Towards (Dis)Trust-Based Simulations of Agent Networks

G. Gans,[1] M. Jarke,[1,2] S. Kethers,[1] G. Lakemeyer[1]

L. Ellrich[3], C. Funken[3], M. Meister[3]

[1]RWTH Aachen, Informatik V, Ahornstr. 55, 52056 Aachen, Germany

[2] GMD-FIT, Schloss Birlinghoven, 53754 Sankt Augustin, Germany

[3] Institut für Informatik und Gesellschaft, Universität Freiburg, Germany

ABSTRACT

While the importance of trust in agent networks has been recognized for some time, the role of distrust has largely been ignored. In this paper, we argue that an explicit consideration of distrust and its complex interaction with individual trust and confidence in the network as a whole is necessary for the design and analysis of hybrid networks of human and machine agents. We propose a trust-confidence-distrust (TCD) model of agent network dynamics, and show how the dynamics of such networks can be modeled by extending Yu's i* framework and combining it with the plan language ConGolog. This results in a simulation environment in which network processes and their dependence on trust, confidence in the networks and distrust can be studied.

1 INTRODUCTION

Recently, there has been a growing interest in modeling trust, mainly driven by the advent of the internet and electronic commerce (see, for example, [Falcone et al. 2000, CACM 2000]). Much of the work is concerned with trust in connection with online interactions, where one of its characteristics is that agents normally do not know each other. In this paper, we focus on formalizing trust in social networks, a recently popular form of "coopetition," that promises to combine the benefits of two traditional coordination mechanisms of modern societies [Powell 1990]: the flexibility and speed of competitive market relationships, and the stability and long duration of cooperative, organizational relationships. More specifically, we are interested in networks created among independent organizations to pursue some shared strategic goals, but always with the risk of falling apart.

The relatively small existing literature in this field typically pursues two avenues: viewing trust as a subjective probability, or modeling it in logic. In the collection of papers in [Gambetta 1990], the prevalent view of trust is that of a subjective probability, which, roughly, amounts to the likelihood (assigned by the trusting agent) that another agent will perform a task or bring about a desired situation on which the trusting agent depends. Other work along this line includes [Coleman 1990], who considers trust as a decision under risk. Quantitative measures of trust can also be found in [Marsh 1994, Witkowski et al. 2000], and the game-theoretic approaches to trust [Axelrod 1984, Boon and Holmes 1991, Birk 1999]. Rather than condensing trust to a single value, Castelfranchi and Falcone [1999] propose a more fine-grained model. It takes into account the agents' mental attitudes such as the trusting agent's beliefs about the trustee's opportunity, ability, and willingness to perform a desired task.

Trust being a modality, it seems natural to model trust within modal logic. Such approaches include [Demolombe 1998, Liau 2000] and [Broersen et al. 2000]. The latter consider the notion of "agent i trusts agent j more after doing A than after doing B," which is formalized within the framework of propositional dynamic and deontic logic. Also, Castelfranchi and Falcone [1999] formalize aspects of the mental state underlying trust using a multi-modal logic [Meyer and van der Hoek 1992, Linder 1996].

A very different approach is taken by [Yu and Liu 2000]. They model trust as a so-called soft goal within the i* framework [Yu 1995] which will be discussed in more detail in the next section. Among other things, i* allows to explicitly model goals and tasks of agents and the dependencies between agents as they arise, for example, when a goal of one agent can only be accomplished with the help of another. It is possible to represent how the fulfillment of trust goals can change indicating an increase or loss of trust. In contrast to most other approaches, Yu and Lin's proposal is purely qualitative and the questions of how trust affects an agent's decisions or how to update trust are left open.

Interestingly, none of the approaches in the literature seem to give distrust a special status. By and large distrust is regarded as just the other side of the coin, that is, there is generally a symmetric scale with complete trust on one end and absolute distrust on the other (see, for example, [Marsh 1994]). As we shall argue in section 2, recent sociological research has shown that the relationship between trust and distrust is much more complicated, and that indeed a reasonable amount of both trust and distrust is necessary to keep a social network successful. We condense this research in a dynamic conceptual model called the TCD model (TCD = Trust, Confidence, Distrust) and then show, in section 3, how it can be mapped to a practical modeling and simulation environment for the analysis and support of social networks. The concepts are illustrated with an example taken from an ongoing case study in computer support for entrepreneurship networks. Finally, in section 4, we summarize the status of our prototype implementation and outline plans for further research.

While there is some overlap between the ideas presented here and a companion paper [Gans et al. 2001], the latter differs significantly in that it focuses on a multi-perspective requirements engineering methodology for agent networks.

2 TRUST, CONFIDENCE AND DISTRUST IN SOCIAL NETWORKS

We follow Weyer's definition of a social network as an autonomous form of coordination of interactions whose essence is the trusting cooperation of autonomous, but interdependent agents who cooperate for a limited time, considering their partners' interests, because they can thus fulfil their individual goals better than through non-coordinated activities [Weyer 2000a, Sydow 1992]. We agree with the recently established network sociology (cf. e.g. [Weyer 2000, Sydow and Windeler 2000]) in that we consider the concept of social networks as a phenomenon in its own right, which has to be dealt with by means of new approaches. The distinguishing factor of social networks is their reliance on the mutual trust of the network partners as the main coordination and reproduction mechanism. While this idea has been recognized in recent literature, there has been little research on making it fruitful for the design and ongoing support of networked organizations in a similar way that business process modeling and requirements engineering have been attempting this for traditional organizations and human-machine systems. Moreover, the equally important issue of distrust in organizational networks has been largely ignored or at least over-simplified.

A typical definition in the network literature sees trust as "the willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party" [Mayer et al. 1995, p. 712]. There is no formal agreement on reciprocity, i.e. the relationship between give and take, investment and return where the partners profit mutually from the other partners' actions [Weyer 2000a]. Often, the concept of trust is defined in a rather vague and misleadingly standardized way, disregarding the focal point of network research: what is the relationship between trust in a given situation that the trustor exhibits towards concrete persons or organizations, and the confidence in the network as a whole?

