the perceptions of individual stakeholders regarding the behaviour of themselves and other stakeholders as well as their perceptions of the integrated physical/social system and the ways in which the stakeholders do and can interact and the consequences of different modes of interaction.

Of interest to the multi agent systems community more generally will be the complementary involvement of both representational and foundational ABSS. The representational modelling in this project is intended to incorporate and assess the role and importance of concepts generally considered by means of the more formalistic approaches described by Woodridge: beliefs, desire, intensions, interests, norms, and so on. In effect, observation and representational ABSS is being used not only to inform but also to evaluate the intuition on which fundationalists rely to develop formal theories of these concepts.

4. IMPLEMENTATION ISSUES

In order to achieve coherence between the formalism based concepts and the representational models, the choice of abstraction must capture the essential features of shifting goals, the conflict inherent in the consequences of actions by different individuals and the devices developed and used by

[1] I am grateful for this account to our colleagues in the International Centre for Integrated Studies in the University of Maastricht, especially Jan Rotmans, Anne van der Veen and Jeorg Krywkow.

socially situated individuals to reconcile goal conflict with the need to pursue essential activities in the face of considerable uncertainty. The research plan is to engage in an interplay between concrete representation models and abstract canonical models. In the this section, the canonical model is described, followed in the next section by some simulation results obtained with that model and an exploration of the means by which that model and extensions of it can be used to integrate observation and intuition and, therefore, validation and verification or quasi-verification.

The canonical environment model is a variant of the sandpile model used in statistical mechanics to generate self organised critical processes.[3] Such processes yield clusters of extreme events of unpredictable magnitude and at unpredictable time intervals. Both the time pattern and cross-sectional data generated by simulations of self organised critical processes carry the same statistical signature as do many natural and social phenomena such as earthquakes, avalanches, sunspots, river bifurcations, species extinctions, traffic jams, financial market prices and volumes, sales values and volumes in fast moving consumer goods markets, personal income distributions, city sizes and many more such phenomena. [3, 15-17, 20]

Although there are hardly any analytical results on self organised critical systems and processes, simulation results are extensive and consistent.[12] Unpredictable clusters of extreme events and leptokurtic frequency distributions of event magnitude occur when there is a system of densely interacting agents or other types of component, where the behaviour of those components is metastable (stable below some threshold stimulus), where the system is dissipative (in agent terms, the agents are influenced by, but do not imitate, other agents) and the system is not dominated by exogenous inputs.

Most self organised critical systems are simulated on a grid with “sand grains” being dropped into random cells of the grid. When the number of grains in a cell reaches some critical level, the sand “topples”, a phenomenon represented by the reallocation of some of the grains in the cell to other cells. As more and more of these cells approach their critical values, it becomes increasingly likely that a toppling from one cell will cause other cells to reach their critical values and topple as well. The number of these topplings at each time step is what has the same statistical signature as the natural and social phenomena mentioned above.

The difference between the usual sand pile model and the representation of the environment in the model reported here, is that in the present model the sand is dropped onto a network that can be anything from a ring lattice to a small world network to a random network with any chosen degree of connectivity. Watt’s [25] β-graph algorithm is used to generate the instantiations of this environment. The vertices of the environment correspond to the cells of the more conventional sandpile models. When the contents of a vertex topple, the “sand grains” are distributed individually and at random to neighbouring vertices. Depending on the parameters of the network generating algorithm, there can be clusters of neighbours with sparse links between the clusters or a uniform distribution of random links among the vertices.

The addition of grains of sand to the vertices represents the actions of the agents in the model. Consequently, the actions of the agents can generate changes in the values of the vertices in the same way that the actions of speculators in the financial and organised commodity markets can generate waves of activity and unpredictable clusters of movements in prices and volumes.

The virtue of Watt’s β-graph algorithm in this context is that it permits us to experiment with the structure of the underlying relations. In the Limberg region, for example, there appear to be distinct clusters of goals, actions and the effects of actions on various aspects of the physical and social environment. Though evidence is required, a plausible hypothesis is that agents engaged in competitive activity in cyberspace will find that there are spheres of influence that are loosely linked to other such spheres.

