Approaching Social Coordination from a Normative Perspective

Draft for the SINTELNET workshop on Social Coordination and Analytical Sociology, Stockholm

Pablo Noriega[1] and Julian Padget[2]

[1] IIIA-CSIC, Barcelona, Spain pablo@iiia.csic.es
[2] Department of Computer Science, University of Bath, Bath, UK
j.a.padget@bath.ac.uk

Abstract. We outline a research programme for a class of artificial socio-cognitive systems to support web-enabled collective activities. We claim that it is possible to identify a class of "interesting" socio-technical systems of that type, which share some common features that may give grounds for a formal description of the class. We contend that one can develop a theory about such systems, and develop alongside that theory the methodological guidelines to support the analysis, design and use of new systems. In this paper we introduce the main components of the programme and illustrate our approach with a brief discussion of two topics that we believe may be of interest to social scientists: the notion of "shared context" and a description of participatory agent-based simulation systems in this perspective. Our main objective in this paper is to describe the types of concerns and tools that characterise related work in computer science, so that social scientists may identify areas where an opportunity for collaboration might exist.[3]

1 Introduction

The subject of this paper is a class of systems that are characterised by an entanglement of human and software actions, mediated by various computer interfaces, amongst which browsers and smartphones apps are the most common. We shall call them *artificial socio-cognitive systems.* Such systems are becoming increasingly significant in society around the world as enablers of collective activity that transcend traditional constraints such as co-location, contemporaneaity and identity. Such systems are not just the likes of Wikipedia, Facebook, Twitter and Linked-in, but also Ushahidi, mPedigree, 2go, Farmerline and Mxit, not only helping human/human coordination, but also human/environment coordination.

Since such systems are obviously working, what is there to investigate? We dare to suggest two answers. First, we believe that many of these systems are successful – for some definition of successful – or at least just work, by a degree of good fortune rather

[3] This paper reflects ideas from conversations with Mark d'Inverno and Harko Verhagen. Both they and Julian Padget received support from the European Network for Social Intelligence, SINTELNET (FET Open Coordinated Action FP7-ICT-2009-C Project No. 286370) for short term visits to the IIIA.

than careful in-depth preparatory research and development. [4] Second, we would like to point out that different kinds of generic applications that are frequent, often valuable and arguably unavoidable, share some core features that the previous examples also exhibit, features that we would like make explicit and understand.

Artificial socio-cognitive systems need not be and are not limited to "live" applications, like those mentioned earlier. For example, simulation and gaming, which are becoming increasingly hard to separate (through the phenomenon of 'serious games'), require human participants as much as non-player characters, whether they are being used for training humans, training software entities to behave plausibly or for entertainment. Indeed, the testing and validation of systems, whose complexity may not be fully appreciated in advance, is a problem that stretches conventional notions of system correctness, particularly in respect of adherence to legal codes, regarding, say privacy. While such systems may not have safety-critical roles, they can come to be relied upon by their users and continuous monitoring of integrity constraints – notions of expected correct system behaviour – and recognition of anomalies become necessary aspects of deployment. Consideration of such issues leads to the realization that describing what is right and what is not acquires a key role in system design, from which it is a small step to observing the similarities to and opportunities for policy making – and software processable policy representation – in the context of software-mediated interaction. Furthermore, once policy-making becomes part of the agenda, so does policy revision, the means to capture common patterns of behaviour – and violations that might perhaps indicate mal-specified policy – and how to accommodate these behavioural 'desire lines'[23]. Rather than such revisions being the responsibility of some over-seeing architect, we look forward to a collaborative approach in the spirit of Ostrom's self-governing systems (e.g. ,[19], Ch.9) where changes are initiated, developed and incorporated through the actions of system participants.

We want to explore these issues taking a principled approach. The purpose of this exploration is to develop a theoretical understanding of what these systems are, what goals they help participants realize and what values they help sustain (or repress). The longer term aim is to develop models, methodologies and tools that help in the design, deployment and management of such systems, particularly with an eye on sustainability and mechanisms for self-regulation.

Our immediate objective, in this paper, is to open a conversation with analytical sociologists: a conversation intended to explore whether and how the ideas that we draw

[4] Indeed, it is more likely that a small development effort will go into an app, it is launched and it sinks, swims or occasionally goes viral. In the latter two cases, there may be further development, which may be followed by continued popularity, or collapse. We have not investigated any studies on the rise and fall of the organizations or individuals behind such mechanisms: there is plenty of material about the dotcom era, but whether that remains applicable when development and (trial) deployment costs are so much lower and speculative marketing via app stores is relatively straightforward, is a question for researchers in other fields. Besides, we do not consider ourselves qualified to examine such issues, rather our interest is in understanding what such an app is used for – which may not be the same as the designer's intention – whether it is possible to uncover (through observation or data) sufficient of the norms that appear to govern participant behaviour and whether the sustainability or otherwise of what follows can be attributed to any aspects of those norms.

from the topic of normative multiagent systems [2] can contribute to theory, analysis and experimentation that is of interest to sociologists, and for joint work in the understanding and tooling of social coordination.

The rest of our paper is set out as follows: in the next section we explain our conception of artificial socio-cognitive systems and look into a central conceptual element in our understanding of these systems, namely the notion of "shared context", examining some issues that we hope may be seen as being of interest to social scientists. We then illustrate how these ideas may take form in the particular case of participatory agent-based simulation systems. We conclude with an outline of a potential research programme.

2 Towards a Principled View of Artificial Socio-Cognitive Systems

To support this research programme, we need to understand the scope of the class of artificial systems that might be studied, and thus to identify characteristics that individually and collectively typify instances of the class. Broadly speaking, we want to study systems that involve participants who come together to perform a collective activity that they cannot accomplish on their own and that such participation is not direct among individuals but is mediated by technological artefacts (some sort of interface in a possibly web-supported artificial system).