The network as a whole consists of a mesh of dependencies that is not manageable or controllable, nor even completely visible to the trustor, thus requiring *confidence in the system* ("Systemvertrauen" [Luhmann 1988]; cf. also the distinction between personal and institutional trust [Zucker 1986], and between "facework" and "faceless commitments" [Giddens 1990]; cf. also [Scheidt 1995, Loose and Sydow 1997]). Thus, participation in a network results in a double vulnerability, on the one hand to identifiable opportunists, on the other to the generally incomprehensible mesh of dependencies of all network partners. This *distinction between trust and confidence* plays an important role for the regulation and control of social networks. Although networks offer the advantages of organizational cooperation without the disadvantages of organizational bureaucracies and hierarchies, networks need to develop binding rules regulating members' behavior. These rules aim at facilitating trust-based interaction, e.g. by ensuring the confidentiality of information exchanged among partners, by supporting network culture (fair play), reputation, regulation of access [Jones 1997, Staber 2000], or by explicitly defining sanctions for breaches of trust [Loose and Sydow 1997, Ortmann and Schnelle 2000].

Finally, although coordination by means of trust and confidence can enable and facilitate cooperation, it has its costs. In networks, trust and confidence need to be *watchful*, i.e. the partners need to be continually aware of their investments and thus the risks that they incur. This watchfulness leads to a continuous (and potentially costly) monitoring of the individual partners' behavior (trust) and the perceived efficiency of the network as a whole (confidence). On the other hand, watchfulness may also be caused by distrust of or against individuals, where distrust is defined as the expectation of opportunistic behavior from partners, thus breaking the reciprocity of trust-based interaction.

Early research on distrust [Luhmann 1989, Gambetta 1988] treated distrust as danger to be avoided (cf. also [Scheidt 1995]). Only recently, distrust is recognized as an opportunity for making network structures less rigid, and thus more suitable for innovations (cf. [Kern 1998]). Recent investigations on conflict and distrust in organizations [Kramer and Tyler 1996, Lewicki et al. 1998] have established the fact that distrust is an irreducible phenomenon that cannot be offset against any other social mechanisms. Distrust is extremely relevant to social networks, as it not only has a negative influence on networks (as described above), but also can influence the network in a positive way.

Summarizing, we need an approach that addresses trust, confidence, and distrust as separate and simultaneous phenomena in a joint framework. We call this framework the Trust-Confidence-Distrust (TCD) model of success or failure of networks. This model is shown in the three "columns" (thick arrows) of Figure 1, each leading up from actions in the network to changes in the structure – with a feedback loop downwards to the actions via rules created by the structure. In the left columns, confidence-based decisions to incur strategic vulnerabilities create mutual dependencies, in the middle trustful decisions for risky and traceable investments increase reputation, goodwill, and moral integrity, whereas the watchful distrust on the right aggregates latent conflicts by collection, storage and (usually negative) interpretation of events. A balanced mix of all three aspects forms the small corridor for success in networks. The upper part of the figures shows three possible ways of failure caused by imbalances. On the upper left, too many dependencies and goodwill without trust may lead to *successful failure,* which refers to scenarios where networks degenerate to family-like or even mafiose relationships. In contrast, on the upper right, over-aggregated distrust may cause final conflict for the network. And finally, a balanced mix cannot be ensured by simply creating a lot of network rules, because then the transition of the network into an organization will also cause the end of the network.

