AGENT-BASED SIMULATION ANALYSIS OF PATH DEPENDENCE IN CORPORATE IS NETWORKS FOR STRATEGIC IT MANAGEMENT

Daniel Fürstenau
Freie Universität Berlin
School of Business & Economics
Garystr. 21, 14195 Berlin, Germany
E-mail: daniel.fuerstenau@fu-berlin.de

KEYWORDS
Agent-based simulation, path dependence, strategic IT management, business information systems, IS networks

ABSTRACT

This study extends Arthur's model of path dependence for strategic IT management by introducing a more complex organizational structure with topology-adjusted network externalities and complementarities. A partial lock-in measure is developed to distinguish partial and global lock-ins. The study finds that complementarities can result in partial rather than global lock-ins. These findings suggest that, contrary to classical path dependence theory, strategic IT managers can contain global lock-ins by partitioning organizational structures. Results from an agent-based simulation study are presented and discussed.

1. INTRODUCTION

What are the features of path dependence in strongly coupled information system (IS) networks in organizations as opposed to technology markets? Previous simulation research tremendously improved our knowledge on path dependence (e.g. Arthur 1989; Leydesdorff and van den Besselaar 1997; Frenken et al. 2012), but focused mostly on technology markets. Generative mechanisms of path dependence inside complex business organizations are not yet sufficiently understood.

Applying simulation research in the field of complex business organizations is important, as will be illustrated at the example of the airline industry. In particular, interviews with airline managers suggest inertia to overcome rigidities in the pricing domain (Isler and D'Souza 2009). Pricing, distribution and other airline capabilities co-evolved and mutually adjusted for decades based on common standards, such as booking classes. This created and escalated a positive feedback loop; lock-in of the pricing capability resulted.

This extreme case carries several interrelated implications. One is that although existing models of path dependence enable a good understanding of market-based technology adoption, the airline case points directly to the appealing theoretical possibility that technological standards go on to diffuse into many interrelated domains *inside* an organization. Over time, this builds additional barriers to change. Thereby, managerial choice is a function of the value that agents assign to information systems and standards in a restricted neighborhood of complementaries rather than to the entire organization or market. Another implication is that agents' repeated choices to exploit a technology may appear beneficial in the short run but produce sticky local optima over time. I believe that an agent-based simulation model should figure centrally the effects of local complementarities and positive feedback to adaptation in a portrayal of path dependence in business organizations.

In this contribution, I propose an extension of Arthur's path dependence model using an agent-based simulation within the field of strategic IT management (refer to Ward and Peppard 2009 for general background). My primarily theoretical aim is to gain a better understanding of path dependence in corporate IS networks. The paper proceeds as follows. Section 2 introduces theoretical antecedents. Section 3 describes the model based on deviations from Arthur's efforts. Section 4 presents the experimental setup and results. The contribution ends with a conclusion and discussion of future research opportunities.

2. THEORETICAL BACKGROUND

2.1 Strategic Planning for Corporate IS Networks

The airline case illustrates the role of complementarities in creating IT value. This section briefly introduces a related technique from strategic IS planning. As shown in Figure 1, *master plans* conceptualize an organization as a two-dimensional matrix (a grid) where the x-axis represents business processes (or value chain elements) and the y-axis represents organizational units, e.g. divisions, departments or regions (Lankhorst et al. 2009).

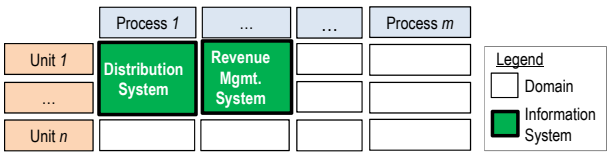

Figure 1: Master plan as *m x n* grid (processes and units)

The *m* x *n* grid positions, called domains, depict current and envisioned IS support. Master plans are used to identify complementarities: For instance, the airline IT manager from section 1 may propose to standardize

Proceedings 27th European Conference on Modelling and Simulation ©ECMS Webjørn Rekdalsbakken, Robin T. Bye, Houxiang Zhang (Editors)
ISBN: 978-0-9564944-6-7 / ISBN: 978-0-9564944-7-4 (CD)

information systems in neighboring domains, e.g. in sales/distribution ($process_1$) and pricing/managing revenues ($process_2$). Problems of path dependence in business organizations can ensue when IT managers overstress short-term opportunities from complementarities while overlooking potential long-term downsides.