The relevant parameters in the graph generating algorithm are the number of vertices (*n*), the number of edges from each vertex (*k*) and the rewiring probability (*p*). The initial position is a ring lattice in which every vertex is positioned on a ring with edges connected to the *k*/2 vertices to either side. For each vertex, each edge is then replaced with probability *p* with an edge to some other vertex chosen at random. If p=1, then every vertex has *k* edges to *k* randomly chosen vertices. If *k* is positive but close to 0, then there will be neighbourhoods of vertices in which, for any vertex in the neighbourhood, any pair of vertices to which it is connected by an edge will themselves be connected by an edge. But the occasional rewiring of edges will mean that there are short cuts to other neighbourhoods of vertices. Effectively, the lower the value of *p*, the more structure there is to this environment.

As indicated, agents’ actions are represented as additions by the agents to the values at each vertex. Each vertex has a critical value so that when, at any time step, the values added at any vertex by all agents collectively exceeds that critical value, the excess over the critical value is redistributed at random to neighbouring vertices and the value at the original vertex reverts to zero.

The agents are assumed to be able to observe the vertex values but have no knowledge of the edges. This knowledge limitation is encoded by generating a randomly ordered list of vertices at the beginning of each simulation and then generating at each time step a list of the values of the vertices in the same order as in the list of vertices themselves. Each agent can then instantiate the clause (positionValue <index> <value>) where <index> is a position on the list of vertex values and <value>is the value at that position.

The consequence of these assumptions is that, by modelling the agents as acting synchronously but in parallel (so that no agent can take into account the other agents’ simultaneous actions), the changes in the values at any vertex of concern to more than one agent will be what neither of them expected. Moreover, because the agents know the vertex values but have no knowledge of the network structure, they will not be able to distinguish between the effects of several agents acting on a single vertex and the consequence of a toppling from other vertices as a result of actions by agents on any number of other vertices. The inability to distinguish will be particularly acute when the whole system is close to a critical state.

In keeping with the multi agent systems planning literature, each agent is assumed to know a set of recipes. The recipes state that adding any value *x* to an existing vertex value *X* will result in a new value $x + X$ if that sum is less than the critical value and zero otherwise. These recipes are encoded as a set of mental model templates. The general form of these templates is:

```
(modelTemplate <identifier>
        [[(positionValue <index1> <init_value1>) …
```

```
                    (positionValue <index-n> <init_value-n>)]
          [(addedAtPosition <index1> <action1>) …
                    [(addedAtPosition <index-n> <action-n>)]
          [(positionValue <index1> <end_value1>) …
                    (positionValue < end -n> < end _value-n>)]]
```

The first set of clauses is a set of current values at specified positions of the list of vertex values, the second is a set of actions on the values at specified positions and the third is the clauses giving the values at specified positions after the actions have been taken in the specified initial conditions. A plan in the usual way is a sequence of these model templates or recipes. The initial set of recipes given to the agents are of the form

```
(modelTemplate <identifier>
          [[(positionValue ?index 2)]

          [(addedAtPosition ?index  3)]
          [(positionValue ?index 5)]]])
```

It is up to the agents to determine the reliability of these templates and to specialise the templates to conditions where they are reliable. This might take the form of uniquifying the indices or uniquified values at other indices. The point here is that agents can be designed to specialise and combine the initial set of templates to produce a more elaborate set of recipes that are then combined as appropriate into plans

In addition to the initial knowledge of these basic recipe/templates, the agents are allocated goals defined as values at specified positions in the list of vertex values. They have the ability to formulate plans by stringing together the elementary recipes backward from the goal value of a vertex to its current value. However, in the model as implemented so far the agents are not able to anticipate and do not value parsimony. That is, the agents are as willing to string together a set of recipes taking them from the current value to their goal value in unit increments as in the minimal one or (if the goal value is less than the current value) two steps.