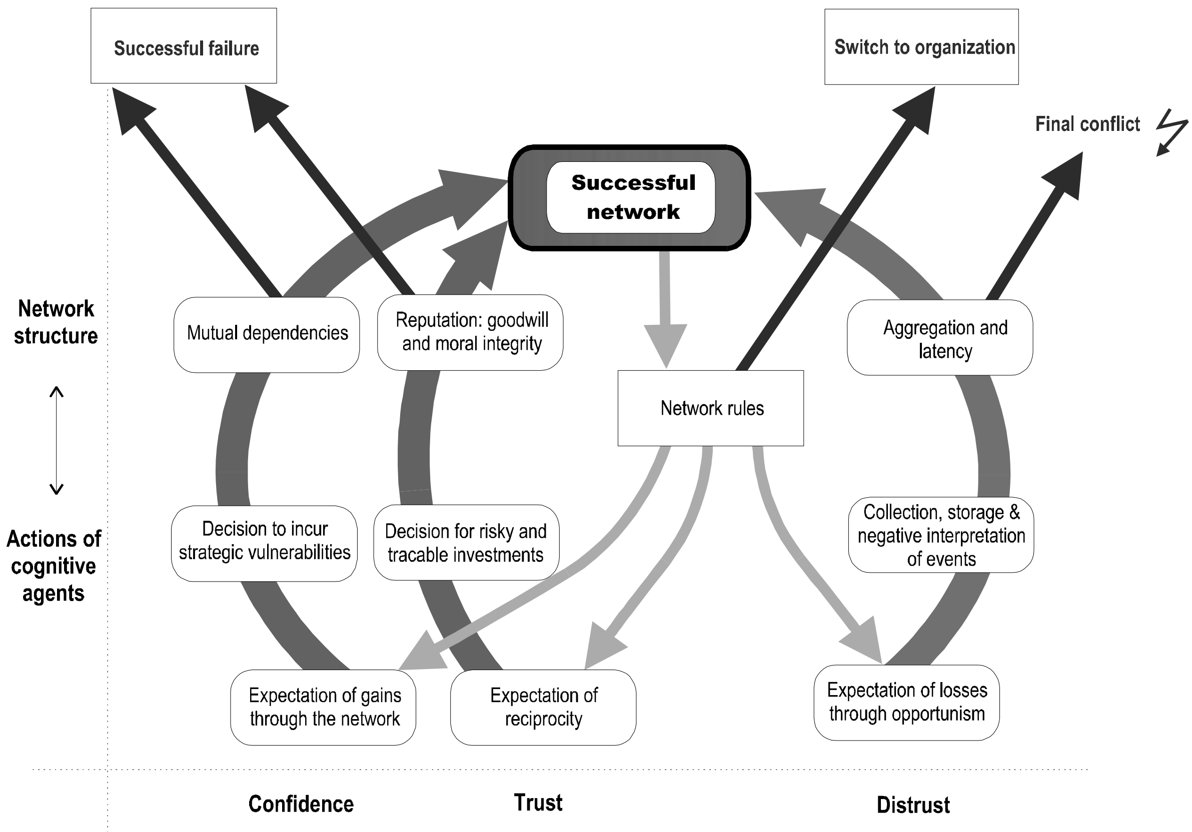

Figure 1: The Trust-Confidence-Distrust (TCD) model of social networks

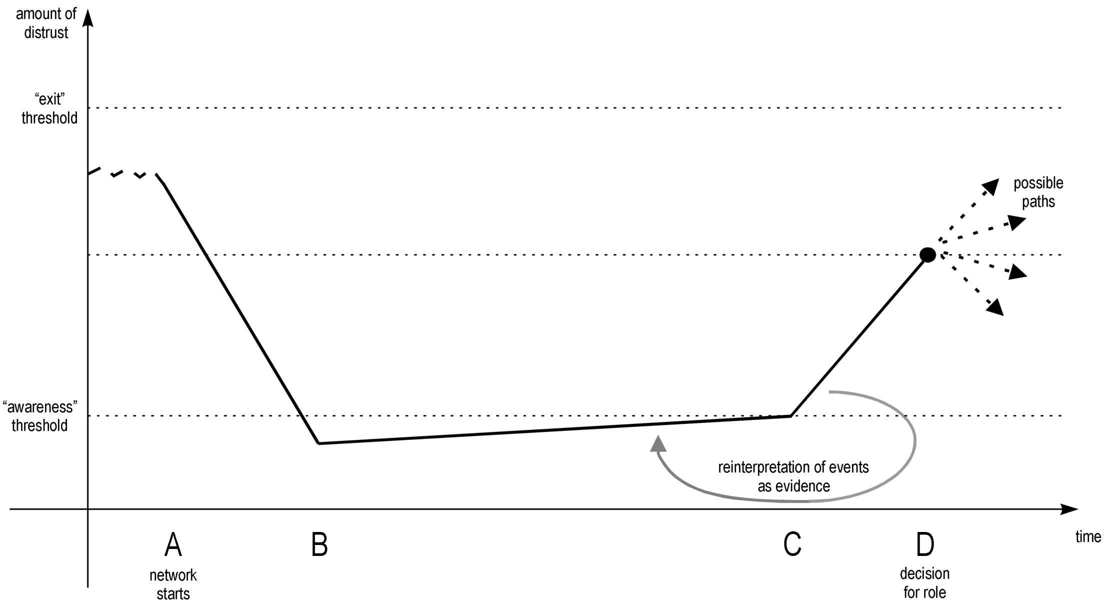

Figure 2: Dynamics of distrust

2.1 The Dynamics of Distrust

Throughout the lifecycle of a network, the development of confidence, trust and distrust seem to follow typical patterns that define thresholds and points of decision where the further development of the network can be decided upon by the network members (cf. Figure 2). In requirements engineering, the value of such patterns lies in the fact that they can be used to identify critical situations and opportunities in a network, as demonstrated for the case of organization analysis in [Nissen et al. 1996].

Prior to the actual formation of a network, empirical research has found evidence for the existence of distrust between the network partners: at the start of a network, the network members seem to be distrustful towards unknown (or little known) cooperating partners [Ortmann and Schnelle 2000].

A general and diffuse consciousness of the dangers related to cooperation requiring major investments leads to a reluctance to act, or to an interaction that relies on very small steps. Once the perception of the network transforms the diffuse dangers of cooperation into a set of calculable risks, the agents think in terms of investments and potential losses and gains. The agents now perceive their actions as influencing other agents' actions, thus losses and gains are perceived as dependent on and influenced by the own actions. Trust and confidence increase the agent's readiness to take risks, thus helping the network to overcome the initial threshold of diffuse distrust.

After the network has been established, a certain euphoria influences the agents' actions (A-B in Figure 2). Once routines are formed, or the network is perceived as not fulfilling the great expectations, the second phase (B-C) follows. Although some network members may develop fears that their expected gains will be reduced, leading to an increase in distrust, a generally positive view of the network will enable the partners to deal with the negative effects. Individual agents start to collect information, but without interpreting it in a negative way. This happens only after a significant and disturbing event has the effect that the agent's distrust crosses an "awareness" threshold. From now on, distrust increases dramatically, absorbing energy and resources of those who experience it. At the next stage (D), the reinterpretation of the network's history forces the distrustful members to make decisions concerning their future behavior. They can then take one of at least three different roles:

- The *distrust maximizer* hides his distrust by openly signaling his willingness for trusting cooperation (potentially even increasing his investments), but also increases his monitoring activities. In other words, the distrust maximizer pretends to be trustful towards other network partners while, at the same time, he is looking for evidence which substantiate his suspicions.
- The *distrust minimizer* also hides his own distrust and instead draws his partners' attention to a perceived distrust problem within the network, hence, in a sense, projecting his own distrust onto the other partners. In

this way, the distrust minimizer achieves a position of superiority, enabling him to demand from the network partners to reevaluate the current situation and perhaps develop and implement new trust-building measures.

- The *quasi-neutral third party* hides his distrust by openly pointing at the network nodes as potential points of conflict, proposing to openly discuss the problems, and offering to become a moderator in the conflict.

Finally, let us stress again why we feel that trust and distrust must be considered separately and not just zero-sum as in most previous approaches. For one, the two notions play very different roles in social networks: the level of trust is a key factor when agents decide whether or not to engage in a risky investment; the level of distrust, on the other hand, controls, among other things, the degree to which an agent monitors others, which can lead to a significant overhead. For another, and perhaps more importantly, both trust and distrust co-exist and vary independently of each other, at least to some degree.[1] For example, increasing the level of trust towards another agent and, at the same time, raising the monitoring activity, i.e. distrust, may be quite reasonable when the latter can be viewed as a sign of healthy watchfulness. Also, as the example of the distrust maximizer shows, an agent may try to hide his rising distrust by not reducing or even increasing his investments, a phenomenon which can be modeled nicely by varying the levels of trust and distrust independently.

In the remainder of this paper, we aim at designing a methodology based on agent models which would on the theoretical side be able to reproduce the dynamics of the TCD model as sketched above, and on the practical side help the designers or coordinators of organization networks assess and improve the status of their network.

3 MODELING THE DYNAMICS OF TRUST-BASED AGENT NETWORKS

We agree with Yu, Coleman, and many others that *explicit modeling of goals and dependencies* is crucial with respect to networks in general and our special focus on trust, confidence, and distrust in particular. We therefore start with Yu's strategic rationale and strategic dependency model. However, as we will see, extensions are necessary to model notions like the (temporal) ordering of tasks. In addition, in order to capture the dynamics of agent networks, we complement the extended strategic rationale models with plans in the plan language ConGolog [De Giacomo et al. 2000], which provides the basis for simulations in the spirit of [Lesperance et al. 1999].

[1] We do not deny that the two notions are correlated. It is just that the correlation is not as strong as it is often assumed.

