Modeling Dependence Networks for Agent Based Simulation of Online and Offline Communities

Francesca Marzo, Stefano Za, and Paolo Spagnoletti

CeRSI – LUISS Guido Carli University, Rome, Italy
{fmarzo,sza,pspagnoletti}@luiss.it

Abstract. Far from simply being a concept useful in investigating social relationships, social networks are rapidly becoming a diffuse phenomenon to deal with in everyday life. The goal of this paper is to provide insights from the design research perspective, both for online and offline communities. Starting from the idea that the phenomenon under investigation emerges from the interaction of autonomous agents in an environment in which other agents interact with each other in order to reach their own goals, we adopt a Multi-Agent Simulation (MAS) approach to study social networks dynamics of online and offline communities. In particular, we built an agent-based simulation of dependence networks, considered crucial for the interaction of cognitive agents and for the exchange of resources between them. As results we have been able both to better define some hypotheses on dependence networks dynamics and to highlight possible future research particularly useful for the design of digital platforms.

Keywords: Social networks, Communities, Dependence Networks, Cognitive Agents, Multi-Agent Simulation.

1 Introduction

With the rapid growth of online social networking and open source models, individuals are increasingly engaged in online activities by interacting with peers and organizations through digital platforms and applications running on personal computers and mobile devices [1–3]. Such social network applications have the potential to facilitate the exchange of resources among members by providing platforms for the exchange of ideas and quality information [4]. When effectively designed and implemented, digital platforms enable information exchange, collaboration and collective action within online and offline communities [5]. Although the behavior and motivation of members participating in online (virtual) communities has been widely investigated, more research is needed on how to design, build, and sustain a vibrant platform [2].

The presence of multi-level factors in the ecological model of community behavior, poses many challenges when investigating the mutual relationship between agency and structure in this context [6]. Previous studies on social networks mainly focus on a behavioral approach that reflects a positivist stance by performing both quantitative and qualitative analysis for validating causal models. Our goal is to

Y. Demazeau et al. (Eds.): PAAMS 2013, LNAI 7879, pp. 192–203, 2013.

complement these studies by adopting a constructivist viewpoint which extends the methodological framework for investigating this phenomenon and provide insights also from a design research perspective [7]. Drawing on complex adaptive systems theory, we recognize that social phenomena emerge from the bottom-up interactions among learning agents in a given environment [8]. This view has recently gained much attention in different fields of management and organizations [9–12] and integrate contributions from cybernetics, cognitive sciences, decision and organization sciences [13].

In this paper we present an agent based model for simulating the dynamics of small groups of interacting agents whose behavior is determined by their learning capability and a set of environmental rules. The model is grounded on the theory of dependence networks [14, 15] and provides a tool for studying emergent properties/phenomena within online and offline communities. Following previous works that introduce formal models of dependence networks, we develop an architecture of cognitive agents and of the environment in which they act and interact. This architecture will be the basis for implementing a platform for agent based simulation that will serve as a tool for supporting decision making in the design of digital platforms and in the governance of online and offline communities.

The paper is structured as follows. We first introduce the theoretical framework on which the model is grounded. Then we describe the architecture of the model and we illustrate an example of simulation. Finally we discuss about implications and research directions.

2 Cognitive Model of Dependence Networks

Starting from Conte and Castelfranchi's theory [16] of dependence network, we assume the idea that agents are part of a network of social relationships and that these relationships are fundamentally based on agent's mental states. Therefore, social networks are built on networks of goals. Among these relationships, one of the most important type is social dependence, consisting in some agents needing other agents' to reach their goals.

We take our move from a previous study on agent-based simulation of dependence network aimed to supply a tool for improving coordination in multi-agent systems [17] and we integrate this model by assuming an epistemically different point of view: there exists an objective reality not necessarily known by agents as it is. Indeed, with reference to Conte and Sichman formal theory of social dependence [17], we propose an updated model of dependence networks based on the evolution described in more recent works [14]. Therefore we extend the three basic notions of the first model, i.e. external description, dependence relationship and dependence situation. In particular, while we build external descriptions and dependence relationships following the instruction presented in the previous model of dependence network we consider some additional features of dependence situations in order to classify them. Together with the nature (given by dependence relations) and the locality (considered by Conte and Sichman), we add the distance between the locally believed dependence and the real dependence. In other terms, we extend the model by explicitly considering subjective and objective points of view in order to test how their distance influences agents' behaviors in the networks. Therefore, the cognitive model we used as theoretical framework takes into account both an objective dependence network, built on the real dependence relation between agents in the network, and multiple believed dependence networks (as many as the number of agents in the network).