The agents are able to observe the activities of other agents. This is in keeping with the real system properties to be captured by the canonical model. When a plan step fails to yield the expected result, the planning agent looks to see whether any other agents have acted on the same vertex. If any have, the agents involved require to engage in a negotiation if any or all are to achieve their goals.

Agent cognition is represented by a problem space architecture as developed for Soar [13] and ACT-R [2]. The architecture assumed to characterise all agents is depicted in Figure 1.

Whenever the agent found that the current value of a vertex differed from its goal value, the agent would enter the plan execution problem space in order to resolve that difference. The first step was to identify or at need to create an appropriate set of templates or recipes. In so doing, two further problem spaces would have to be entered. The first, called endorsing, entails the attachment of mnemonic tokens to models – instantiated templates – that were utilised during the previous time step to determine an action. If the action had yielded the result predicted by the mental model, then that model was endorsed as having succeeded at that time step. If the predicted result had not been realised, the model was endorsed as having failed at that time step. Such endorsements were also attached to the template with an annotation of the particular model that succeeded or failed and when it was invoked.

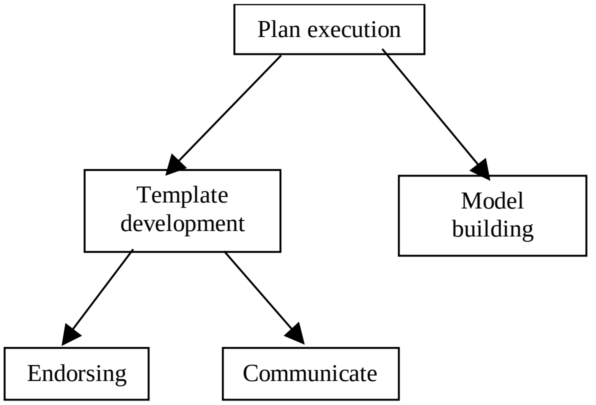

Figure 1. Agents' problem space architecture

If any model failed in the predictions of the outcomes from an action, the agent would look to see if any other agents had acted on the same vertex during the previous time step. If any agent had, a conversation was begun in which the agents could negotiate a reconciliation of any differences between them either with regard to their individual goals or their respective plans of action. Conversations among agents take place over a course of communication cycles within each main time step. Though synchronous, the agents act in parallel. In this case, the parallel synchrony implies that messages passed from one agent to another cannot be read until the following communication cycle. Consequently, it is not uncommon for messages to "pass in the post" so that two agents recognise the effects of each other's actions on a vertex value and simultaneously send messages to on another initiating a negotiation.

These negotiations – whether successful or not – then influence the choice of existing recipes or result in new recipes or model templates used to construct new plans. These new plans are built on instantiations of the best endorsed templates. Among the endorsements are tokens representing the fact of any agreement with other agents regarding an action or set of actions. This feature clearly gives the endorsements a particularly important role in determining the behaviour of the agents.

The endorsements mechanism used in these models is derived from Cohen's [5] conflict resolution scheme. Although the model reported here was implemented in a strictly declarative language (SDML [21]) so that there is no conflict resolution with regard to rule firing, there is still conflict among alternative courses of action by agents and these conflicts are resolved by using endorsements. In effect, each endorsement is placed in a category of endorsements of a given value and these categories are either of positive or of negative endorsements (*e.g.*, modelSucceeded or modelFailed, respectively). The categories are ranked according to importance and the weight differences given to these ranks are allocated randomly to agents. Both the order of the importance of different endorsements and the weighting given to endorsements of different ranks are allocated randomly to agents. So one agent might consider a history of model reliability to be more important in selecting plan components while another might consider it to more important that it had agreed with another agent to engage in the action implied by a particular model. Where an endorsement has a rank one higher than another for two agents – say that model reliability has rank 3 while the existence of an agreement about an action has rank 2, model reliability might be three times as important to one agent but only 1.1 times as important to the other.