Among other things, our methodology allows us to model patterns such as the following, which show how trust, confidence, and distrust crucially affect the behavior of agents within a network.

- Existing core trust to specific network agents will enhance the possibility for network action rather than individual action, and thus increase the capabilities of the network.
- Existing network trust (confidence) will enable agents to commit more rapidly to action requested by customers, without prior interaction with possible subcontractors collaborators. This significantly increases the reactiveness of the network as a whole. In contrast, lack of trust will make the network slow and bureaucratic.
- Both of the above will have an impact on the complexity, reliability and speed of collaborative action plans generated.
- Performance monitoring and thus the evolution of trust, distrust, and confidence will be based on relationships between goals, expectations, plans and actual processes. A certain degree of institutionalized network distrust can, for example, be offered by monitoring rules.

3.1 Extending the i* Framework for a Trust-Based Approach

The i* framework proposed in [Yu 1995] offers a conceptual framework for modeling social settings, based on the notions of actor and goal. It assumes that social settings involve social actors who depend on each other for goals to be achieved, tasks to be performed, and resources to be furnished. The i* framework includes the *strategic dependency (SD) model* for describing the network of relationships among actors, as well as the *strategic rationale (SR) model* for describing and supporting the reasoning that each actor performs concerning his relationships with other actors. We will not go over the details of the SD and SR models here, but instead illustrate some of their key features by way of our example.

Graphically, an SD model features actors (drawn as circles) which are connected according to the dependencies they engage in (see Figure 3).[2] As a running example we use a network of specialists who intend to cooperate in performing seminars for outside customers. The specialists considered here are a seminar organizer and a number of speakers available to hold seminars in their respective areas of expertise. Figure 3 shows, for example, that a speaker depends on the seminar organizer for payment and for the opportunity to hold seminars in the future ("further jobs").

[2] While SD models offer various forms of dependencies, Figure 3 only uses so-called goal dependencies.

The strategic rationale (SR) model describes the intentional relationships that are internal to actors, so that they can be reasoned about. Process alternatives can be generated and evaluated. Elements of the model are nodes representing goals, tasks, resources, and softgoals, respectively, and links, representing either means-ends links, or task decomposition links. In Figure 4, we essentially confine ourselves to task decomposition links. The dependencies among agents are included as well. Since agents now have structure, they can be specified at a more fine-grained level of detail.

SR models contain strategically relevant elements only, hence are not suitable for operational use. Perhaps the main deficiency of SR models is that they do not provide the means for specifying an ordering of tasks. Yet in scenarios like social networks where one actor depends on another to achieve certain subgoals or tasks, the order in which things happen is essential. Other aspects not currently covered in SR models include an explicit model of time or the conditional execution of a task. The extensions we propose to SR models draw their inspiration directly from the plan language ConGolog, which has at least two advantages. For one, the concepts ConGolog provides are well-understood and they come equipped with a formal semantics. For another, as we would like to map SR models into ConGolog plans, which are executable and thus usable for simulations, enriching SR model with some of ConGolog's features narrows the gap between the two formalisms and, therefore, eases the task of mapping one into the other.

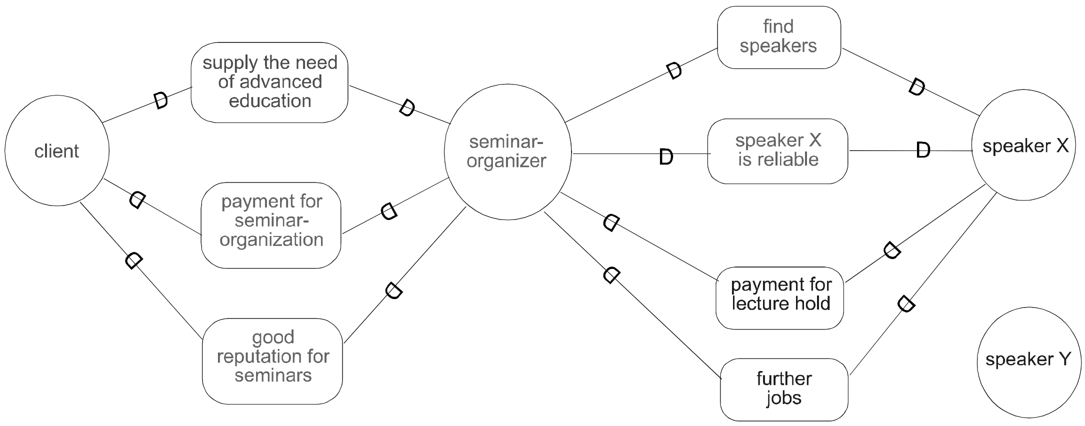

Figure 3: Strategic dependency model for seminar organization example

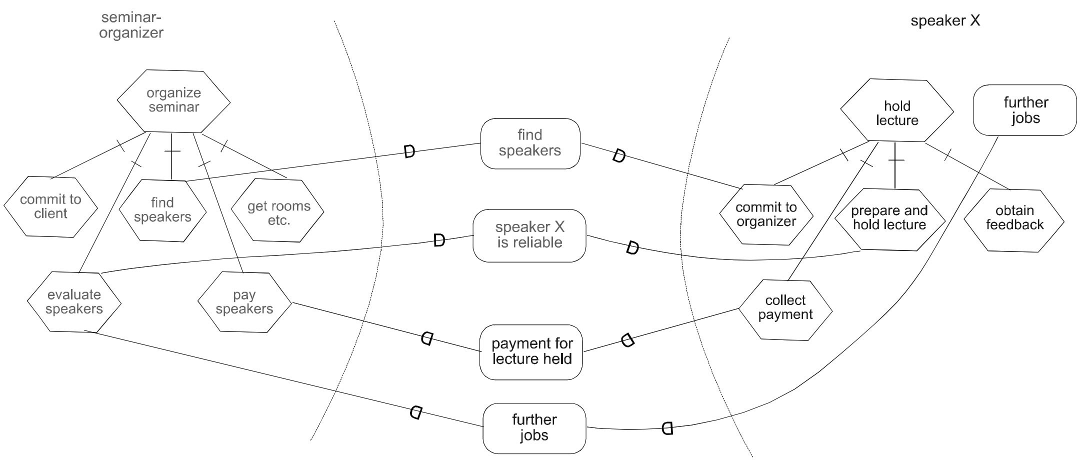

Figure 4: Strategic Rationale model of the "seminar" example

It turns out that only very few extensions to the original SR formalism are needed. The most important new feature is the ability to specify tasks which have *preconditions* as well as postconditions.[3] This allows us to capture the ordering of events in a natural way: A task which has preconditions attached to it can only be performed if all its preconditions are satisfied. Only tasks can satisfy preconditions of other tasks. In our formalism, preconditions are viewed as being both necessary and sufficient for the task to be executed. In other words, as soon as all the prerequisites of a task are satisfied, the task will be performed. Graphically, (see figures below), task preconditions are denoted as triangles labeled with assertions. While the assertions are written here in natural language for readability, they should be thought of as sentences in some formal declarative language like first-order logic. Unlabeled directed edges from a task (hexagon) to a precondition (triangle) are implicitly meant to satisfy or "achieve" the precondition.