We can consider the model as made up by an environment exogenously given, characterized by different dependence relations, and a set of goal oriented agents autonomous in making decision but dependent by other agents to reach their own goal. On the base of their locally believed dependence network, agents can proceed by trial and error in order to reach the goal (updating their own beliefs about the network in a process that can lead to reduce the divergence between it and the real environment).

As for this first attempt, in order to make it as simple as possible, we tested only a single type of dependence relationships among the set defined in [17], namely the mutual one, which means that agents depend on some of the others in the network to reach a common goal. The idea is to test interactions more and more complex once the simulator described in the next section is built and operative.

3 The Architecture for Simulation

Each agent in the environment has her own representation of the reality on the basis of her beliefs. She well knows her available actions and resources, and she starts to reach some goals, combining her actions and resources and/or asking some of them to other agents in the environment. In this way, she interacts with other agents either to perform an exchange of resources or to involve other agents in performing a specific action. The interaction is based on the believed dependence network, and it can be near or not to the reality. After every interaction the agent can update her believed dependence network on the basis of the information exchanged. The interaction among the agent happens every slot of time defined as rule of the environment (called round). For example: if an agent needs a resource or an action but she doesn't know who can perform that action or give her that resource, she sends the request to the environment (as a broadcast message). In this case every agent in the environment receives that request and can contact directly the requester or broadcast the answer (on the basis of the simulation design). In this manner all agents know what the applicant agent needs. At the same time the latter can discover a new dependence link with an agent (not necessarily a new link in general) from which she depends for a resource or an action. Furthermore everybody in the network, if the answer was broadcasted, may discover new information (new agent and/or new dependence link). In the end, this kind of information is stored in the work memory and afterwards it will be used to update the believed dependence network. We can therefore summarize as follows the research questions that guide our simulation: [RQ1] Does a knowledge of dependence network as much as possible close to the real configuration give an advantage to the agent that has that knowledge? In other terms can that agent reach the goal before the others? [RQ2] Is being part of a network where information is broadcasted an asset for agents in the network?

3.1 Agent Mind and Environment Configuration

Each agent has two kind of memories for storing several information, namely the Long Term Memory (LTM) and the Working Memory (WM). The former contains all information needed by the agent to act (goals, actions, plans, resources, etc.), whereas the latter is used to know what an agent will do in the next round and also to store the information produced by the interactions in the environment. This last kind of information may also be useful to update in the LTM the Believed Dependence Network by the agent. In particular the Long Term Memory (LTM) contains:

- a set of goals $G = \{g_1,\ldots,g_n\}$;
- a set of plans $P = \{p_1,\ldots,p_n\}$ where each p_i is the collection of actions to reach the goal g_i by the agent (each plan can be updated in runtime if needed);
- a set of possible actions $Act = \{a_1,\ldots,a_m\}$ where for each action one or more resources are associated
- a set of resources $R = \{r_1,\ldots,r_l\}$ owned by the agent
- a network dependence of the society built on the basis of the agent's beliefs
- a set of rules (Rules for Updating - RU) with which the information present in the Working Memory can be considered reliable and useful for updating the believed dependence network and the RU themselves.

As regards the Working Memory (WM), it contains the step of the plan to execute, and the stored information received (obtained during the interaction). The acting of agents is driven not only by the information stored in their own LTM but also by the information or constraints inherited from the environment. The environment settings contains:

- the real dependence network (it can be unknown to all the agents)
- the set of priority rules for executing some actions (not necessarily they involve all possible actions (i.e. some actions are executable independently from others)
- the set of possible resources needed for executing an action (i.e. some actions require using certain resources)
- the information about the latency between one round and the next one (an arbitrary technical requirement).