In fact, we have several implicit assumptions in mind:

mixed Participants may be human or software entities (we shall call them "agents").

rationality Agents are capable of choosing different courses of action based on their own decision-models (however simple or complex these may be).

socio-cognitive We assume that participants act with some degree of rationality that involves individual's cognitive capabilities (like the capability of planning or reasoning about motivations) as well as social capabilities (awareness of others, norm-compliance attitudes, holding altruistic views).

opacity The system, in principle, has no access to the decision-making models, or internal states of participating agents.

openness Agents may enter and leave the system and a priori, it is not known (by the system and other agents) which agents may be active at a given time, nor whether new agents will join at some point or not.

regulated In order to coordinate actions, the system establishes (and governs) some regulations, norms or conventions that agents are in principle supposed to follow.

autonomy Agents are not necessarily competent or benevolent, hence they may fail to act as expected or demanded from them.

dialogical All interactions are mediated by technological artefacts and may therefore be wrapped as communicative acts or messages.

separation Agents are not part of the system.

We may think of these systems as sociotechnical systems because of the participation of humans and software components [24], but because of the assumption of explicit regulation, they are better understood as sociotechnical in the sense of [16] or better yet [22]. We use the term "artificial" because we want to stress the fact that there is some

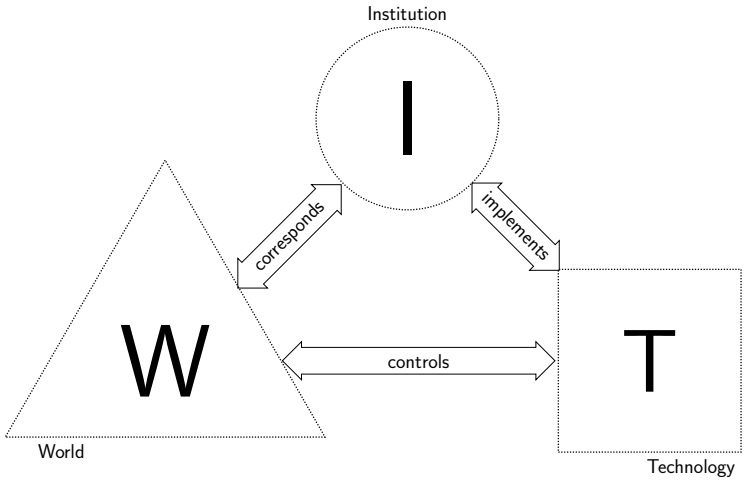

Fig. 1. The three-fold view of artificial socio-cognitive systems: The ideal system (I), the technological artefacts that implement it (T) and the actual world where the system is used (W). After [17].

external design of the system and the term socio-cognitive to stress the fact the we glimpse some notion of social intelligence that is yet to be elicited. In standard multiagent systems terminology, the above assumptions characterise a type of *normative mutliagent systems* [2] that are *open* and *mixed*. This last bias is more evident in the discussion that follows but we believe the artificial socio-cognitive systems deserve a separate treatment.

Although it is still too early to propose a broad taxonomy of artificial socio-cognitive systems, it is nevertheless easy to identify application domains where these mixed socio-technical systems will be paradigmatic. For example, agent-based participatory simulation, hybrid social on-line games, open innovation environments and on-line alternative dispute resolution.

Keeping the aforementioned assumptions and these examples in mind, it becomes natural to conceive of artificial socio-cognitive systems as having three complementary and interrelated, views:

(i) the real-world system, "W", as the users see it and relate to it, and (ii) an ideal institutional system, "I", that stipulates the way the system should behave, while (iii) "T" constitutes the technological artefacts that implement the ideal system and runs the application that enable users to accomplish collective actions in the real world according to the rules set out in I (as depicted in Fig. 1).

These three views are interrelated through three binary relationships. The institutional world is connected with the real world by what is known as a "counts-as" relationship [21, 13] by which actions actions in the real world correspond to actions in the institutional world only when these comply with the institutional conventions, in which case the institutional effects of those institutional actions carry over to have effects in the real world. Secondly, the conventions prescribed in the institutional world

have their counterparts in the technological world in the sense that institutional conventions constitute a specification of the functionality of the system that is implemented in T. In turn, the system, as implemented in T is what enables interactions in W, thus controlling whatever inputs and outputs allow agents in the real world to interact.

It should be noted that each of these three binary relationships needs to satisfy certain "integrity conditions". Thus, the *corresponds* relationship needs: (i) to guarantee that the objects and concepts involved in the descriptions and functioning in I, are properly associated with entities in W, (ii) that the identity of agents in W is properly reflected in their counterparts in I and is preserved as long as the agents are active in the system, and (iii) that the agents that participate in W have the proper entitlements to be subject to the conventions that regulate their interactions and in particular to fulfil in W those commitments that they establish in I. The *implements* relationship needs to be a faithful programming of the institutional conventions (actions and effects are well programmed, norms are properly represented and enforced, etc.). Finally, the *controls* relationship needs to make sure that: (i) the technological artefacts work properly (communication is not scrambled, data bases are not corrupted, etc.) and (ii) inputs and outputs are properly handled in W, according to the implementation of the corresponding processes in I.

Consequently, a fundamental step towards a principled approach to the study of such artificial socio-cognitive systems is to analyse a range of examples of web-based social platforms. This is a task where collaboration among social and computer scientists is essential. The aim would be to understand how people see these platforms, and see themselves within them. The first is a matter of function: what actions does the system allow (by accident or design) participants. The second derives from the first, but tells us more about the analogies through which users see the system and the (multiple, in some cases) personae they adopt when they inhabit that space. The second is particularly valuable because it communicates notions of identity, which are key to the construction of an appropriate security model, in which roles play an essential part.