We consider two examples where the delegation of subtasks plays a key role and which illustrates how trust, confidence, and distrust crucially affect agent plans.

In Figure 5, the seminar organizer delegates holding the seminar to one or more outside speakers, provided he has enough core trust in them. An advantage of delegation in this case is that the competence is enlarged due to the accumulated competencies of the different potential trustworthy speakers. The downside is that having to find speakers before committing to the client may be time consuming.

Note that, in contrast to [Yu and Liu 2000], trust is not represented structurally as a soft goal but appears instead as part of preconditions. (When mapping SR models into ConGolog for simulations, trust will be denoted by real-valued terms in logic corresponding, roughly, to subjective probabilities which will be updated during simulations.) Note also that, in contrast to Yu's formulation of SR models of different actors, the dependencies between the actors are no longer explicitly represented. In fact, they are now derivable from the SR model. For example, the case where the seminar organizer is the depender, the speaker is the dependee and finding a speaker is the dependum, is now reflected in the fact that the precondition "speaker found," which belongs to the seminar organizer, can only be satisfied by the commitment of the respective speaker. We believe that the initial SD model is usually the result of a preliminary analysis of the dependencies between actors. Once we take a closer look at the agents themselves by designing their corresponding SR models, the dependencies will follow from them. We expect that, in many cases, new dependencies will be discovered in this process which were not considered in the initial SD model

In Figure 6, we illustrate the situation where watchful confidence exists. As described earlier, confidence is different from interpersonal trust in that it expresses trust in the network as a whole being beneficial. The two problems of both narrowed competence and the delayed commitment are now eliminated. The organizer and a pool of speakers are actors in a network. Since the organizer has trust in this network, he can commit to the client immediately. On the other hand, it may also be the case that the seminar organizer distrusts some speakers to a certain extent. As explained earlier, there is no contradiction in trustful actions and hidden distrust at the same time. In our view, distrust is reflected by a certain amount of overhead due to the monitoring of other actors. In Figure 6, a monitor is invoked by the seminar organizer once a speaker he distrusts commits to holding the seminar. The monitor will watch for critical deadlines like the time when the seminar organizer expects the speaker to deliver transparencies to be included in the seminar folder. Should a deadline be missed (or even before that, depending on the level of distrust), the monitor would alert the seminar organizer or send a reminder to the speaker. Since the monitor keeps an eye on the speaker, this also seems to be the natural place where eventually an update of the trust/distrust values occurs, after the overall task of organizing and holding the seminar has ended. Besides its function as a "watchdog," the monitor is also in charge of bookkeeping, that is, it keeps a log of the interaction with the speaker and the eventual outcome of the overall task. Having a log of past interactions with another actor seems to play an important role when assessing the trustworthiness of the actor or, when aggregated over many actors, confidence in the whole network. In our diagram, the details of the monitor are left out for readability.

Figure 6 contains two more features not present in Yu's original proposal: For one, we explicitly refer to a simple (linear) model of time in the *conditional achieve* links. The other new feature concerns the *conditional execution* of tasks, exhibited in the example of creating a seminar folder. The idea is that if the speaker manages to deliver his transparencies on time, then these are included in the folder, otherwise they are not. Rather than introducing conditional tasks as a primitive, we have opted to define them instead in terms of a (non-deterministic) choice between tasks (denoted by "or" in the diagram). Together with the mutually exclusive preconditions attached to those subtasks, we obtain the desired effect.

[3] In Yu's original formalism, postconditions are already captured to some extent in that tasks can achieve or break goals.