3.2 A Simple Scenario

As a first step of our research, we consider a simplified and generic scenario with the following assumptions:

- The Goal is the same for each agent: consuming resources following a given sequence; each agent can start from a different position in this sequence depending on the resources she has; she reaches the goal if the sequence is complete. Given the goal G={R1,R2,R3,R4} if the agent Ai has the set of resources R={R3,R5,R3,R2,R4}, she will start to find a combination of her resources as to reach the longest sequence she can: R2, R3, and R4; then she must start looking for

R1; once obtained R1 by another agent, she reaches the goal G (in this example in five rounds should she have received R1 in one interaction); resources not needed to achieve G, can be externally provided upon request (i.e. R3 and R5 in the example)

- The actions for each agent can be: Act = {consume a resource; ask for a resource to a given agent; ask for a resource through a broadcasting request}; for now we consider the only possibility for an agent to give the requested resource, when that resource is not needed by the owner (in which case it will be pre-allocated to be consumed by the latter in next rounds)
- Each agent can execute only one action in a given round
- The number of resources owned by the agents either is the same or it is higher than the number of resources for achieving the goal. In this scenario resources are generic items owned by the agents, in order to allow further specification in different application domains
- The unique set of given priority rules is related to the sequence of the resources to consume
- Each agent has her own believed dependence network (on which she bases her interactions)
- There is a unique updating rule: the new information collected in a certain round updates the believed dependence network and it will be used in the next round.

4 Simulation Design

According with the rules and assumptions described in the previous paragraph concerning the definition of a simple scenario, we describe here the simulation settings.

4.1 Environmental Settings

In the simulation each agent has the common goal G associated with a consume of resources that follows the sequence {①②③④⑤⑥⑦} in a circular way. This means that in case an agent starts in the middle, the previous sub-sequence is shifted to the end. As pointed out in previous section, possible actions of each agent are: allocate the resources (ALL), consume a resource (CON), ask for a resource to a given agent (ARA), ask for a resource as a broadcasting request (ARB), broadcast an answer (BRA), wait for the BRA. Moreover for each round each agent must perform only one of the following actions:

- ALL: in the first round all the agents strategically allocate their resources in order to reach their goal; the resources not allocated are made available for answering possible requests; in the next rounds each agent consumes the allocated resources starting from the resource that allows her to consume the longest sequence without asking for missing resources;
- CON: in each round an agent can consume one of the allocated resources or one received in the previous round, following the sequence imposed by the goal ($A_i(CON(R_j))$=agent Ai consumes resource R_j);

- ARA: if an agent needs a resource and knows who can give it to her then she will ask directly (A_i(ARA(R_j,A_k)= agent A_i directly ask for R_j to A_k); if A_k owns the requested resource A_i receives and consumes it in the same round by notifying the transaction to all the agents; if A_k does not have R_j, A_i has wasted a round;
- ARB: if an agent needs a resource and does not know who can give it to her, then she will broadcast the request (A_i(ARB(R_j) = agent A_i broadcast the request for R_j);
- BRA: when agents receive a broadcasted request they must answer with a broadcast message only in case they have that resource available (A_i(BRA(R_j)= agent A_i broadcasts her answer about the availability of R_j).
- WAIT: A_i(WAIT)= agent A_i wait for the answer

In this first simulation, note that if a resource is requested by multiple-agents, the agent who gets the resource will be randomly selected. This policy will be changed in future works, by adding variables able to influence the allocation of requested resources (such as level of trust, level of dependence etc.). Also the agent schedule for activation (i.e. who acts first) is random for this simple scenario but will be subject to modification in the future.

For the simulation run we illustrate, agents can update their BDN following the common rule that the new information are used at the end of each round for updating the believed dependence network. In future works, in order to understand some important phenomena such as how results are influenced by ordering agents' activation, we will modify the policy of random choice and we will adapt it to specific contexts. Furthermore, to give more straight to the model, we will test the implication of dealing with a bigger number of agents.