Alongside is the task of laying down the foundations of a theoretical framework, grounded in the experience of computer science, that can capture the essential features of those platforms that may support socio-cognitive systems and which also begins to offer the opportunity to articulate and perhaps begin to answer some questions that sociologists have about these alternative artificial worlds.

From a technical point of view, one of the most important objectives is the development of metamodels, as means to capture the conceptual approximations to a general characterisation of the class of systems under examination. In a first instance, the metamodel should encapsulate the principal attributes of the studied systems. As such, it functions as a repository of knowledge about such systems. However, it may also be augmented with (formal) transformations to permit the construction of implementations. The objective here would not be to construct the next Facebook, but rather to create a framework (theoretical and computational) to enable the rapid prototyping of systems with different characteristics, as specified by their requirements, processed via the metamodel and turned into code, with which either human subjects (as participants in studies) or software agents might interact. as part of a programme of research into the guidelines on how to study such systems. A consequent step, would be the develop-

ment of accompanying tools, such as perhaps a specification language to describe the properties of existing systems and the creation of new ones, as alluded to above.

We illustrate this with a key concept, "shared context".

3 Shared Context

Everyone appears to recognize examples of context, but a general-purpose definition is elusive, so that commonly, the difficulty is resolved by choosing a domain-specific description. To limit diversion from our main objectives, we will not spend time pursuing generalisations, but aim for a working definition that is fit for our particular type of system.

Interaction between humans leads to the creation of some mutual understanding of the knowledge and intentions of others in the minds of participants. Although the understanding of each individual is inaccessible to other individuals, there is nevertheless a distributed, partial understanding among individuals that allows interactions to progress. This partial understanding constitutes a form of shared context in which to situate and inform, however inadequately, subsequent interactions. Social scientists attempt to record and probe this knowledge base through time diaries and follow-up surveys (etc.), whose limitations are well-known.

In contrast, when dealing with *artificial socio-cognitive* environments, like the ones we are interested in, one may recognise a shared context on more objective grounds. Note that in distributed computation, interactions between software components may lead to changes in their internal state. Likewise, this state is an inaccessible, distributed and partial view of the state of the overall computation. However, if those interactions are somehow within the scope of some larger system, that system may be capable of recording and searching among the interactions and from these interactions obtain and evaluate the global state (and even some individudal states). Thus, such global state of the computation is the core of the shared context in which subsequent actions are situated. Similarly, from an institutional perspective, only those interactions that comply with the institutional conventions may change the institutional state. Therefore, if one has access to these interactions, one may know what the institutional state is. That capability of having access to relevant interactions is the cornerstone of our notion of socio-cognitive systems.

The notion of institutional state is the key to our understanding of shared context. Our intuitions on this matter are easy to grasp with the example of an auction. While an auction is taking place, the shared context is what happens in the auction room. The state of the auction at a given time consists of the people who are present in the room, the picture that is being sold (for example) and where the process of selling it is at that moment. That state is unique, it is objective and applies to all present. So, for example, when the auctioneer says "sold", the picture that was then for sale, at the instant he bangs the gavel is in fact sold to the last bidder and is not for sale thereafter. Similarly, if the auctioneer says "I hear 20" and you raise your bidding paddle, everyone present will acknowledge your action as meaning that you are willing to pay twenty. However, if you scratch your nose or raise your paddle when the auctioneer knocked the gavel, those actions have no institutional meaning and therefore do not change the state of the

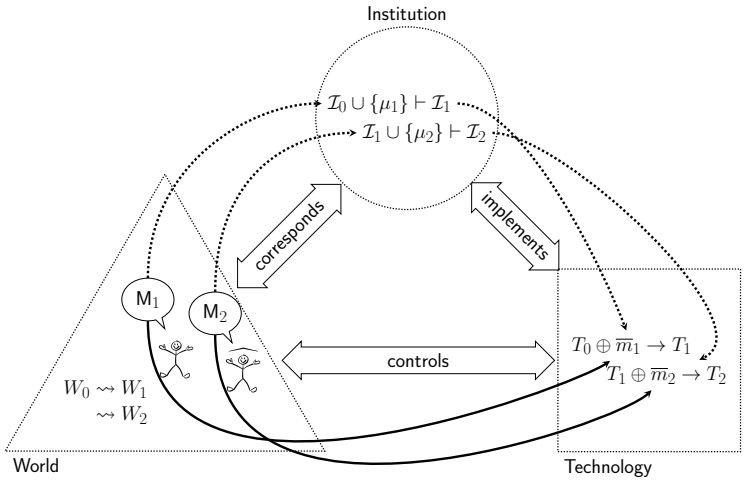

Fig. 2. Shared context in a socio-cognitive system

world. In more abstract terms, there is a correspondence between the shared physical context (the room) and the shared institutional context (the auctioning process), so that a physical action (raising a hand) can change the shared context only when it happens in the appropriate conditions, and if so, it has institutional effects that may in turn produce effects in the physical world (the picture is sold and money will change hands). Notice that in this example we always refer to a *social* state. The internal states of individual participants is evidently changing but we never have to make any reference to them: the only state we need to take into account to describe (and implement) socio-cognitive auction system is the social state.

In order to make the notion of shared context more precise, first we will use the three-fold view of socio-cognitive systems (Fig. 1) to explain how the state changes. Next, we will discuss what is inside the shared context.

3.1 The core of shared context: state, interactions and validity

Fig. 2 illustrates how interactions among individuals take place – in the three-fold view – within a socio-cognitive system. Take two agents a_1 and a_2, who are about to interact, each through its own interface device. Since these individuals are real – they may be human or software agents, no matter – they are present in the real world and, within the system, they share the same *state of the world*, W_0: the objects that exist, the facts that are true and whatever changes take place in that part of the world are the same for both agents *as far as the system is concerned*.