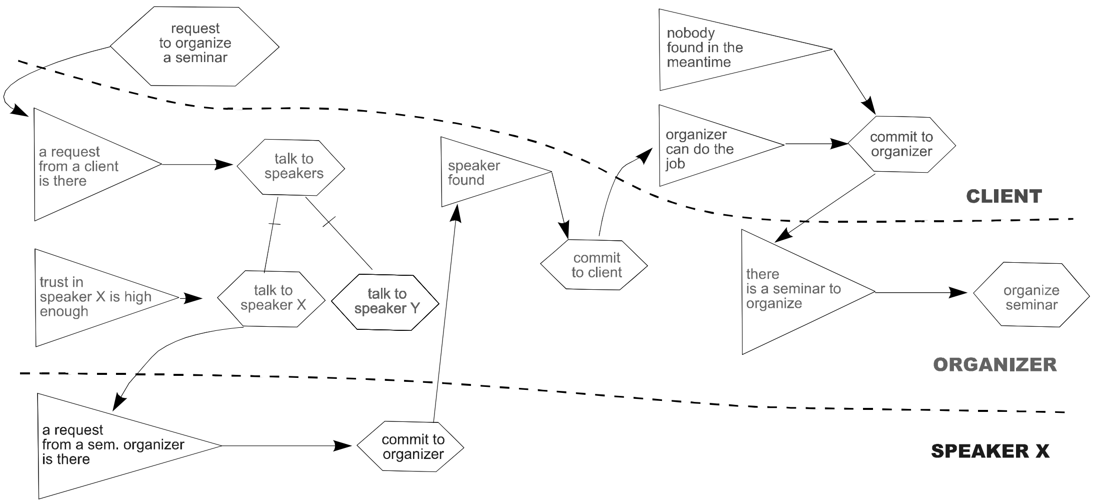

Figure 5: Organizer's plan with trust in individual speakers

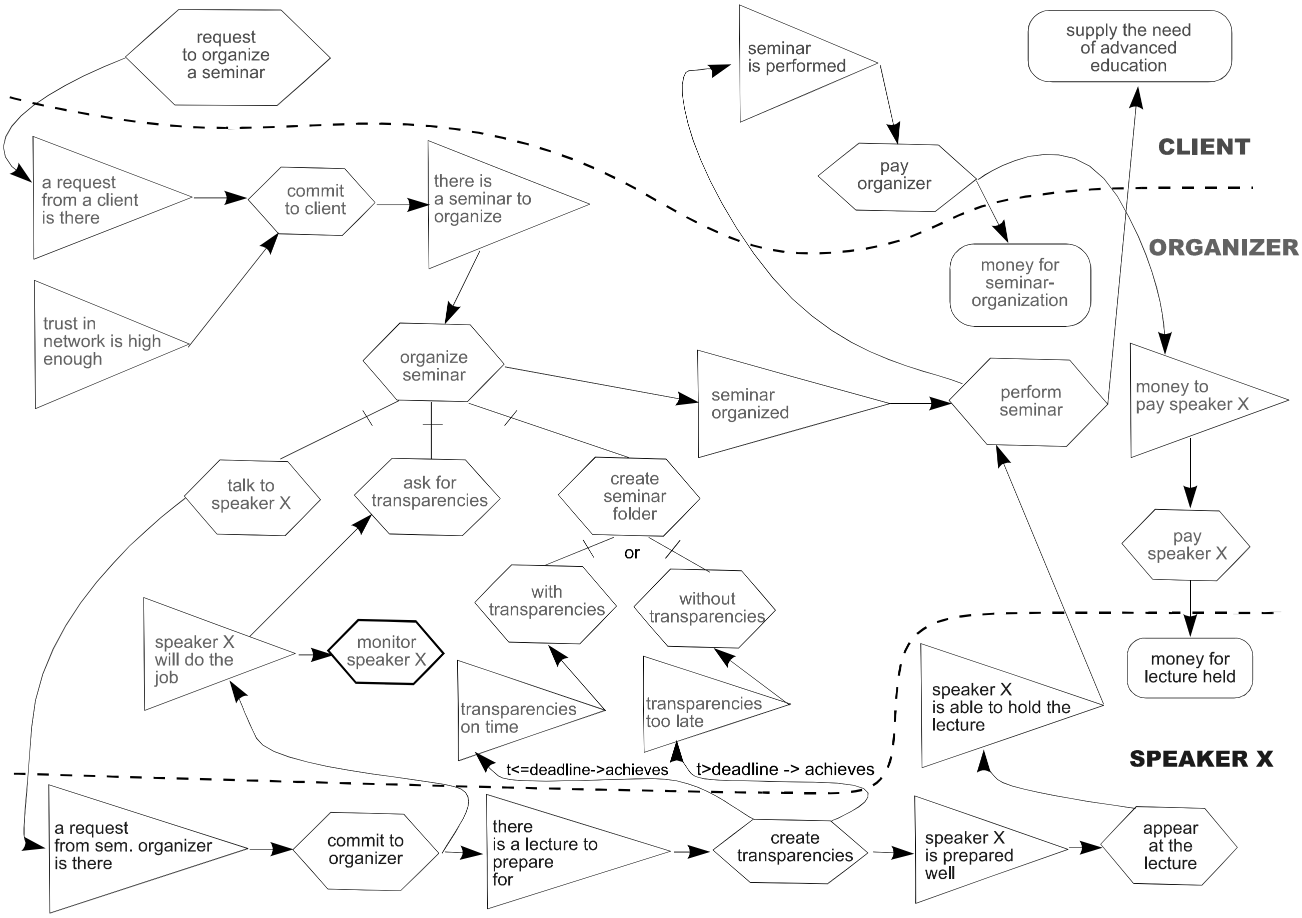

Figure 6: The organizer trusts the network in a watchful way

3.2 Mapping SR Models into ConGolog

ConGolog is a logic-based plan language suitable to model and simulate dynamic domains [Lesperance et al. 1999]. In contrast to most plan languages, ConGolog features concurrent actions and comes equipped with control structures from imperative programming languages such as conditionals, loops, and recursive procedures. Facilitated by the fact that our extensions of SR models are directly inspired by ConGolog, it turns out to be fairly straightforward to map SR models into ConGolog plans. For example, tasks are mapped into ConGolog procedure definitions. Task preconditions correspond to conditionals or interrupts. The latter is useful when a task is to be performed repeatedly (like organizing a seminar), triggered by its preconditions being satisfied (like a new request from a client). Time can be added very easily by introducing a discrete clock which is treated as a separate process run concurrently and with lowest priority with respect to all other processes.

To illustrate the mapping from extended SR diagrams into ConGolog, let us consider the case where the seminar organizer acts based on network confidence with monitoring depending on the level of distrust (Figure 6). Here we confine ourselves to fragments of procedures which are part of the seminar organizer. For space reasons, we will not introduce the language nor its underlying logic-based semantics. The interested reader is referred to [De Giacomo et al. 2000].

proc sem_org

$< \exists c,l,t.\, request(c,l,t)$

/* client ***c*** requests seminar ***l*** to be held at time ***t */***

$\rightarrow$ ***if*** $nettrust(c) \geq NWthreshold$ ***then***

(commit(c,l,t); organize(c,l,t))

else

/* call a proc. to find speaker first and then commit */

endif $>$

end proc

The procedure sem_org can be thought of as the top-level routine which drives the whole process of organizing seminars. It consists of a single interrupt which is triggered whenever there is a request by a client *c* to hold a seminar *l* at time *t* (represented by the fluent *request(c,l,t),* where a fluent is a predicate or function which can be changed by actions). In this case, the organizer immediately commits to the client and then proceeds with the actual organization of the seminar (*organize(c,l,t)*), provided the network trust (confidence) is high enough. Otherwise the organizer would first look for an appropriate speaker, the details of which are left out for simplicity.

proc monitor(x,l,t)

/* Monitor speaker ***x*** regarding seminar ***l*** to be held at time ***t*** */

case distrust(x)

0.0: noop

0.3: ***if*** $time = t - 15$

then if $\neg\exists\, tr.transparenciesReceived(l,tr)$

then *sendReminder1*(x,l)

else *noop* ***endIf***

else *false*? ***endIf***

if $time = t - 10$

then if $\neg\exists\, tr.\ transparenciesReceived(l,tr)$

then *sendReminder2*(x,l)

else *noop* ***endIf***

else *false*? ***endIf***

0.5: ...

end proc

The activity to monitor the speaker depends on the level of distrust the seminar organizer has with respect to speaker *x*. Here we only consider two cases: if there is no distrust at all (0.0) then no actual monitoring occurs; if there is some distrust (0.3) then the arrival of the transparencies is monitored by sending two reminders if necessary, one at time t-15 and another one at time t-10. (The use of *false?,* which is a test which can never succeed, as part of the if-statements guarantees that the if-statements are blocked until the respective time has arrived, where time is a process running in parallel to the agent processes.)