The initial condition is that there are 4 agents: A1, A2, A3, and A4 with the following sets of resources distributed randomly with the additional condition that the totality of resources is sufficient for making each agent reach her goal:

- R_{A1}={③③④④⑤⑥⑦}
- R_{A2}={②②③③⑤⑤⑥}
- R_{A3}={①①④④⑤⑥⑦}
- R_{A4}={①①②②⑥⑦⑦}

Fig. 1 shows the Real Dependence Network Graphs (RDN graphs), where each node represents the agent whereas ties and their labels represent the dependence links between agents concerning a specific set of resources. Considering the classification proposed in [17], in this scenario only a Mutual Dependence is allowed, because each agent has a potential dependence tie with everyone in the environment for the same goal, as shown in the figure below.

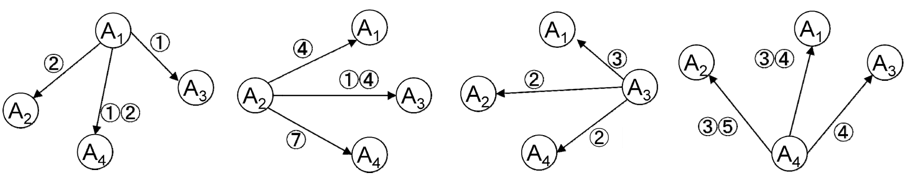


Fig. 1. Real Dependence Network Graphs

Starting from the RDN graphs we build an adjacency matrix named Real Dependence Network Matrix (RDN matrix) in which each row is related to one specific agent and contains the resources upon which the agent depends on the others. Since all ties show a mutual dependence, every cell (except those in the diagonal) contains at least one resource, as show in the table below.

Table 1. Real Dependence Network matrix

Really Needed	A1	A2	A3	A4
A1		②	①	①②
A2	④		①④	⑦
A3	③	②		②
A4	③④	③⑤	④	

The use of matrix for describing the dependence network is useful to represent the believed dependence network (BDN) for each agent acting in the environment, as show in the **Table 2**.

4.2 Agent Settings

The behaviour of an agent can be described as follows. The agent starts with the allocation of the resources (ALL), sorting them and defining the longest sequence from which to start for consuming resources. Afterwards, until the goal is achieved, she performs one of the following actions on the basis of her state. Either she answers to a broadcast request, or she proceeds towards her goal. In this latter case either she consumes one of the owned resources, or she asks for the resource to another agent assuming that the latter has that resource available. When an agent does not have the resource and does not to which agent to ask, she performs a broadcast request to all agents in the environment (this is the only action that requires a further round for waiting the broadcast answer). Therefore the agent's behaviour is led by the resource availability and by the believed dependence network.

In our scenario the four agents have a different perception of the environment. They do not know exactly their real dependence links with the other agents: they have not a complete vision of the dependence links and sometimes they have wrong assumptions. Following we resume the main aspects emerging from the observation of the dependence matrix for each agent in **Table 2**:

- agent A1 knows all the agents but she ignores one dependence link with A3 about the resource ①; furthermore she supposes wrong links about the five resources underlined in **Table 2.**
- agent A2 ignores the presence of agent A1 in the environment – grey column – and hence she ignores also the dependence link with A1 and her dependence link with A3 about resource ④;
- agent A3 is the unique agent that has a complete and exact vision of the dependence network (BDN is equal to RDN);

- agent A4 ignores the presence of agent A3 in the environment – grey column – and hence she ignores also the dependence link with A3 and her dependence link with A1 about resource ④.

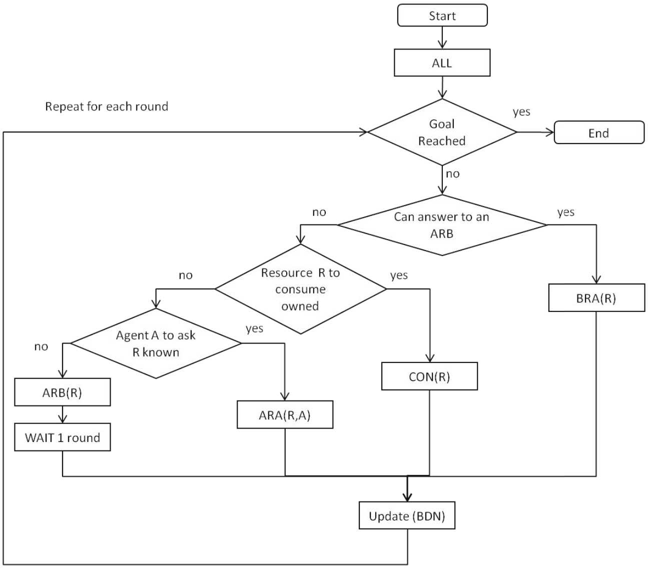


Fig. 2. Agent choice of action flow diagram

Table 2. The Believed Dependence Network Matrix for each agent in the environment