Things start to get interesting when the first agent takes an action M_1 in the world. Provided that M_1 is a feasible action, the state of the world changes from W_0 to W_1 and if a_2 enters a new feasible action M_2 the world, as far as a_1 and a_2 are concerned, changes to W_2. From a computational perspective, inputs M_1 and M_2 correspond to messages m_1 and m_2 that if they can be processed in T, produce new successive *states*

of the system T_1 and T_2. Finally, a similar thing happens in I when an institutional action μ_1, (that corresponds to action M_1 and is implemented as message m_1) takes the system from an *institutional state* I_0 to a new institutional state I_1, if and when μ_1 is an institutionally admissible action.

In other words, we have abstracted the notion of a shared context with two complementary assumptions. On one hand, we propose a three-fold understanding of socio-cognitive systems with a sort of mappings between the three views of the system: (i) mappings among actions, messages and formulas, (ii) mappings among states of the world, system and institution and, (iii) mappings among three *notions of validity* of actions: feasible, processable and admissible. On the other hand, we postulate the notion of state (of the world, system, institution), the use of valid interactions as the sole way of changing that state and the existence of set of conventions that determine when an interaction is valid and if so how it changes the state.

3.2 A normative understanding of the shared context

It should be evident that this abstraction of a shared context in terms of a state that changes with valid interactions, hides a substantial amount of content in the notion of state and in the conditions that regulate interactions. We would like to move towards a more detailed picture of these two aspects without compromising the elegant abstract picture.

If we focus in the W view, it is easy to see that an interaction is feasible in W if and only if it goes into I through the corresponding interface and (assuming I is faithfully programmed) has the intended (programmed) effects on the databases and variables that represent the state in T. Assuming that the implementation and the counts-as correspondences are perfect, the problem of validity is therefore reduced to the notion of *admissibility in* I, hence to an explanation of the expression $\mathcal{I}_n \cup \{\mu_1\} \vdash \mathcal{I}_{n+1}$, whose intuitive meaning is that an action μ changes the institutional state only when it is *compatible* with the institutional conventions and the prevailing facts.

Making that intuitive meaning precise is not straightforward because we may prefer to hold different views about compatibility and each carries its own technical difficulties. One possible approach is to think of compatibility in normative terms. In this case, the classical solution is to say that the system conventions are stated as norms and that an action is compatible with some norms if that action does not contradict those norms. This, however, raises the question of what does it mean for a formula to contradict a set of norms.

The classical solution, again, is to rely in a notion of inference with the standard well-understood (albeit questionable) conception:

Definition 1. *Given a logic* L *with logical language* $\mathcal{L}$*, a formula* φ *and a set* Γ *of formulas in* $\mathcal{L}$*,*
We say that φ can be inferred from *(or is a consequence of)* Γ*:*

$\Gamma \vdash \varphi$ *iff there is a procedure in* L *that proves that if* Γ *holds in* L*, then* φ *also holds.*

We can now express with more precision the intuitive notion of compatibility that we had before making it depend on the notion of inference: let us assume that a state

of the I world at moment m, $\mathcal{I}_m$, consists of a set of formulas in a modal language $\mathcal{L}$, and let us also assume that inference can be predicated with respect to some deontic logic L. We may then say that the set of compatible consequences of $\mathcal{I}_m$ is the set of formulas that can be inferred form it. Actions are also formulas in $\mathcal{L}$.

In order to have a useful formalization of when an attempted action changes the world and how, we may want to introduce some useful distinctions. First we want to separate actions that are institutionally meaningless from actions that are meaningful but undesirable. The first ones are like, in an auction, the one of rising a paddle when the auctioneer is not calling for bids: it has no effect on the sale. The second ones are like a traffic violation that can be detected and should be sanctioned. A meaningless action should not count-as an institutional action, therefore it should not modify the state of the world. Violations on the other hand, are allowed to happen, therefore they have institutional effects and consequently the normative system has to establish the way they are dealt with. To distinguish meaningless actions from those actions that have an effect we use the notion of "regimented" norm. These are norms that we define in such a way that they cannot be violated, ever. One way of, formally, achieving that effect in the I world is as follows:

Definition 2. *Given a set $\mathcal{I}_m$ of institutional facts and norms at time m, a set $\Gamma \subset \mathcal{I}_m$ of regimented institutional conventions, and an attempted action μ, the set of* compatible consequences *of μ in $\mathcal{I}_m$, (denoted $\mathcal{I}_m \cup \{\mu\} \rhd \mathcal{I}_{m'}$), is the set of institutional facts $\mathcal{I}_{m'}$ such that*

$$\mathcal{I}_{m'} = \begin{cases} \mathcal{I}_m & \text{if for some formula } \gamma \in \Gamma,\ (\mathcal{I}_m \cup \{\mu\}) \nvdash \gamma \\ \{\varphi : (\mathcal{I}_m \cup \{\mu\}) \rhd \varphi\} & \text{otherwise} \end{cases}$$

Note that we still left undefined the notion of compatible consequence ($\rhd$). One possibility is to make it identical with our previous notion of consistency ($\vdash$) but it may be the case that we want to have a more parsimonious notion of change of the state of the world. For instance, we may want to keep track not of all possible consistent formulas but only to those that are part of an original set of norms and a few atomic facts. The matter is important for several reasons, not the least that we want to have an implementation of the formal definitions.

Non-regimented norms need a different treatment. On one hand we need to decide what types of norms we want to have and the degree of automation we would like to endow to the system that monitors their application. On the other hand, we need to commit to a model of governance by which we choose the particular ways of identifying violations and how to react to them.