Note that the concurrency mechanisms in ConGolog allow a natural specification of monitors, since these are best thought of as running in parallel to other activities. Of course, concurrency is already essential for the simulation of multiple interacting agents as in our case.

Apart from specifying plans for the respective agents in terms of ConGolog procedures, we also need precondition axioms for each primitive action and so-called successor state axioms, one for each fluent, which state precisely how a fluent changes or does not change as a result of a primitive action. An example successor state axiom is the following:

$$\forall a,s,x.trust(x,sem,do(a,s))=p \equiv$$
$$[a = updateTrust(x,sem) \wedge$$
$$\exists v.currentSem(v) \wedge p = f\_trust(hist(x,sem),v)] \vee$$
$$[a \neq updateTrust(x,sem) \wedge p = trust(x,sem,s)]$$

The fluent trust(x,sem,s) refers to the seminar organizer's trust in speaker x with respect to holding seminars (represented by the constant sem). The axiom is only given in a schematic form since the new trust value depends on the function *f_trust*, which takes as parameters the history of interactions with speaker x with respect to seminars and the current seminar event *v*. We plan to compare different versions of *f_trust* found in the literature, e.g. [Coleman 1990] and [Marsh 1994], by testing them in our simulation environment currently under development.

4 SUMMARY AND CONCLUSIONS

The key sociological idea underlying our work is a three-column success model for social networks, resting on the columns' individual core Trust, network Confidence, and Distrust. This TCD model requires that we do not confine ourselves to structural ingredients like goals and dependencies as in Yu's model, but explicitly address the dynamics of agent networks, which we achieve by deriving ConGolog plans from suitably extended strategic rationale models. The execution of such plans will allow us to simulate and predict the evolution of trust, confidence, and distrust in agent networks.

In [Gans et al. 2001], we incorporate, in addition, a language-action perspective based on [Medina-Mora et al. 1992, Winograd and Flores 1986] into our methodology, which allows us to explicitly model *expectations*, which also play a crucial role in agent networks. We are presently completing a first implementation of this integrated formalism, based on an embedding of ConGolog with the ConceptBase metadata manager [Jarke et al. 1995]. This implementation will then be used with examples from our ongoing case study in cross-Atlantic entrepreneurship networks to validate the TCD model itself and predictions resulting from it. Specific TCD-based lifecycle hypotheses as discussed in Section 2.1 can, after this initial validation and calibration, be used for problem analysis in ongoing requirements management efforts in networks, e.g. concerning the appropriateness of proposed network rules, the situation-dependent optimal mix of trust, confidence, and distrust, evaluation of specific strategic actions, and the like, as suggested by the nodes and links in the TCD model of Figure 1. In the longer run, we plan to integrate these mechanisms in the design an infrastructure for computer-supported cooperative work for distributed internet-based communities [Appelt et al. 2001] which is tuned to the specific needs of organization networks, as seen from the viewpoint of the TCD model.

Finally, regarding the more formal aspects of our work, an interesting analogy exists between trust and distrust on the one hand and belief and disbelief in an Assumption-Based Truth Maintenance System [de Kleer 1986]. In both cases, the respective values need not be zero-sum. Whether there are deeper connections and, if so, how to exploit them remains to be seen.

5 ACKNOWLEDGMENTS

This work is supported in part by the Deutsche Forschungsgemeinschaft in its Focused Research Program on Socionics. We would like to thank the anonymous referees, whose comments have helped us improve the presentation. Numerous discussions in the emerging MIT-NRW entrepreneurship network are gratefully appreciated.

6 REFERENCES

Appelt, W., Jarke, M. and Prinz, W. (2001*): Emergence of a global cooperation network infrastructure*, submitted to a special issue on globalization in CSCW, Comm. ACM.

Axelrod, R (1984): *The Evolution of Cooperation*, New York: Basic Books.

Birk, A. (1999): *Learning to Trust*, in [Castelfranchi et al. 1999]

Boon, S.D. and Holmes, J.G. (1991): *The dynamics of interpersonal trust: resolving uncertainty in the face of risk*, Hinde, R.A. and Groebel, J. (eds.): Cooperation and Prosocial Behaviour, Cambridge University Press, 190-211.

Broersen, J., Dastani, M. and van der Torre, L. (2000): *Leveled Commitment and Trust Negotiation*, in [Falcone et al. 2000].

CACM (2000): Special Issue on "*Trusting Technologies*", Comm. ACM 43, 12, Dec. 2000.

Castelfranchi, C. and Falcone, R. (1999): Social Trust: *A Cognitive Approach, invited talk at International Workshop on Agent-Oriented Information Systems*, CAiSE 1999, Heidelberg. http://www.aois.org/99/castelfranchi-Social-Trust-paper.doc

Castelfranchi, C., Tan, Y.H., Falcone, R. and Firozabadi, B.S. (eds.) (1999): *Proceedings of the Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies*, Seattle, Wa, 1999.

Coleman, J.S. (1990): *Foundations of Social Theory*, Harvard University Press, Cambridge/Mass.

De Giacomo, G., Lespérance, Y. and Levesque, H. (2000): *ConGolog, a concurrent programming language based on the situation calculus*. Artificial Intelligence 121, 1-2, 109-169.

de Kleer, J. (1986): *An assumption-based TMS*, Artificial

Intelligence 28, 127-162.

Demolombe, R. (1998): *To trust information sources: a proposal for a modal logical framework,* in Proceedings of the Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies, Roma, Italy, 1998, 9-19.

Falcone, R., Singh, M. and Tan, Y.H. (eds.) (2000): *Proceedings of the Autonomous Agents Workshop on Deception, Fraud and Trust in Agent Societies*, Barcelona, Spain, June, 2000.

Gambetta, D. (1988): *Mafia: the price of distrust*, in Ders (ed.): Trust. Making and Breaking Cooperative Relations, New York, 158-175.

Gambetta, D. (1990): *Trust. Making and Breaking Cooperate Relations*, Oxford: Basil Blackwell.