Really Needed	DD(A1)=12 BDN_{A1}				DD(A2)=6 BDN_{A2}				DD(A3)=0 BDN_{A3}				DD(A4)=8 BDN_{A4}			
	A1	A2	A3	A4	A1	A2	A3	A4	A1	A2	A3	A4	A1	A2	A3	A4
A1		②	④	① ②		②	①	① ②		②	①	① ②		②		①
A2	③		④	⑦			①	⑦	④		①④	⑦				
A3	④	②		②		②		②	③	②		②	③	②		②
A4	③ ④	④	⑤			③ ⑤			③④	③ ⑤	④		③	③ ⑤		

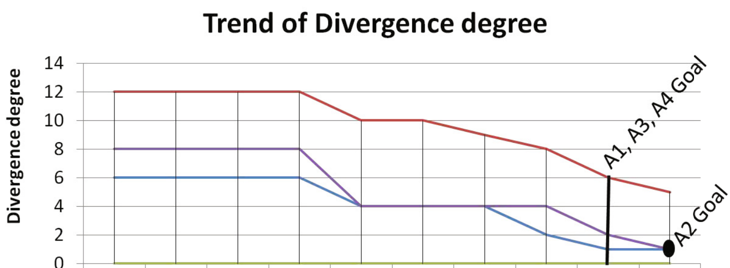
With this setting, during the simulation agent A2 or A4 may ask for the resource ④ because at the start time they do not know who can provide it in the environment, unless they acquire that information from previous interactions performed by others. In fact, as mentioned in the **Fig. 2** at the end of each round the agent can update her BDN on the basis of information arisen during the interaction among all the agents (respecting the unique updating rule RU).

4.3 Simulation Run

Taking into account the research questions mentioned in section 3, and the scenario described before, we formulate the following hypotheses to be tested in the simulation run:

- H1: A3 reaches the goal before the others;
- H2: A2 or A4 can update their knowledge through broadcasted answers, to the request "who has the resource ④?"; therefore either A2 or A4 can take advantage (one round) from the information circulated following a broadcast request from A4 or A2 respectively;
- H3: Round after round all the BDNs converge to the Real Dependence Network.

H1 and H2 concern advantages at the individual level, whereas H3 is related to a collective improvement (network performance). The figure below traces the results of the simulation until each agent reaches her goal. Round by round, the distance between an agent believed dependence network and the real dependence network, named Divergence Degree (DD), is represented. The DD is calculated as the sum of the missed links and the wrong links for each resource (i.e. if A1 ignores her dependence from A2 about the resources R5 and R6, the DD is equal to 2). An agent reaches the goal when she has consumed the sequence of resources (also shifted compared to the original one).



Rounds	1	2	3	4	5	6	7	8	9	10
Agent A1	12	12	12	12	10	10	9	8	6	5
Agent A2	6	6	6	6	4	4	4	2	1	1
Agent A3	0	0	0	0	0	0	0	0	0	0
Agent A4	8	8	8	8	4	4	4	4	2	1

Fig. 3. Simulation run

As depicted in **Fig. 3**, the simulation shows the convergence of the BDN to the real dependence network for all the agents, as supposed in H3. When all the agents have achieved their goals DD is equal to 5 (4 wrong links) for A1, 1 for A2 and A4, and 0 for A3. The decreasing of DD starts decreasing after the fourth round, when some agent start sending messages for retrieving resources. The most relevant rounds, in which a variation of DD takes place, are described below. In each round "DD(Ai)" indicates the DD for the agent "Ai".