We may prefer, for example, to separate procedural norms from functional norms,. In this way, procedural norms may be expressed —and programmed— as protocols, and some functional norms attached to each step of a protocol as pre- and post-conditions of the admissible actions established by the protocol. In this way we may implement these norms as finite state machines with their well-know qualities and limitations. On the other hand, if we adhere to a logical approach, we need to write norms in such a way that all pertinent aspects of violations and sanctioning are explicitly represented and appropriate automated inference resources are well-implemented.

Governance also requires several design decisions that depend on one hand on the pragmatics of the system we are building, and on the other the generality and robustness of the normative framework we want to implement. One option is to have a fully "governed" supervision by which every attempted action is available to the system and the system itself takes care of all the processing of a violation and enactment of sanctions. Another possibility is to delegate governance in some ad-hoc participants that fulfil the role of norm-enforcers, and yet a third one is to allow for self-governance.

Design concerns do not stop here because we also need to take into account the cognitive make-up of participants. This leads us to the fact that we need to choose languages to express norms and also to decide how much of the system is to be "wired" and where flexibility is needed. In this respect we need to ask ourselves how much rationality we can expect from participants and therefore what parts of the system design may be expressed as explicit norms, how these will become known to participants and how is the evolving state of the world revealed so that agents can act proficiently.

By now it should be clear that when we start thinking of the shared context in normative terms, we need to be aware of the several options that we have at hand at design time, of their operational consequences at run-time, and yet others that will be present when we design for the continued evolution of an artificial socio-cognitive system. A more systematic discussion of these matters is available in [17]. There, in particular, is a list of some of the challenges that the community of normative mulitagent systems has identified with respect to these normative understanding of shared context.

3.3 Non-normative attributes of the shared context

By now it should also be clear that when we start thinking about shared context in practical terms we need to take into considerations several other aspects. We will not discuss them in detail here. One description of these attributes using the case of on-line games is made in [25]. The point we would like to make here is that one may identify a few conceptual attributes that are present in (all?) artificial socio-cognitive systems and that one may concoct an abstract "metamodel" that brings together instances of these attributes to produce a precise specification of the particular socio-cognitive system we are designing or studying.

The attributes we have in mind are the following and we give a hint of how these may be understood in participatroy agent-based simulation systems.

Ontology. These are those "entities", or a collection of "terms" in"I" that are eventually mapped into "W". They will include the elements the are used to define the content of collective contexts and "interactions" (actions). For example, water-use rights, household income, forest, wall; acquire role, improve prestige,...; raise hand, ..

Simulation-generic local contexts are needed to define contexts of collective interaction and their interrelations. For example: action, agent, role, and notably *collective contexts* (ideal locations or activities where several agents interact simultaneously, sharing the *same state*) like "activity" or "scene" or "institution" where different individuals come together an where the same individual may be active in more than one at the same time (e.g., an individual in a primitive society may be involved in resource gathering, household chores, reproductive behaviour, resource gathering;

each of these may have its own norms and certainly a state of that world that is shared by the individuals present in that activity at a given time). Here we may need to decide how several local context may be connected and how individuals may move between them.

Agent types In simulation, these include three main types of "embodied" participants: Humans that play in the simulated world (H), external software agents (X) that model individuals and are not programmed by the simulation designers and simulated agents (A) developed as part of the model. In addition there usually are some server agents, which are not visible to players, that deal with some simulation management functions (for instance performing police-like and time-keeping functions).

Social constructs. Describe the way individuals are related among themselves and also serve as means to refer to individuals and collectives by the role they play rather than by who they actually are. These may include: roles; relations among roles (n-ary relationships between individuals as well as higher-order relationships. i.e, groups, hierarchies of roles, power relationships and so on); organisations (groups plus coordination conventions)

Actions . It is worth distinguishing at least three types: individual actions (get money from the bank); interactions (actions involving two or more agents like make an offer, ask for directions, proclaim an outcome) and actions towards simulation-generic constructs (enter in activities, adopt a new role)

Languages. These are needed to define the behaviour of the system and the way it is regulated. These may be organised as a hierarchy of languages that starts with a *domain* language (to refer to the basic simulation objects: mountains, dispute, sex, role, ,...) that includes terms of higher *action* languages (description of an action); followed by *constraint* languages (preconditions and post-conditions of actions); then *normative* languages (procedural, functional or operational directions; behavioural rules,...) and so on, depending on the complexity of the definition of the simulation and the particular choice of attributes.

Social order constructs. To allow top-down or bottom-up articulation of interactions, the usual device is to use different types of norms: procedural, constitutional, rules of behaviour,...

Social order mechanisms. To allow top-down or bottom-up governance. Among these: regimentation (rendering some actions impossible, strict application of sanctions,...); social devices (trust, reputation, prestige, status, gossip); policing devices (law enforcement),...

Evolution. The rules of some parts of the simulation may evolve in order to adapt to changes in the population, learning of participants, change of welfare function. The definition of the simulation should include the devices through which that change happens: performance indicators, normative transition functions and such.

Inference. A we mentioned above, in case the description of situations is somewhat normative, the designer may want to postulate different ways of inferring intended or observed behaviour. Ways to model reasoning under *uncertainty* and alternatives to classical forms of inference like argumantation of coherence.

Information structures. that are associated with the main entities of the game, agent profiles and the profiles of active simulation constructs. In particular the (shared)

state of the system (the value of each and every variable that may change through the action of some agent or the passing of time) and the shared state of local contexts (generally, subsets of the state of the system).

3.4 Towards a metamodel

Having explained the notion of shared context and enumerated the elements that need to be inside that context in the institutional world I, we postulate that it is feasible to make all these ideas precise and therefore construct formal frameworks that allow for a detailed description, and specification, of concrete, particular, artificial socio-cognitive systems (see [**?**] for a thorough discussion of one such example: electronic institutions).