2.2 Problems of Path Dependence in IS Networks

The field of strategic IT management offers numerous anecdotal examples of inert IT systems resulting in organizational "capability lock-ins" (Ross et al. 2006, p. 50) or "rigidity traps" (Bharadwaj 2000, p. 187). From these predecessors follows that the managerial scope of action rises and falls with contingent decisions on capabilities to develop by providing advanced IT systems. Scholarly work has, however, rarely explicitly modeled the conditions when and where in turn possible negative consequences of such capability-building processes will occur. Antecedents for this can be found in the literature on path dependence.

The Arthur model of *technological path dependence* (Arthur 1989), which is explained in the next section, shows when standards in markets of adopters can lock-in. Lock-ins occur when one technology gains momentum under positive network externalities and becomes dominant. Previous studies illustrate that path dependence related to network effects also impacts corporate IS networks (Weitzel et al. 2006) and that technical standards are incorporated in components forming "artifacts" (Widjaja 2011, p. 35). Network effect theory assumes that it gets more and more attractive for adopters to choose a blooming technology. Recent studies (Weitzel et al. 2006), considered in the model conception, found that the outcome is thereby influenced by the topology.

Building on the Arthur model, the *Berliner Modell* of organizational path dependence (Sydow et al. 2009) highlights several positive feedback mechanisms to explain lock-ins of social processes in organizations. Complementarities, defined as two or more activities that interact in mutually stimulating ways, are one key mechanism (Sydow et al. 2009). Organizational path dependence theory assumes that complementary processes show super-additive payoffs. In turn, deviations from this set of processes can become unattractive due to prior mutual adaptations.

These two streams of literature allow to distill two key feedback mechanisms expected to produce path dependence in organizational IS networks:

- Topology-adjusted network externalities and
- Complementarities

Consistent with this view, the following sections develop an agent-based simulation model of path dependence in organizations to address the following question: *When and where will topology-adjusted network externalities and complementarities in IS networks result in lock-ins?*

3. DESCRIPTION OF THE MODEL

Building on the airline case and the theoretical antecedents, this section formally models path dependence in corporate IS network. Starting with the Arthur model to account for general properties of path dependent processes, topology-adjusted network externalities and complementarities are examined regarding their influence on lock-ins.

3.1 Baseline Model

The Arthur model has long been used to illustrate problems of path dependence in technological markets, where the main ingredients are network effects *r* and *s* (see Table 1). These lead to lock-in when the parameters are positive (Arthur 1989). Arthur also informed models of innovation dynamics (e.g. Leydesdorff and van den Besselaar 1997; Frenken et al. 2012), standard diffusion in technological networks (e.g. Weitzel et al. 2006; Beck et al., 2008), and institutional rule adoption (e.g. Petermann 2010). The model functions as a starting point here as it aptly captures the path dependent features of technological standard diffusion processes resulting from individual agents' decisions. More specifically, the model assumes that two types of agents (*R* and *S* agents) enter a market sequentially and adopt a technology as described in Table 1.

Table 1: Utilities in Arthur's model of path dependence

	Technology A	Technology B
R-agent	$a_R + r*n_A$	$b_R + r*n_B$
S-agent	$a_S + s*n_A$	$b_S + s*n_B$

When applied to situations with two technologies, the model thus involves the following straightforward steps: each tick, the model creates a new agent of a certain type with corresponding base preferences (a_R or a_S) and network preferences (*r* and *s*), sums up the previous adopters for each technology (n_A and n_B), determines the winning technology (*A* or *B*), and increases the adoption counter (n_A or n_B) by one for the winner. The Arthur model captures the positive feedback from network influences and therefore serves as a constructive baseline for modeling standard diffusion in IS networks.

3.2 IS Network Structure

To implement the mechanisms described in section 2 in the model, the organizational IS network of a firm is conceptualized as shown in Figure 2.