- Round 5: agents A1 and A3 answer in broadcast to the A2's request for ④ made in the round 4; A2 and A4 discover their dependence link with A1 and A3, but A2 knows only her two links; A4 knows both her two links and also those of A2; therefore DD(A4) and DD(A2) decrease of 4 and 2 units respectively; furthermore A1 discovers that A2 does not have the resource ④, and that the latter depends on the former, so DD(A1) decreases of 2 units;
- Round 7: agent A4 takes ③ from A2; A1 discovers a dependence link between A4 and A2 gaining 1 unit for her DD;
- Round 8: among the three ARA actions performed in this round only the action of A4 (A4(ARA(④, A3))) produces an upgrade of some BDNs; in fact the DD(A1) and DD(A2) decrease of 1 and 2 units respectively;
- Round 9: agent A1 takes ② and agent A2 takes ⑦ from A4, implying a decrease of DD(A4) of 2 units; agent A3 takes ③ from A1 and A4 takes ⑤ from A2, producing a reduction of DD(A1) for 2 and DD(A2) for 1 units;
- Round 10: only A2 still need to achieve the goal and she takes ① from A3; this action allows to A1 and A4 to decrease their DD for another 1 units.

Considering the rounds described so far, H3 is confirmed, in fact BDN converge to RDN in all cases. Also H2 is confirmed, since the agent A4 benefits from the request asked by A2 (in the fourth round). Actually the agent A4 gains more than A2, in fact in that round she gains 4 units of DD while A2 only 2. The only hypothesis that is not confirmed is H1. In fact the agent A3, which knows from the initial state the overall RDN, does not reach the goal before all the other agents but at the same time as A1 and A4; only the agent A2 (the only agent that sent a broadcast request) achieves the goal one round after A3.

5 Discussion and Conclusion

The simulation run, performed as a pilot for the computer based simulation that is under development, allows to highlight some interesting dynamics of the dependence networks. Considering both the objective and the subjective point of views, such dynamics are characterized in different ways. In the RDN there are only updates related with the exchange of resources. In fact when an exchange is performed the related dependence tie disappears, producing an information update. In the BDN, dynamics are mostly characterized by the information exchanged when agents ask for resources by the means of an intentional communicative act (i.e. broadcast request).

Although in both cases the exchange of resources produces information, in the BDN a communicative level provides an additional means for learning and for achieving goals. In fact in RDN and BDN the exchange of resources implies, at the information level, an environmental update and a cognitive update respectively. Furthermore broadcast communication (requests and responses) allows agents to update their BDN and to reduce the gap with RDN. As consequence it represents an advantage to achieve the goal for agents in the network. This can be considered as a first positive answer to RQ2, to be deeply investigated in a more complex scenario.

Another aspect concerns the relationship between the cognitive representation of the real environmental configuration and the performance of the agent. The extent to which RDN and BDN overlap (DD) is intuitively related to the capabilities to reach the goal: the more they overlap the faster an agent achieve her goal. However the simulation shows that this behaviour does not holds in such environment: pilot results show that the advantage derived from having DD equal to 0 can be caught up by other agents thanks to the communicative level (e.g. broadcasting). Nevertheless additional evidences are needed to further investigate this phenomenon, and hence fully answer RQ1. A possibility is to consider more complex scenarios in which, for example, agents can deceive or ignore the requests.

By answering the two research questions, this study provides interesting insights on the behaviour of communities, both online and offline, in which members interact for achieving their personal goals. First, under the environmental settings modelled in our simulation, a complete knowledge of the real dependence network does not represent a sufficient condition for allowing members of the network to reach their goals in a shorter time frame with respect to other members whose perception of the resource distribution is far from the real situation. Second, being part of a network in which it is possible either to broadcast messages and to receive requests from the environment allows members of a community to exchange resources purposefully and to achieve their individual goals.

Clearly, since this is only a first attempt to use agent-based simulation for the domain under investigation, several improvements in next steps are already considered as necessary. Among them, as already pointed out, we are considering to test our model with a bigger number of agents involved, so that online and offline communities specific features can be taken into account. In fact, it can be hypothesized that the topology of the network will have great impact when considering larger networks. As well as the size and the type of networks, we will study the importance of different levels of dependence and different dynamics that can characterize different networks and communities (i.e. how agents enter and exit from the dependence networks can influence how quickly BDN and RDN converge).