One result of this difference is that some agents will be more reliable in carrying out agreed actions and so, will come to be seen by other agents as more trustworthy.[2]

The overall structure of the model and basic agent and mechanism design place no meaningful limits on the particular strategies employed by agents. In some cases there will be directly conflicting goals. In other cases agents either share goals but have devised different plans for achieving them or their goals do not relate to values at the same vertices but the actions required to achieve their separate goals interfere with the plans of other agents. In the latter case, this will be a result of the vertices of interest to each agent themselves being in the same cluster or there being a wider "avalanche" as a result of the activities of all agents bringing the environment network into a highly critical state.

An important feature of this system is that it is not in general possible for all agents to impose their own goal values on all vertices of concern to them. This is obviously the case if they have conflicting goal values at the same vertices. Even where all goals relate to different purposes or they have the same goal values at some vertices, the interactions among neighbouring vertex values prevents the agents from reaching their own or their common goals. Consequently, for the system to reach anything like a Nash equilibrium, the agents will have to reconcile differences regarding both goals and plans.

5. NORMS, INTUITION AND A REFERENCE STRATEGY

The modelling framework described in the previous section supports descriptions of beliefs, desires, intentions, norms, interests and trust. Beliefs are represented by the model templates or recipes, intentions are represented as plans, desires are the goal values of specific vertices, interests relate to the values of vertices neighbouring an agent's goal vertices since changes in neighbouring values will affect the agent's ability to achieve its own goal values. Norms, which will be the main subject of this section, are encompassed by the set of acceptable negotiation strategies and consequent actions by agents.

Instead of representing behaviour in relation to a reference formalism – whether BDI or deontic logic or some mathematical formalism such as game theory – the chosen reference concept is a particular social norm. This norm defines a reference strategy which is a convenient starting point from which other strategies can be represented.

The particular reference strategy reported here is one that has not, in a score of simulation experiments, failed to direct the system into a state that no agent seeks to change. The number of time steps required to reach that state varies in an entirely unsurprising way with the characteristics of the environment network. In particular, the number of time steps to the steady state increases as the rewiring parameter is set closer to 1, the number of edges per vertex is increased and the number of goal vertices per agent is higher. In all cases, the longer path to the steady state is a result of increased interaction among vertices. The scope for independent action by any agent is reduced.

The reference strategy for all agents is the following:

- Whenever any agent know of other agents operating on the same vertices, it informs all of those agents of its interest in that vertex including the agent's goal value.
- Every agent sharing an interest in the value of a vertex realises a uniform random number over the unit interval and the agent with the largest such realisation becomes the only agent to act on that vertex.
- Every agent sharing an interest in the value of a vertex realises a uniform random number over the unit interval and all such agents adopt the goal value of the agent with the largest realisation.

In this way the agents build coalitions centred on vertices of common interest. Since the agents have several vertices of interest, there will be coalitions based on every vertex in which more than one agent has an interest. These coalitions will obviously be overlapping whenever there are three or more agents with enough vertex value goals to ensure that at least some refer to the same vertices as the goals of other agents.

Even this simple strategy offers a range of issues to be resolved. When a coalition is expanded because some agents perceive a common vertex interest with an existing coalition, does the value agreed by the existing coalition predominate or is there a chance that the coalition will adopt the goal value of an entrant to the coalition? In the reference strategy reported here, a new of random number will be realised by each agent and the winning agent's *original* goal value will become the goal value of the whole, expanded coalition. Other options are possible including, for example, the selection of the original goal held by a plurality of the members of the coalition or, if there is more than one such goal value, the random selection of one of them.