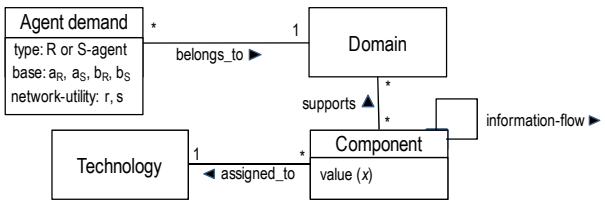

Figure 2: Meta model of IS network structure

The extension accounts for the organizational and the IS network dimension. The organizational dimension is expressed as a two-dimensional matrix (a grid), where each matrix cell represents a particular *domain* (see Figure 2). For instance, distribution is an exemplary airline domain. *Demands* of agents are assigned to exactly one domain. Agents evaluate a *set of components* available in their domain to fulfill the demand. Components are assigned to strictly one of two *technologies* (see Figure 2) and support one or more domains. Thus, domains are resource slots, housing multiple components with different qualities. Components are active or inactive, depending on agents' prior decisions. More specifically, components become active only after an agent decides to implement his or her demand using them. When activating a component, the agent's base utility is decreased by fixed setup costs. When agents adapt a component, its value increases and a special link type, called information-flow, is created (Figure 2). This works as follows: If the component is not active yet, increase the value by 1 and set its state to *active*; otherwise just increase the value. Consistent with predecessors in the field (e.g. Widjaja 2011), the network utility is conceptualized as the sum of the value of the components. In a network of components, the value of a component thus influences later decisions to choose other components based on the same technological platform.

Based on these descriptions, the agents' decision making can be defined in more detail. In Arthur's model, agents calculate the utility to choose technology A or B by summing up the base utility of these technologies (a_R, b_S) with the network effect strength (r, s) multiplied by the adoption count (n_A, n_B). Used by itself, however, the Arthur model has limitations. When applied to a given organizational IS network, it considers complete connectedness, making no allowance for the varying benefits flowing in from components with different positions in the IS networks. Although complete connectedness may accurately describe some IS structures, it is likely to mischaracterize most networks of interest to IT managers (Weitzel et al. 2006). Consequently, to build in more complex network influences into the model, I turn to topology-adjusted *network externalities* and *complementarities.* Their mechanics are now explained as the backcloth of the approach.

Topology-adjusted network externalities. Based on Arthur's utility function (see Table 1), the network terms are adapted to account for varying influences of the network structure. Table 2 shows the modified utility functions for R-agents and S-agents.

Table 2: Modified agents' utility functions

	Technology A	Technology B
R-agent	$a_R + r^* \sum_{i \in N(j)} x_{A,i}$	$b_R + r^* \sum_{i \in N(j)} x_{B,i}$
S-agent	$a_S + s^* \sum_{i \in N(j)} x_{A,i}$	$b_S + s^* \sum_{i \in N(j)} x_{B,i}$

Essentially, the only difference to Arthur's model is that not all adopters are connected. The departure from Arthur's model lies in the network influences (n_A and n_B), considered when choosing a component. The factors n_A and n_B for a given component i are replaced by the summed value of all neighboring components j based on the same technology A or B, where N again denotes the number of all active components. The important classification becomes that of neighborhood. More specially, two types of neighbors are distinguished: First order neighbors j_1 are components based on the same technology, and are connected to i by a flow of information. Second order neighbors j_2 are components based on the same technology, but are not (yet) connected to i by a flow of information. If we consider a hypothetical relational matrix X, where $\boldsymbol{X} = [x_{ij}]$, $x_{ii} = 0$ denotes the inflow of value between component i (self) and component j (neighbor) based on the same technology, then we can model each agents' decisions to invest in component i by extending the calculus of Table 2 as follows:

$$\sum_{i \in N(j)} x_{P,i} = w_1 \sum_{i \in N(j1)} x_{P,i} + w_2 \sum_{i \in N(j2)} x_{P,i} \text{ with } P\{A,B\} \quad (1)$$

Where w_1 and w_2 denote weighting terms that agents put on first order and second order neighbors. When applied to IS networks represented by a relational matrix $\boldsymbol{X}$, Equation (1) thus involves the following steps: Collect the first order neighbors j_1 of i, sum up their inflowing value x, and then collect the second order neighbors j_2 of i and sum up their inflowing value. If w_1 and w_2 are 0, i is completely autonomous. If, by contrast, the number of neighbors (j_1 and j_2) increases, then i will be more strongly influenced by other components. Again, the winning component is activated, the base utility is decreased by fixed setup costs, if necessary, and the function counter is incremented by 1.

Complementarities. To bring in complementary-based decision making, the factors n_A and n_B are adjusted.