These results have important implications for guiding the emergence of desired outcomes in the governance of online and offline communities. In fact the research shows the potential of adopting agent based models, which mix objectives and subjective views of the reality, for exploring the effects of structural characteristics of networks (environment) and of agents' cognitive models. Additional insights can be gained by representing other aspects of cognitive models and environmental constraints such as for instance trust and digital platforms functionalities (i.e. public and private groups, noticeboard, feeds, etc.). Furthermore, computational simulations can provide a means for exploring multiple trajectories of community behaviours in some specific domain by taking into account the complex nature of the phenomena under investigation and hence complementing with a constructivist approach the mainstream of positivistic studies of social networks. Therefore we argue that future simulation studies in management and information systems [18] can benefit from the proposed architecture in the design and evaluation of digital platforms and their governance models.

References

1. Spagnoletti, P., Federici, T.: Exploring the Interplay Between FLOSS Adoption and Organizational Innovation. Communications of the Association for Information Systems 29, 279–298 (2011)
2. Yoo, Y., Boland Jr., R.J., Lyytinen, K., Majchrzak, A.: Organizing for Innovation in the Digitized World 23, 1398–1408 (2012)
3. Resca, A., Spagnoletti, P., Za, S.: An attempt to outline strategic frameworks facing digital innovation. In: Innovation in Information Infrastructures (III 2012), Edinburgh, UK, October 9-11 (2012)
4. Shirky, C.: Here Comes Everybody: The Power of Organizing Without Organizations. Penguin Press, New York (2008)
5. Spagnoletti, P., Resca, A.: A Design Theory for IT supporting Online Communities. In: Proceedings of the 45th Hawaii International Conference on System Sciences, pp. 4082–4091 (2012)
6. Resca, A., Tozzi, M.: Offline and Online Communities: Great Differences and Some Similarities. In: Baskerville, R., De Marco, M., Spagnoletti, P. (eds.) Designing Organizational Systems. LNISO, vol. 1, pp. 310–318. Springer, Heidelberg (2013)
7. Spagnoletti, P., Baskerville, R., De Marco, M.: The Contributions of Alessandro D'Atri to Organization and Information Systems Studies. In: Baskerville, R., De Marco, M., Spagnoletti, P. (eds.) Designing Organizational Systems. LNISO, vol. 1, pp. 1–18. Springer, Heidelberg (2013)
8. Holland, J.: Emergence: From chaos to order. Basic Books, New York (1998)
9. Lewin, A.Y.: Application of Complexity Theory to Organization Science. Organization Science 10, 215 (1999)
10. Allen, P.M., Varga, L.: A co-evolutionary complex systems perspective on information systems. Journal of Information Technology 21, 229–238 (2006)
11. Amaral, L.A.N., Uzzi, B.: Complex Systems–A New Paradigm for the Integrative Study of Management, Physical, and Technological Systems. Management Science 53, 1033–1035 (2007)
12. Anderson, P.: Complexity Theory and Organization Science 10, 216–232 (2008)
13. Simon, H.: The Sciences of the Artificial. MIT Press (1996)
14. Castelfranchi, C., Falcone, R., Marzo, F.: Being Trusted in a Social Network: Trust as Relational Capital. In: Stølen, K., Winsborough, W.H., Martinelli, F., Massacci, F. (eds.) iTrust 2006. LNCS, vol. 3986, pp. 19–32. Springer, Heidelberg (2006)
15. Marzo, F., Castelfranchi, C.: Trust as individual asset in a network: A cognitive analysis. In: Spagnoletti, P. (ed.) Organization Change and Information Systems. LNISO, vol. 2. Springer, Heidelberg (2013)
16. Conte, R., Castelfranchi, C.: Simulating multi-agent interdependencies. A two-way approach to the micro-macro. In: Mueller, U., Troitzsch, K. (eds.) Microsimulation and the Social Science. Lecture Notes in Economics. Springer, Heidelberg (1996)
17. Conte, R., Sichman, J.S.: DEPNET: How to Benefit from Social Dependence. Journal of Mathematical Sociology 20, 161–177 (1995)
18. Za, S., Spagnoletti, P.: Knowledge creation processes in Information Systems and Management: Lessons from simulation studies. In: Spagnoletti, P. (ed.) Organization Change and Information Systems. LNISO, vol. 2. Springer, Heidelberg (2013)