Gans, G., Jarke, M., Kethers, S., Lakemeyer, (2001*), Modeling the Impact of Trust and Distrust in Agent Networks*, to appear in: Proceedings of the 3rd International Workshop on Agent-Oriented Information Systems, Interlaken.

Giddens, A. (1990): *The Consequences of Modernity*, Oxford.

Jarke, M., Gallersdörfer, R., Jeusfeld, M.A., Staudt, M. and Eherer, S. (1995): *ConceptBase: A deductive object base for metadata management*, International Journal of Intelligent Information Systems 4, 2, 67-95.

Kern, H. (1998): *Lack of Trust, Surfeit of Trust: Some Causes of the Innovation Crises in German Industry*, in Lane, C. and Bachmann, R. (eds.): Trust Within and Between Organizations, Oxford, 203-213.

Kethers, S. (2000*): Multi-Perspective Modeling of Cooperation Processes*. Doctoral Thesis, RWTH Aachen, Germany.

Kramer, R.M. and Tyler, T.R. (eds.) (1996): *Trust in Organizations: Frontiers of Theory and Research*, Thousand Oaks, Sage

Lespérance, Y., Kelley T.G., Mylopoulos, J. and Yu, E. (1999): *Modeling dynamic domains with ConGolog, in Jarke, M. and Oberweis, A. (eds.): Advances in Information Systems Engineering*. Proceedings CAiSE 1999. LNCS 1626, Springer Verlag, 365-380.

Lewicki, R.J., McAllister, D.J. and Bies, R.J. (1998): *Trust and Distrust: New Relationsships and Realities*. Academy of Management Review 23, 3: 438-458.

Liau, C.-J. (2000), *Logical Systems for Reasoning about Multi-Agent Belief, Information Acquisition and Trust*, Proc. ECAI-2000, 368-372.

Loose, A. and Sydow, J. (1997): *Vertrauen und Ökonomie in Netzwerkbezeihungen – Strukturationstheoretische Betrachtungen*, in: [Sydow et al. 2000], 161-192.

Luhmann, N. (1988): *Familiarity, confidence, trust: problems and alternatives*, in: Gambetta, D. (ed.): *Trust. Making and Breaking Cooperative Relations*, Oxford, 94-107.

Malone, T.W., Yates, J. and Benjamin, R.B. (1987): *Electronic Markets and Electronic Hierarchies*. Communications of the ACM 30 (6), 484-497.

Marsh, S. (1994): *Trust in Distributed Artificial Intelligence*, in: Castelfranchi and Werner (eds.): *Artificial Social Systems*, Berlin/New York, 94-111.

Mayer, R.C., Davis, J.H. and Schoorman, F,.D. (1995): *An integrative model of organizational trust*, Academy of Management Review 20, 3, 709-734.

Medina-Mora, R., Winograd, T., Flores, R. and Flores, C.F. (1992): *The Action Workflow Approach to Workflow Management Technology*, Proceedings 4th ACM Conf. *Computer-Supported Cooperative Work*, Toronto, 281-288.

Meyer, J.J. Ch. and van der Hoek, W. (1992): *A modal logic for nonmonotonic reasoning*, in van der Hoek, W., Meyer, J.J. Ch., Tan, Y. H. and Witteveen, C., (eds.): *Non-Monotonic Reasoning and Partial Semantics*, 37-77. Ellis Horwood, Chichester.

Nissen, H.W., Jeusfeld, M.A., Jarke, M. and Huber, H. (1996): *Managing multiple requirements perspectives with meta models*, IEEE Software, March 1995.

Nuseibeh, B., Finkelstein, A. and Kramer, J. (1996): *Method engineering for multi-perspective software development*, Information and Software Technology 38, 4, 267-274.

Ortmann, G., Schnelle, W. (2000*): Medizinische Qualitätsnetze – Steuerung und Selbststeuerung*, in: Sydow, J./Windeler, A. (eds.): *Management interorganisationaler Beziehungen. Vertrauen, Kontrolle und Informationstechnik*, Opladen, 206-233.

Powell, W. (1990): *Neither market nor hierarchy: Network forms of organization*, in Staw, B. and Cummings, L.L. (eds.): *Research in Organizational Behavior*, Greenwich, 295-336.

Scheidt, B. (1995): *Die Einbindung junger Technologieunternehmen in Unternehmens- und Politiknetzwerke. Eine theoretische, empirische und strukturpolitische Analyse*, Berlin.

Staber, U. (2000): *Steuerung von Unternehmensnetzwerken: Organisationstheoretische Perspektiven und soziale Mechanismen*, in Sydow, J., Windeler, A. (eds.): *Steuerung von Netzwerken*. Opladen: Westdeutscher Verlag, 58-87.

Sydow, J. (1992): *Strategische Netzwerke. Evolution und Organisation*, Wiesbaden: Gabler.

Sydow, J., Windeler, A., (eds.) (2000): *Steuerung von Netzwerken*, Opladen/Wiesbaden: Westdeutscher Verlag.

Tan, Y.H. and Thoen W.A.W. (2000): *Formal aspects of a generic model of trust for electronic commerce*, in [Falcone et al. 2000].

van Linder, B. (1996): *Modal Logics for Rational Agents*, Ph.D.

Thesis, Department of Computing Science, University of Utrecht.

Weyer, J., (ed.) (2000): *Soziale Netzwerke. Konzepte und Methoden der sozialwissen-schaftlichen Netzwerkforschung*, München.

Weyer, J. (2000a): Zum Stand der Netzwerkforschung in den Sozialwissenschaften, in: [Weyer 2000], 1-34.

Williamson, O.E. (1975): *Markets and Hierarchies: Analysis and Antitrust Implications*, New York 1975.

Winograd, T. and Flores, F. (1986): *Understanding Computers and Cognition: A New Foundation for Design*, Norwood, NJ: Ablex.

Witkowski, M., Artikis A. and Pitt J. (2000): *Trust and Cooperation in a Trading Society of Objective-Trust Based Agents*, in [Falcone et al. 2000].

Yu, E., Liu, L. (2000): *Modelling Trust in the i* Strategic Actors Framework*, in [Falcone et al. 2000].

Yu, E. (1995): *Modeling Strategic Relationships for Process Reengineering*. Ph.D. Thesis, University of Toronto, Canada. Also published as Technical Report No. DKBS-TR-94-6.

Zucker, L. (1986): *Production of Trust: Institutional Sources of Economic Structure*, Research in *Organizational Behaviour* 8, 53-111.