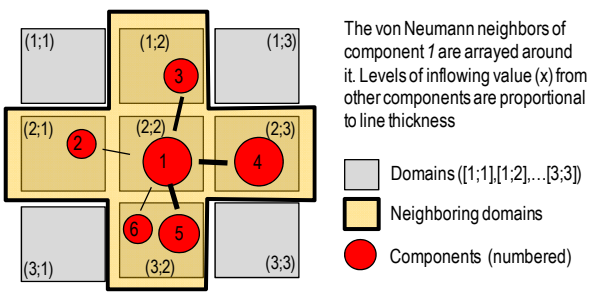

Figure 3: The von Neumann neighborhood

This time, the core idea is that agents only benefit from the value of components in neighboring domains. As illustrated in Figure 3, the agents' inflowing value from other components is restricted to von Neumann neighbors (Gilbert and Troitzsch 2005, p. 134). In a nutshell, the proposed model is best understood as a set of small-scale Arthur models. The calculation of the value $x_{P,i}$ for

a component is described in Equation (2), which extends the agents' utility functions of Table 2 as follows:

$$\sum_{i \in N(j)} x_{P,i} = \sum_{i \in N(j,vonNeumann)} x_{P,i} \quad \text{with } P\{A,B\} \quad (2)$$

This involves the following steps: Collect all components on the same technological platform (*A* or *B*) assigned to the von Neumann neighborhood domains. Sum up their value, determine the winning component, and set it active. As it is theorized that complementarities are often "mutually reinforcing" (Porter and Siggelkow 2009, p. 50; Sydow et al. 2009), the value of the chosen component *and* the value of all (technologically equal) components assigned to neighboring domains is incremented by 1. Focusing on the von Neumann neighbors thus limits the network effects. Consistent with the airline example (see section 1), agents will now strive for complementary IS support in neighboring domains.

Lock-In. In what follows, a measure for *partial lock-ins* is developed to separate partial from global lock-ins. Arthur (1989, p. 120) outlines an absorbing barrier, which is described as a situation where, for example, S-agents must switch to technology *A* despite their opposing preference for technology *B*, because the difference in adoption pushed technology *A* too far ahead. He shows that if $n_A - n_B > ((b_S - a_S)/s)$ holds, all further S-agents must choose *A* (and vice versa). It follows from the Arthur model that individual misfits boost lock-ins on the global level. It is thus a useful proxy for the expected number of global lock-ins (Draisbach et al. 2012). However, Arthur's absorbing barrier assumes complete connectedness. Consequently, to build in the network structure, replace n_A and n_B by the summed value of components relying on technology A ($x_{A,k}$) minus those on B ($x_{B,k}$) in a particular domain *k*. Agents in other domains rely on inflowing value from that domain. The restriction to domains accounts for the fact that agents are dragged to the leading technology of neighboring domains. For instance, a domain *distribution* (refer to coordinates [2;2] in Figure 3) might be locked-in, because it is strongly dominated by a global distribution system on GDS technology. Other domains, e.g. the *pricing* domain [2;1], might in turn become increasingly forced to adopt GDS technology as *distribution*. An agent in the *pricing* domain might thus decide to choose a system based on GDS technology, although he or she prefers another technology. Thus, if Equation (3) holds a *partial lock-in* in domain *k* for technology *A* is indicated:

$$\sum x_{A,k} - \sum x_{B,k} > \frac{b_s - a_s}{s} \quad (3)$$

Where $n_{k,A}$ ($n_{k,B}$) is a component in domain *k* on technology *A* (*B*) and *x* is the value of that component. Similarly, to pass the barrier of technology *B* in domain *k* the equation $sum(x_{A,k}) - sum(x_{B,k}) < (b_R - a_R)/r$ must hold. Note that partial lock-in is a binary variable that switches to 1 if the barrier is passed. It thus indicates when S-agents in neighboring domains are forced to choose *A* despite preferring *B*. Consequently, *global lock-in* occurres if the sum over the lock-ins of all domains divided by the number of domains is *1*. Note that the partial lock-in measure equals Arthur's absorbing barrier for settings with one domain, because if *k=1* then the value x_A (x_B) in domain *k* equals the value n_A (n_B) over the entire network.

3.3 Simulation Procedure

The next section presents simulation experiments to examine the effect of IS network structures on the likelihood of partial and global lock-ins (see Figure 4).