There may be several frameworks that contain alternative formalizations of the attributes we mentioned. Each of those—we will call them *metamodels*— will allow for specification of systems that may exhibit different features and therefore two metamodel allow for similar but not identical models. Each framework, ideally, will have an implementation platform associated, so that an system specified in that framework (a "platform independent model") is faithfully implemented in the platform (as a "platform specific model") (see Fig. 3). Evidently there is an interplay between what one would like to express in the metamodel and what one is able to program in the platform. For example simulation models programmed in Repast or Netlogo have to buy inherent assumptions that make them better adapted to different modelling techniques.

Actual simulation platforms include not only the platform specific implementation of the socio-cognitive system but also modules and services that capture the platform specific implementation of the attributes described is Sec. 3.3, as illustrated in Fig. 4

A demonstrator developed for the ALIVE project provides a training environment for crisis management officials in the Netherlands[5]. As well as being participatory in nature, its objective was to provide a platform for the evaluation of the regulations governing crisis management and in particular its escalation and the consequent dynamic reorganization, to ensure appropriate continuity at all stages. The simulation comprised: (i) organizational models, expressed in Opera [1], (ii) the actors in the organization(s), modelled as agents on the Agentscape platform [18], (iii) the actions available to the actors, modelled as semantic web services, and (iv) the implementations of those actions, in the form of conventional web services.

3.5 Implementation

There are several approaches to implementation, each of which can play a role in the exploration of the design and realization of social coordination systems. The potential complexity, by which we intentionally mean to refer to unexpected or emergent behaviour, of such systems encourages a cautious start, in which high-level specifications, either equation- or logic-based can help in acquiring an initial understanding of the dynamics of the system. The former is seen as a kind of simulation technique, while the latter is seen as formal specification, according to current computer science conventions: from the point of view of this domain, both are modelling techniques that are more

[5] Developed in conjunction with THALES Nederlands.

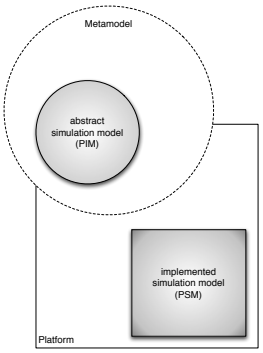

Fig. 3. The simulation model in I is specified as a Platform Independent Model (PIM) that is implemented and runs in T, on a particular platform as a Platform Specific Model (PSM).

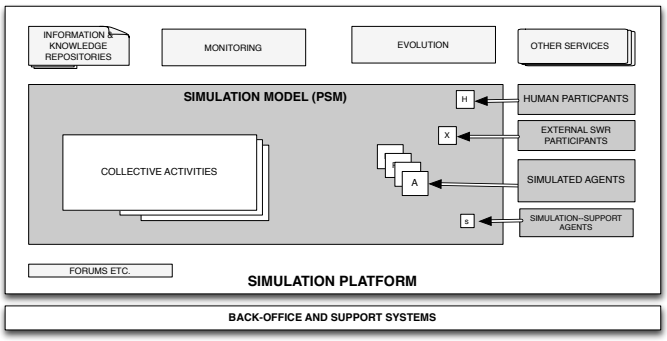

Fig. 4. The Simulation Environment includes the implemented platform-specific simulation model plus those services that are needed to run participatory simulation exercises for model calibration, visualization, policy design or stakeholder negotiations

or less apt, depending on circumstances and offer the opportunity to refine the design through iteration and the testing of model properties. Equally valuable, though lacking the apparent formality of the above approaches, is agent-based simulation. However, it can be argued [10] that agent-based models (ABM) occupy a continuum, of which equational modelling – where all the entities are homogeneous and operated on simultaneously subject to the same reasoning process – is one extreme and agent-based applications – where all the entities *can* be heterogeneous and operate asynchronously, subject a range of reasoning processes, including too the governance of the original logic specification.

As the design evolves towards an implementation, there are two issues of importance: (i) how to ensure that the functional requirements established in the first phase are preserved in the second, and (ii) how to decide upon and achieve non-functional requirements regarding persistence, fault-tolerance, accessibility, etc.. The first is a matter of how to translate the behaviours expressed in the model into the software components provides by the tools and frameworks available. We believe the principles put forward

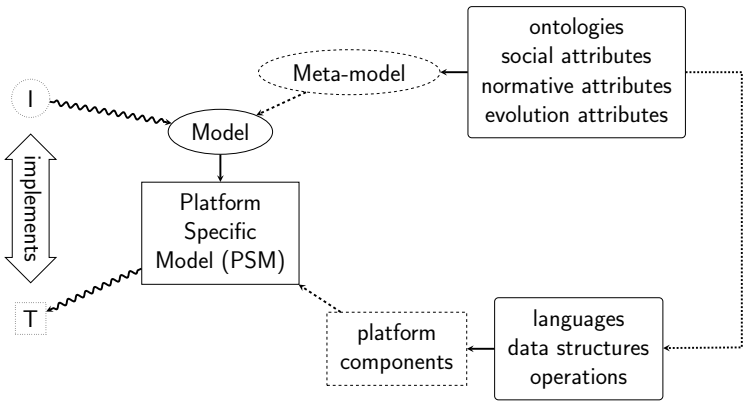

Fig. 5. Realising the technological component from the institutional component

by model-driven design can make a critical contribution here, by formalizing and automating the translation process from model to platform. This is depicted in Figure:5, where the institutional aspect is expressed in terms of the concepts defined in the meta-model, then model transformers emit a platform-specific model (PSM), comprising the code and data structures necessary to interface to particular libraries and frameworks, as exemplified by the Alive project [6] [15]. Clearly, there is a substantial technical overhead to the use of the model-driven approach, but this needs to be traded-off against the adherence of the implementation to the specification (subject to the correctness of the model transformers). The second issue to some extent depends on the facilities provided by those platforms, but also how much information they reveal (reification) about themselves to the application they are hosting and what capabilities they provide to that application to direct platform behaviour (reflection). Ideally, this would permit the governance of the platform hosting the application by the application, expressed as a further collection of institutions.