There are a number of such design choices to be made and there seems no reason to argue that any one of them is particularly appropriate since none is in any sense intended to be realistic. The purpose of this completely cooperative strategy is to define an initial norm and then to experiment with alternatives. The alternatives are to be taken from actual negotiating environments. As pointed out, the environment network defined here was designed to reflect the reality of a real negotiating and planning environment relating to common pool resource management. The actual set of negotiating strategies and realised actions by stakeholders that are deemed to be acceptable to all of them constitute the relevant social norms. The goals of the different stakeholders determine their interests analogously to the representation of interests in the canonical model reported here. In the canonical model, each agent has an interest in determining not only the actions on the particular vertices in which it has an interest but also on neighbouring vertices and, in a highly critical state of the environment on the neighbours of neighbours and, from time to time, higher degrees of separation.

6. Simulation results

The simulation results reported here are suggestive rather than exhaustive. They are intended to give a flavour of the value of the canonical model approach.

[2] For a more extended description of this endorsements mechanism, see [18]

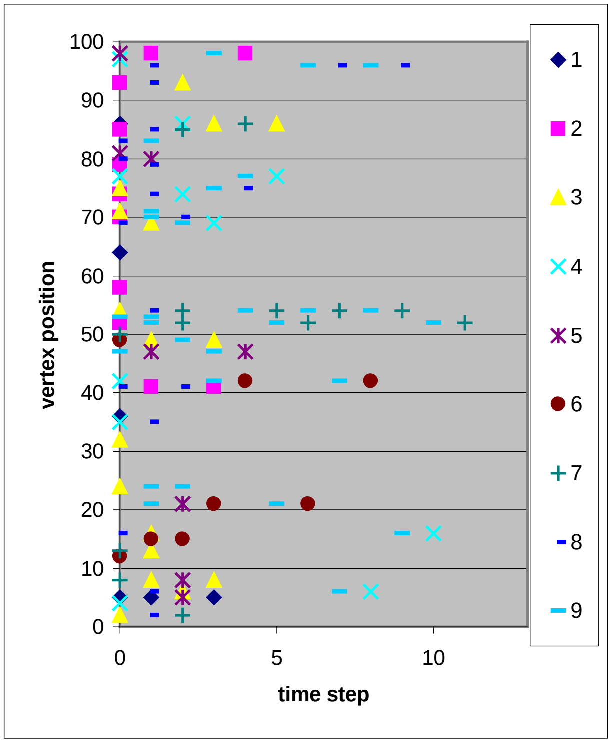

Figure 2. Differences between goal and actual values

One notable result is that all differences in the values of the most highly interconnected vertices were resolved at an early stage. This is seen from Figures 2 and 3 which are taken from the results of a simulation run with two agents, 100 verticesan edge factor (*k*) of 10 and a rewiring probability of 1. The goal distances in Figure 2 are those of an agent with 42 goals. The other agent had 32 goals. There were 12 vertices of common interest to the two agents with 10 goals values were conflicting.

In Figure 3, only those vertices are represented that are directly connected to at least 6 (of a maximum of 10) other vertices of interest to at least one of the two agents. None were connected in this run to more than 7 other such vertices. The reason for this early resolution of conflict among the most apparently intractable goals was that the failure of agents' plans focused their attention on vertices that were in fact most persistently deviating from their goal values because of the interactions among them. The less connected vertices of interest were never out of their goal states for more than one or two consecutive time steps. However, the vertices out of their goal states over the longest unbroken sequence of time steps (the vertices at positions 24, 26 and 51) were among the most closely connected with other vertices. Of these recalcitrant vertices, the vertices at positions 24 and 26 were direct neighbours while the shortest path from the vertex at position 51 to either of the other two was of length 2 (to the vertex at position 24). Evidently, direct connections among goals can cause more difficulty than conflicting values of the same goals under the social norms assumed here. This is not surprising since the impact of the goal conflict is that all parties seek and reach a resolution of the conflict.