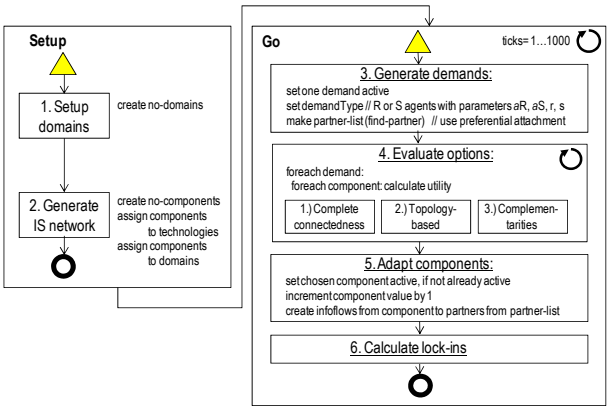

Figure 4: Simulation Procedure

At the start of the simulation (*step 1*), domains are set. The domains are modeled as a nested list that is a sublist of the list [1 1], [1 2], […], [m n]. Thereby, each domain is a two-dimensional vector with an x and y position. In model 1 and model 2 the number of domains is restricted to one domain. In model 3, the number of domains is varied iteratively (*i.e. 2, 5, 10, 15, 20, 25*). Afterwards, the following IS networks are created (*step 2*): (1.) In the baseline model, *two* components are created and assigned to exactly *two* technologies. As the components in the model are not linked and the model ensures that both technologies are available, the situation in model 1 is equal to the Arthur model. (2.) In model 2, *n* components (with *n = 5, 10, 25, 100)* are created and randomly assigned to strictly one of two technologies. Thereof, a random number of components is activated and linked using Erdös and Renyi's random network algorithm as presented by Jackson (2008) with a link probability of 0.1. (3.) In model 3, each domain is equipped with one active component based on technology A and B. Thus, the component number is double the domain number. Subsequently, the simulation is executed so that in step 3 of each round one agent (demand) of a particular type (R-agent or S-agent) is created (both types are equally probable) and assigned to strictly one domain. The agents' demand is set *active* and a list of potential link partners is created. Partners are found via the preferential attachment algorithm as described by Jackson (2008, p. 130 et seq.). The *find-partners* proce-

dure is repeated five times; duplicates are removed. In what follows (step 4), the agents evaluate options based on the decision logic described in the previous section: *(a)* complete connectedness as in the Arthur model, *(b)* topology-adjusted network externalities or *(c)* complementarities. After strictly one option is chosen in step 5, the component is activated, links are created, and value is incremented. Finally, step 6 calculates the lock-in measures.

4. SIMULATION RESULTS

4.1 Research Overview

This study uses an agent-based simulation approach (Gilbert and Troitzsch 2005). The concept was implemented with NetLogo 4.1.3. The work is currently in a prototypical stage.

4.2 Experimental Setup

The experiments were designed around three distinct models (see Table 3 for parameter descriptions).

Table 3: Parameter description for the experiments

	baseline model 1	model 2 (topology)	model 3 (compl.)
# ticks	1,000		
# demands	1,000 (per tick 1 demand is created)		
# techs.	2 (technologies A and B)		
# domains	1	1	2, 5, 10, 15, 20, 25
# components	2	5, 10, 25, 100	*2 * # domains*
link-prob	n.a.	0.1	
w_1, w_2	n.a.	(1 1), (1.0 0.5),(1.0 0.0)	n.a.
$r = s$	0.1		
a_R, a_S	R-agent: [0.8 0.2], S-agent: [0.2 0.8]		
setupCosts	0.0	2.0	0.0
# runs	100	3 * w_2 * 4 * #comp = 1,200	6*# domains* 100 = 600

Model 1, the baseline model, aims to confirm Arthur's findings and thus to increase the validity of the model and the lock-in measure. For model 2 and model 3, which extend the base model, a full-factorial design (Law and Kelton 1991, p. 656 et seq.) was chosen to develop an understanding of the solution space for experimental factors not yet analysed. In model 2, which examines the effect of *topology-adjusted network externalities* on lock-ins, the experimenter manipulates the factor *weighting term* (w_2) in 3 levels, controlling for the factor *number of components*. In model 3, under complementary-based decision making, the factor *number of domains* was varied in 6 levels.

4.3 Results

Model 1. Model 1 is intended to reproduce Arthur's findings. The plot **A** on the left in Figure 5, which shows the fraction of value of components on technological platform *A* vs. *B* against time, offers a portrayal of typical diffusion patterns in model 1. As suggested by Arthur's model, in model 1 initially one technology gains momentum and new agents are pulled towards the leading technology. As observable from the right-hand plot (**B**) in Figure 5, the misfit rate initially increases and then plateaus around 50% (*mean of misfit rate 0.48, std.dev. 0.023, N=100)*. Model 1 integrates previous efforts from Arthur's model as both the measure for partial lock-ins (see section 3.2) and the absorbing barrier from Arthur (1989, p. 120) equally produce global lock-ins over all simulation runs.