4 Some concluding remarks

4.1 Remarks on technologies and resources

Ontologies The formalization of ontologies has been progressing steadily for at least two decades. In many specialist cases, there will be an ontology suitable for re-use, or perhaps several that taken together cover the domain. The latter case raises the issue of alignment, for which effective on-the-fly approaches now exists. For general cases, there are WordNet and CYC. In the same time frame, there has been substantial convergence on ontological reasoning with the specification of OWL and the development of several description-logic capable reasoners. All of these elements should provide at least the means to bootstrap this research programme

[6] The ALIVE project (Coordination, Organisation and Model Driven Approaches for Dynamic, Flexible, Robust Software and Services), FP7-215890

Model-driven development This is an emerging technology for software development providing sophisticated tool support, in particular through the Eclipse IDE, and standards from the Object Management Group (OMG). The essential ideas are outlined through the example in section 3.5, which shows how to achieve the objective of increasing the level of abstraction in the design, through the definition of a meta-model, that describes the key domain concept and the relationships between them, and then use translators to generate highly parameterized implementations automatically from system specifications.

Institutional modelling systems These illustrate a research trend that began in the late 1990s, perhaps best typified by the Fishmarket [20], and now exemplified by tools such as its direct descendant, the Electronic Institutions Development Environment (EIDE) [3], Moise+ [12], Instal [7] and Opera/Operetta [1], amongst others. All of these provide ways in which to model collections of norms (so-called institutions), for the purpose of governing agent behaviour, either directly by regimentation, or indirectly through regulation.

Simulation A key tool for exploring the systems design space, but also for exploring models with increasing fidelity, as both individual behaviours of agents are enriched and the governing institutions are refined to reflect more aspects of the actual mixed environment. Initial tools, because of the abstract nature of agent behaviours, include Repast and Netlogo, moving on to Presage [14] and eventually to agent platforms, such as 2APL [8], GOAL [11] and Jason [6], which support rich cognitive models of agent behaviour that can be governed by norms [5, 4].

Open linked data This refers to the publication of (typically) government data about departments, the services they provide and the studies they undertake. Several governments around the world – notably the United Kingdom and New Zealand – have stated their intentions to publish data about government actions in pursuit of transparency and accountability, and in the former case, have established the Open Data Institute [7]. Such data forms an integral part in both the validation and the application of the simulation frameworks foreseen as part of this research programme.

4.2 Some directions for future work

The long-term goal for which this paper is trying to put forward a research agenda, is to bridge the gap between computer science and policy-makers, who we see as the protagonists, along with the wider citizenry, in the development of artificial socio-cognitive systems. We do not expect this to be easy: changing working practices is difficult, especially if the benefits are not immediately obvious and if they represent challenges to existing power. However, we believe there is a need to start over on the processes surrounding policy development to take proper account of the inevitable adoption of artificial socio-cognitive systems as an increasingly common mechanism of government and democracy. A simplistic analysis of the current position with respect to the matter of policy capture and representation suggests three approaches:

1. natural language analysis of existing policy documents to derive formal models – this means no change for policy makers, but can (current) translation technology deliver the hoped for functionality?

[7] `www.odi.gov.uk`, retrieved 20130529

2. diagrammatic specification languages to express policies formally without using a formal language – challenges on both sides: can the diagrammatic language be rich enough or usable enough to satisfy everyone involved?
3. informal modelling framework using familiar tools (browser) and forms to capture roles, actors, actions – attractive in terms of usability but adequacy remains to be established [9].

While this stage presents significant challenges, there are two much greater ones. Just as with software development, verification and validation, are obviously necessary, but unlike software, the techniques have barely begun to be developed and formal validation in particular is in its earliest stages. Beyond this lies deployment, where monitoring raises new issues, in terms of the need to observe prosecution of policy, but because some of the actors are human rather than software, there are the obverses of the same coin that: (i) monitoring may be perceived as invasive and that (ii) monitoring can reveal not only non-compliance with procedures, but also potentially desirable changes, as human actors work out better solutions in practice. The third and final concern we highlight is maintenance – including appropriate revision, in light of changing circumstances or changing requirements. Again, as with testing, while there is a wealth of practical human experience about this process, there is virtually none in its formal application.

In order to kick this agenda off, we propose engagement with policy makers, policy prosecutors and policy subjects, through studies of examples, from which paradigms can be extracted, evaluation of platforms (such as those identified above) for adequacy – followed by a program to develop them to meet the emerging needs of this area – and identification of domains that will bring the challenges necessary to develop the theory and the practice.