An example of how plans evolve and are buffeted by fortune is the experience of the one agent interested in the vertex at position 51. At the time step 0, the value at the vertex was 5 and the goal was 6. Consequently the agent formulated the one-step plan [[(positionValue 51 5)] [(addAtPosition 51 1)] [(positionValue 51 6)]. However, at time step 1, the value at the vertex was 7 as a result of a some criticality at some other vertex and a direct or indirect redistribution of the vertex of interest. Now the agent had to reduce the value at the vertex which could only be achieved by passing through 0 to the lower values. The next plan had three steps

[[[(positionValue 51 7)]
    [(addAtPosition 51 3)] [(positionValue 51 0)]]
[[(positionValue 51 0)]
    [(addAtPosition 51 2)][(positionValue 51 2)]]
[[(positionValue 51 2)]
    [(addAtPosition 51 4)] [(positionValue 51 6)]]]

The first step of this plan was successful but, in re-evaluating the various plan steps open to it, the agent found a shorter, successful template that took it directly to the goal value at time step 2. There were no further disturbances to the value at vertex 51.

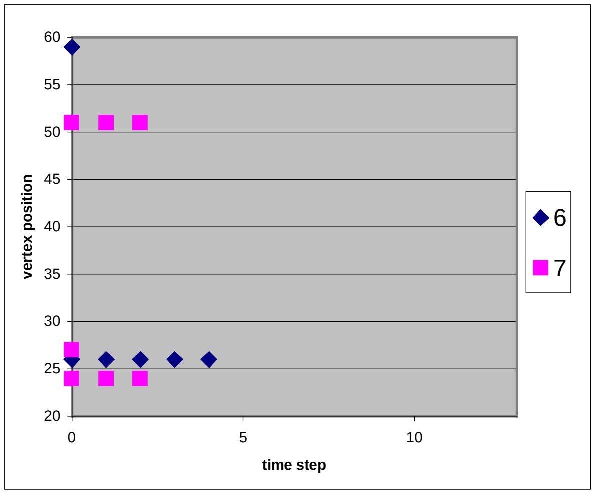

Figure 3. Most connected vertices: goals not achieved

7. Implications for future research

The intuition leading to the model reported here was informed by the situation in the Limberg basin of the River Meuse. The problem is canonical in the sense that it captures in an abstract form a common problem of goal and action of a sort that is found in practice in traditional markets and that designers of agents and mechanisms should keep in mind for applications to competitive environments or environments where agents' actions are likely for technological reasons to affect the outcomes of other agents' actions.

The extreme cooperation imposed as a social norm is wholly unrealistic and certainly not to be found in the Limberg basin. However, the basic technology for capturing actual social norms in the sense of acceptable actions, modes of interaction and commitments among agents to one another is established in this model. By implementing those norms in this model, their representation will be sufficiently abstract and general as to provide pointers for the design of agents and mechanisms and, perhaps more importantly, the identification of social norms that, if imposed on whole systems, will determine the emergent properties of those systems. This approach is, of course, contrary to the more natural bottom-up approach encouraged by the reliance on formalisms for agent architectures. Whether it is a better or more feasible approach is a research question that has not yet been addressed. That there is such a research question suggests that finding means of informing intuition by observation is likely to extend the multi agent systems research agenda. The model reported here is intended to demonstrate that agent based social simulation is an effective means of capturing observation in a form that will support the informing of intuition in the design of multi agent systems.

8. References

[1] Allen, J.F. Towards a general theory of action and time. Artificial Intelligence, *23* (2), (1984), 123-154.

[2] Anderson, J.R. *Rules of the mind*. Lawrence Erlbaum Associates, Hillsdale NJ, 1993.

[3] Bak, P. *How nature works: The science of self organized criticality*. Oxford University Press, Oxford, 1997.

[4] Bratman, M.E. *Intentions, plans and practical reason*. Harvard University Press, Cambridge MA, 1987.

[5] Cohen, P.R. *Heuristic reasoning: An artificial intelligence approach*. Pitman Advanced Publishing Program, Boston, 1985.

[6] Cohen, P.R. and Levesque, H.J. Intention is choice with commitment. Artificial Intelligence, *42* (3), (1990).