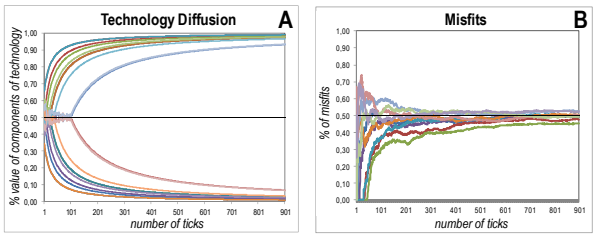

Figure 5: Typical diffusion patterns for model 1

Model 2 (Topology-adjusted network externalities). Here, the inflowing value x_i for component i is constrained to first and (less important) second order neighbors. Diffusion patterns as illustrated in Figure 6 were observed. In contrast to Figure 5, plot **A** and **B** of Figure 6 suggest that model 2 requires more ticks until lock-ins occur. It can be observed that agents, on average, were able to choose their preferred technology for more ticks.

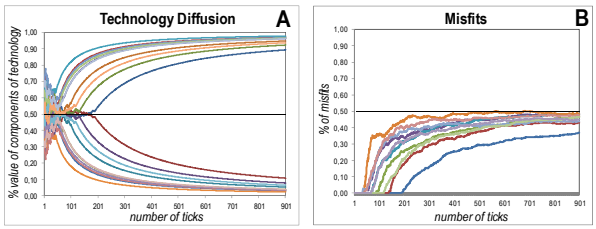

Figure 6: Diffusion pattern for model 2 (25 components)

Descriptive statistics derived from model 2 with three different weighting term factors w_1 and w_2 for 25 components appear in Table 4. Differing misfit rates across different weighting terms indicate the intensity of lock-ins to depend on the strength of the weighting factors *(F-stat.= 14.522, $p < 0.001$; Levene-stat.= 3.128; $p = 0.045$)*. Given the nature of the model, it is reasonable to assume that the decreasing strength of the network effects for lower factor levels of w_2 resulted in longer average time intervals until lock-ins occur.

Table 4: Descriptives for misfit rate in model 2

w_1,w_2	N	Mean	Std.dev.	Std.err.	Min	Max
1.0/0.0	100	0.4685	0.02205	0.00221	0.40	0.52
1.0/0.5	100	0.4706	0.02636	0.00264	0.38	0.51
1.0/1.0	100	0.4844	0.01887	0.00189	0.43	0.53
Total	300	0.4745	0.02364	0.00136	0.38	0.53

Model 3 (Complementarities). Two characteristic diffusion patterns were observed. Consider in this connection the three-by-three plot series (Figure 8), where the upper figures show diffusion and misfit plots for a setting with a global lock-in (**A-C**), and the bottom plots (**D-F**) illustrate runs with partial lock-ins. The outstanding feature of the bottom plots is the convergence to a fractional level for both technologies.

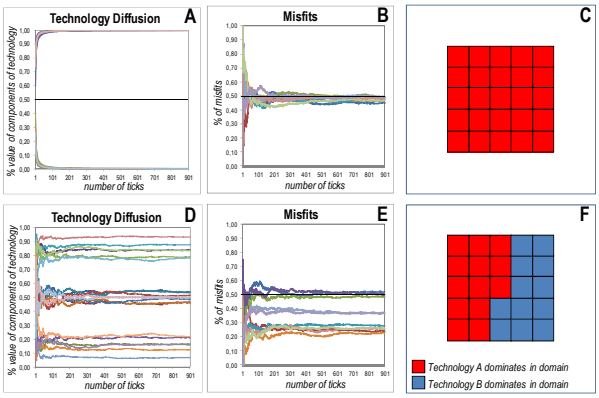

Figure 7: Typical diffusion patterns for model 3 (*illustrated for 25 domains, r = s = 0.1*)

Table 5: Global lock-in * no domains cross tabulation

	Count (for IS networks with 25 components)						
Global Lock-In?	no. domains						Total
	2	5	10	15	20	25	
No	0	11	14	19	32	44	120
Yes	100	89	86	81	68	56	480
Total	100	100	100	100	100	100	100