References

1. Huib Aldewereld and Virginia Dignum. Operetta: Organization-oriented development environment. In Mehdi Dastani, Amal El Fallah-Seghrouchni, Jomi Hübner, and João Leite, editors, *LADS*, volume 6822 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2010.
2. Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leendert W. N. van der Torre, editors. *Normative Multi-Agent Systems*, volume 4 of *Dagstuhl Follow-Ups*. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013.
3. JosepLlus Arcos, Marc Esteva, Pablo Noriega, JuanAntonio Rodrguez-Aguilar, and Carles Sierra. An integrated development environment for electronic institutions. In Rainer Unland, Monique Calisti, and Matthias Klusch, editors, *Software Agent-Based Applications, Platforms and Development Kits*, Whitestein Series in Software Agent Technologies, pages 121–142. Birkhäuser Basel, 2005.
4. Tina Balke, Marina De Vos, and Julian Padget. Normative run-time reasoning for institutionally-situated BDI agents. In *Coordination, Organizations, Institutions, and Norms in Agent Systems V - COIN 2011 International Workshops, COIN@AAMAS 2011, Taipei, Taiwan, May 2011, COIN@WI-IAT 2011, Lyon, France, August 2011, Revised Selected Papers*, volume 7254 of *Lecture Notes in Computer Science*, page tbd. Springer, 2012.
5. Tina Balke, Marina De Vos, and Julian Padget. I-ABM: Combining institutional frameworks and agent-based modelling for the design of enforcement policies. *Artificial Intelligence and Law*, 2013. Accepted for publication.

6. R.H. Bordini, M. Wooldridge, and J.F. Hübner. *Programming Multi-Agent Systems in AgentSpeak using Jason (Wiley Series in Agent Technology)*. John Wiley & Sons, 2007.
7. Owen Cliffe, Marina De Vos, and Julian Padget. Answer set programming for representing and reasoning about virtual institutions. In Katsumi Inoue, Ken Satoh, and Francesca Toni, editors, *CLIMA VII*, volume 4371 of *Lecture Notes in Computer Science*, pages 60–79. Springer, 2006.
8. Mehdi Dastani. 2apl: a practical agent programming language. *Autonomous Agents and Multi-Agent Systems*, 16(3):214–248, June 2008.
9. Amineh Ghorbani, Pieter Bots, Virginia Dignum, and Gerard Dijkema. Maia: a framework for developing agent-based social simulations. *Journal of Artificial Societies and Social Simulation*, 16(2):9, 2013.
10. Lázló Gulyás. *Understanding Emergent Social Phenomena: Methods, Tools, and Applications for Agent-Based Modeling*. PhD thesis, Loránd Eötvös University of Sciences, 2006.
11. KoenV. Hindriks. Programmingrationalagents in goal. In Amal El Fallah Seghrouchni, Jrgen Dix, Mehdi Dastani, and Rafael H. Bordini, editors, *Multi-Agent Programming:*, pages 119–157. Springer US, 2009.
12. JomiF. Hübner, Olivier Boissier, and Rafael H. Bordini. A normative programming language for multi-agent organisations. *Annals of Mathematics and Artificial Intelligence*, 62(1-2):27–53, 2011.
13. Andrew Jones and Marek Sergot. A formal characterization of institutionalized power. *Logic Journal of the IGPL*, 4(3):427–446, 1996.
14. Brendan Neville and Jeremy Pitt. Presage: A programming environment for the simulation of agent societies. In Koen V. Hindriks, Alexander Pokahr, and Sebastian Sardiña, editors, *ProMAS*, volume 5442 of *Lecture Notes in Computer Science*, pages 88–103. Springer, 2008.
15. Juan Carlos Nieves, Julian Padget, Wamberto Vasconcelos, Athanasios Staikopoulos, Owen Cliffe, Frank Dignum, Javier Vázquez-Salceda, Siobhán Clarke, and Chris Reed. Coordination, organisation and model driven approaches for dynamic, flexible, robust software and services engineering. In Dustdar Schahram and Fei Li, editors, *Service Engineering*, pages 85–115. Springer, 2011. ISBN: 978-3-7091-0414-9.
16. I. Nikolic and A. Ghorbani. A method for developing agent-based models of socio-technical systems. In *Networking, Sensing and Control (ICNSC), 2011 IEEE International Conference on*, pages 44–49. IEEE, 2011.
17. Pablo Noriega, Amit K. Chopra, Nicoletta Fornara, Henrique Lopes Cardoso, and Munindar P. Singh. Regulated MAS: Social Perspective. In Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leendert W. N. van der Torre, editors, *Normative Multi-Agent Systems*, volume 4 of *Dagstuhl Follow-Ups*, pages 93–133. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2013.
18. E. Ogston and F. M. T. Brazier. Agentscope: Multi-agent systems development in focus. In *Proceedings of the Tenth International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'11)*, Taipei, Taiwan, May 2011.
19. Elinor Ostrom. *Understanding Institutional Diversity*. Princeton University Press, 2005.
20. Joan-Antoní Rodríguez, Pablo Noriega, Carles Sierra, and Julian Padget. FM96.5 A Java-based Electronic Auction House. In *Proceedings of 2nd Conference on Practical Applications of Intelligent Agents and MultiAgent Technology (PAAM'97)*, pages 207–224, London, UK, April 1997. ISBN 0-9525554-6-8.
21. John R. Searle. What is an institution? *Journal of Institutional Economics*, 1(01):1–22, 2005.
22. Munindar P. Singh. Norms as a basis for governing sociotechnical systems. *ACM Transactions on Intelligent Systems and Technology (TIST)*, in press:1–21, 2013.
23. James A Throgmorton and Barbara Eckstein. Desire lines: The chicago area transportation study and the paradox of self in post-war america. In *Proceedings of*

the 3Cities Conference. Published on-line at http://www.nottingham.ac.uk/3cities/throgeck.htm, retrieved 20130520, 2000.
24. Eric Trist. The evolution of socio-technical systems. *Occasional paper, Ontario Ministry of Labour*, 2, 1981.
25. Harko Verhagen, Pablo Noriega, and Mark d'Inverno. Towards a design framework for controlled hybrid social games. In Harko Verhagen, Pablo Noriega, Tina Balke, and Marina de Vos, editors, *Social Coordination: Principles, Artifacts and Theories (Social.PATH)*, pages 83–87, Exeter, UK, 03/04/2013 2013. The Society for the Study of Artificial Intelligence and the Simulation of Behaviour.