[7] Conte, R. and Castelfranchi, C. Simulating multi agent interdependencies: A two way approach to the micro-macro link. in Troitzsch, K., Mueller, Y., Gilbert, N. and Doran, J.E. eds. *Social science microsimulation*, Springer Verlag, Berlin, 1996.

[8] Dean, J.S., Gumerman, G.J., Epstein Joshua, M., Axtell, R.L., Swedland, A.C., Parker, M.T. and McCarrol, S. Understanding anasazi culture change through agent based modeling. in Kohler, T.A. and Gumerman, G.J. eds. *Dynamics in human and primate societies: Agent-based modeling of social and spatial processes*, Oxford University Press, New York and Oxford, 1999.

[9] Dignum, F., Morley, D., Sonenberg, E. and Cavedon, L., Towards socially sophisticated bdi agents. in *International Conference on Multi Agent Systems (ICMAS-2000),* (Boston MA, 2000), IEEE Computer Society, 111-118.

[10] Grosz, B.J. and Kraus, S. Collaborative plans for complex group action. Artificial Intelligence, *86*, (1996), 269-357.

[11] Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Wooldridge, M. and Sierra, C. Automated negotiation: Prospects, methods and challenges. Group Decision and Negotiation, *10* (2), (2001), 199-215.

[12] Jensen, H. *Self-organized criticality: Emergent complex behavior in physical and biological systems*. Cambridge University Press, Cambridge, 1998.

[13] Laird, J.E., Newell, A. and Rosenbloom, P.S. Soar: An architecture for general intelligence. Artificial Intelligence, *33* (1), (1987), 1-64.

[14] LeBaron, B. Empirical regularities from interacting long- and short-memory investors in an agent-based stock market. Ieee Transactions on Evolutionary Computation, *5* (5), (2001), 442-455.

[15] Lux, T. and Marchesi, M. Scaling and criticality in a stochastic multi-agent model of a financial market. Nature, *397*, (1999), 498-500.

[16] Mandelbrot, B. *Fractales, hasard et finance*. Flammarion, Paris, 1997.

[17] Mandelbrot, B. The variation of certain speculative prices. Journal of Business, *36* (4), (1963), 394-419.

[18] Moss, S., Applications centred multi agent systems design (with special reference to markets and rational agency). in *International Conference on Multi Agent Systems (ICMAS-2000),* (Boston MA, 2000), IEEE Computer Society, 199-206.

[19] Moss, S. Canonical tasks, environments and models for social simulation. Computational and Mathematical Organization Theory, *6* (3), (2000), 249-275.

[20] Moss, S. Policy analysis from first principles. Proceedings of the US National Academy of Sciences, (in press).

[21] Moss, S., Gaylard, H., Wallis, S. and Edmonds, B. Sdml: A multi-agent language for organizational modelling. Computational and Mathematical Organization Theory, *4* (1), (1996), 43-69.

[22] Pollack, M.E. Plans as complex mental attitudes. in Cohen, P.R., Morgan, J. and Pollack, M.E. eds. *Intentions in communication*, MIT Press, Cambridge MA, 1990.

[23] Rao, A.S. and Georgeff, M.P., Modelling rational agents within a bdi-architecture. in *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning,* (1991), Morgan Kaufmann Publishers.

[24] Sullivan, D.G., Grosz, B.J. and Krans, S., Intention reconciliation by collaborative agents. in *Fourth International Conference on MultiAgent Systems,* (Boston MA, 2000), IEEE Computer Society, 293-300.

[25] Watts, D.J. *Small worlds: The dynamics of networks between order and randomness*. Princeton Unversity Press, Princeton NJ, 1999.

[26] Wellman, M.P. and Walsh, W.E., Distributed quiescence detection in multiagent negotiation. in *International Conference on Multi Agent Systems (ICMAS-2000),* (Boston MA, 2000), IEEE Computer Society, 317-324.

[27] Wooldridge, M.J. *Reasoning about rational agents*. MIT Press, Cambridge, Mass., 2000.