Global lock-in was found in 56-100% of the cases in model 3 depending on the number of domains (see Table 5, ^{2}*-stat. = 77,375, p < 0.001)*. The most important observation from Table 5 is the decreasing number of global lock-ins for larger number of domains. An analysis of the underlying decision patterns, as illustrated by plot **F** in Figure 8, revealed that convergence occurred despite different technologies dominating in the different domains. Plot F of Figure 8 illustrates a grid where 13 domains are dominated by technology *A* and 12 domains by technology *B*. Unlike model 2, where strong theoretical expectations about the expected outcome were available from the studies of Weitzel et al. (2006), there was less guidance available in the literature for anticipating the effect of increasing complexity on partial and global lock-ins. Yet, it is possible to sketch informal reasons for the decreasing number of global lock-ins based on the nature of the model. As larger grids have less overlaps – that is, as units are more dispersed in more complex organisations – initially unconnected areas on the grid come into existence and evolve separately. Additionally, as agents decide sequentially in different domains, more complex organizations will have more time to grow local clusters. It is thus possible that two distinct areas initially thrive and eventually sustain.

Robustness checks. To explore the robustness of the findings, I increased and decreased the strength of the network effects (r = s). For higher levels, the results might be weaker as increasing dynamics might boost global lock-ins. Interestingly, an increase of r = s by factor 10 (r = s = 1.0) instead produced a decreasing number of global lock-ins. The level dropped from 81% to 72% against a defined baseline (*15 domains, model 3, r = s = 0.1*), but no conclusive evidence for a significant deviation was observed (*t – stat. = 1.502, p = 0.135 > 0.05, Levene-stat. = 9.176, p < 0.003*). When decreasing the network effects by factor 10 (*r = s = 0.01*) the model was more prone to change - that is domains changed their orientation more often. The level of global lock-ins dropped significantly from 81% to 67% (*t – stat. = 2.275, p < 0.024, lev. stat. = 21.108, p < 0.001*).

5. CONCLUSION

This contribution aimed to advance our understanding of lock-ins in corporate IT infrastructures. The approach is framed on the backdrop of prior analysis of path dependence in technological markets. I extended prior models due to the importance of coupling among the nodes that compromise organizational IS networks. Combining approaches from strategic IT management literature with the Arthur model of path dependence, I developed a model that is attentive to the position components hold in the IS network. Using an agent-based simulation approach, initially replicating Arthur's findings and subsequently bringing in more complex structures as anticipated in the airline case, I illustrated the importance of organizational structures for the likelihood of global lock-ins.

By using an agent-based simulation approach, the (verbal) theory from Sydow et al. (2009) on complementarities was formalized. The proposed model confirmed the effect of complementarities on lock-ins. This highlighted an association that IT managers have to incorporate into their strategic thinking. My primary contribution is, however, to illustrate that *global lock-ins* become less likely in more complex organizational settings (see Table 5) and rather *partial* lock-ins occur in an increasing number of cases. This points to the theoretical possibility that lock-ins can be contained to particular domains to prevent their global spreading.

In the introduction, an example from the airline industry illustrated problems of path dependence inside complex business organizations. The model increases our understanding on the case in two directions. First, it turns attention to the importance of neighborhood in settings where complementarities exist. The model illustrates how local neighbors mutually adjust to each other and how interacting dynamics produce a positive feedback loop, which in turn often results in lock-ins of connected domains (see Figure 8). The model suggests that IT managers should use their knowledge on complementarities to group connected areas as pricing and distribution to profit from positive feedback, while they might

disconnect other areas deliberately, e.g. by dispensing the use of booking classes in bonus miling, to minimize lock-in risks. Secondly, the model also shows that sequence matters, as early decisions in different parts of the organization become reinforced by further adaption decisions and build up additional barriers to change over time. In this context, the model illustrated that global lock-ins might become less likely in more complex organizations (see Table 5), due to sequential decisions enabling different areas of the organization to gain enough momentum to resist a global standard. The model thus suggests that it might be possible to grow an alternative standard, e.g. a new distribution capability, in an independent "incubator".

I emphazise three conditions that limit the generalizability of the model. First, the partial lock-in measure from Equation (3) is preliminary. The measure is a first approach based on the difference in adoption for both technologies. With one domain, it follows directly from the agents' utility functions. For the *n* domain case, however, network effects from further domains might still perturbate the system. Second, the conclusion drawn from Table 5 that global lock-ins become less likely in complex organizational settings might be misleading when other structural conditions exist. The model only accounts for local complementarities and other models, e.g. with connections across *n* domains, could produce different results. Third, the model would benefit from a comparison with data, which is beyond the scope of this paper.

Ongoing work includes gathering exemplary IS network data from two airline companies' IT records to calibrate and validate the proposed model. In this, I employ a history-friendly validation approach (Windrum et al. 2005) for the problem instance of airline revenue management and distribution. Further research should model strategic agent behavior to test the effect of strategic IS planning procedures (Ward and Peppard 2009; Ross et al. 2006) on the likelihood of partial and global lock-ins. Instead of using a "shopping-list" approach (Ward and Peppard 2009, p. 121), where agents' demands are fulfilled sequentially, this should include balancing present and future needs as well as prioritizing agents' demands.

REFERENCES

Arthur, W.B. 1989. "Competing technologies, increasing returns, and lock-in by historical events", *The economic journal*, Vol. 99 No. 394, 116–131.

Beck, R., Beimborn, D., Weitzel, T. and König, W. 2008. "Network effects as drivers of individual technology adoption", *IS Frontiers*, Vol. 10 No. 4, 415–429.

Bharadwaj, A. 2000. "A resource-based perspective on information technology capability and firm performance: An empirical investigation", *MIS Q*, Vol. 24, 169–196.

Draisbach, T., Widjaja, T. and Buxmann, P. 2012. "Lock-Ins auf Netzeffektmärkten - Ergebnisse einer Simulationsstudie", in *MKWI 2012 Conf. Proc.*, 691–704.

Frenken, K., Izquierdo, L.R. and Zeppini, P. 2012. "Branching innovation, recombinant innovation, and endogenous technological transitions", *Environmental Innovation and Societal Transitions*.

Gilbert, G.N. and Troitzsch, K.G. 2005. *Simulation for the social scientist,* 2nd., Open Univ. Press, Maidenhead.

Isler, K. and D'Souza, E. 2009. "GDS capabilities, OD control and dynamic pricing", *Journal of Revenue and Pricing Management*, Vol. 8 2-3, 255–266.

Jackson, M.O. 2008. *Social and economic networks*, Princeton Univ. Press, Princeton, NJ.

Lankhorst, M. 2009. *Enterprise architecture at work: Modelling, communiation and analysis,* 2nd, Springer, NY.

Law, A.M. and Kelton, W.D. 1991. *Simulation modeling and analysis, McGraw Hill series in industrial engineering and management science,* 2nd ed., McGraw Hill, NY.

Leydesdorff, L. and van den Besselaar, P. 1997. "Competing technologies: Lock-ins and lock-outs", in *AIP Conf. Proc.* 437, pp. 309-323

Petermann, A. 2010. *Pfadabhängigkeit und Hierarchie*. Diss., Freie Univ, Berlin.

Ross, J.W., Robertson, D.C. and Weill, P. 2006. *Enterprise architecture as strategy: Creating a foundation for business execution*, HBS Press, Boston, Mass.

Sydow, J., Schreyögg, G. and Koch, J. 2009. "Organizational path dependence: Opening the black box", *Academy of Management Review*, Vol. 34 No. 4, 689–709.

Ward, J. and Peppard, J. 2009. *Strategic planning for information systems, Wiley series in information systems,* 3rd ed., Wiley, Chichester.

Weitzel, T., Beimborn, D. and König, W. 2006. "A Unified Economic Model of Standard Diffusion: The Impact of Standardization Cost, Network Effects, and Network Topology", *MIS Quarterly*, Vol. 30, 489-514.

Widjaja, T. 2011. "Standardisierungsentscheidungen in mehrschichtigen Systemen", Diss., Techn. Univ, Darmstadt.

Windrum, P., Fagiolo, G. and Moneta, A. 2007. "Empirical Validation of Agent-Based Models: Alternatives and Prospects", *JASSS,* Vol. 10 No. 2, p. 8.

DANIEL FUERSTENAU was born in Riesa, Germany, and went to the Universities of Potsdam, Germany, and Âbo, Finland, where he studied business administration and information systems. He worked as an IT consultant before moving to Freie Universität Berlin in 2011 where he is now working on his PhD in the field of simulation and path dependence. Correspondence may be directed to `daniel.fuerstenau@fu-berlin.de`. His web page is `http://www.pfadkolleg.de/fuerstenau`.