A JOURNAL OF THE INSTITUTE FOR OPERATIONS RESEARCH AND THE MANAGEMENT SCIENCES

informs

MANAGEMENT SCIENCE

Volume 53 • Number 7 • July 2007

Special Issue on Complex Systems

Luis A. Nunes Amaral, Brian Uzzi, *Editors*

MANAGEMENT SCIENCE
Vol. 53, No. 7, July 2007, pp. 1127–1145
ISSN 0025-1909 | EISSN 1526-5501 | 07 | 5307 | 1127

informs

DOI 10.1287/mnsc.1060.0617

The Statistical Mechanics of Complex Product Development: Empirical and Analytical Results

Dan Braha
Department of Decision and Information Sciences, University of Massachusetts, 285 Old Westport Road, Dartmouth, Massachusetts 02747 and New England Complex Systems Institute, 24 Mt. Auburn Street, Cambridge, Massachusetts 02138, braha@necsi.org

Yaneer Bar-Yam
New England Complex Systems Institute, 24 Mt. Auburn Street, Cambridge, Massachusetts 02138, yaneer@necsi.org

In recent years, understanding the structure and function of complex networks has become the foundation for explaining many different real-world complex biological, technological, and informal social phenomena. Techniques from statistical physics have been successfully applied to the analysis of these networks, and have uncovered surprising statistical structural properties that have also been shown to have a major effect on their functionality, dynamics, robustness, and fragility. This paper examines, for the first time, the statistical properties of strategically important organizational networks—networks of people engaged in distributed product development (PD)—and discusses the significance of these properties in providing insight into ways of improving the strategic and operational decision making of the organization. We show that the structure of information flow networks that are at the heart of large-scale product development efforts have properties that are similar to those displayed by other social, biological, and technological networks. In this context, we also identify novel properties that may be characteristic of other information-carrying networks. We further present a detailed model and analysis of PD dynamics on complex networks, and show how the underlying network topologies provide direct information about the characteristics of these dynamics. We believe that our new analysis methodology and empirical results are also relevant to other organizational information-carrying networks.

Key words: organizational studies; social networks; large-scale product development; sociotechnical systems; complex engineering systems
History: Accepted by Brian Uzzi and Luis Amaral, special issue editors; received August 31, 2004. This paper was with the authors 2 weeks for 2 revisions.

1. Introduction

The usefulness of understanding organizational network structure as a tool for assessing the effects of decisions on organizational performance has been illustrated in the social science and management literatures (Granovetter 1973, Krackhardt and Hanson 1993, Wasserman and Faust 1999). There it has been shown that informal networks of relationships (e.g., communication, information, and problem-solving networks)—rather than formal organizational charts—determine to a large extent the patterns of coordination and work processes embedded in the organization. In recent years, networks have also become the foundation for understanding numerous and disparate complex systems outside the field of social sciences (e.g., biology, ecology, engineering, and Internet technology (see Albert and Barabási 2002, Newman 2003, and Bar-Yam 1997)).

The goal of this paper is to examine, for the first time, the statistical properties of an important class of large-scale information-carrying networks—new product development. We discuss the significance of these statistical properties in providing insight into ways of improving the strategic and operational decision making of the organization. In general, information-carrying networks constitute the infrastructure for exchanging knowledge that is important to the achievement of work by individual agents. We believe that our results will also be relevant to other information-carrying networks.

Distributed product development (PD), which often involves an intricate set of interconnected tasks carried out by hundreds of designers, is fundamental to the creation of complex man-made systems (Alexander 1964). The interdependence between the various tasks makes system development fundamentally iterative (Braha and Maimon 1998). Iterations are driven by the repetition (rework) of tasks due to the availability of new information generated by other tasks, such as changes in input, updates of shared assumptions, components, boundaries, or the discovery of errors. In such a network of interactions,

iterations occur when some development tasks must be attempted even though the complete predecessor information is not available or known with certainty (Yassine and Braha 2003). As this missing or uncertain information becomes available, the tasks are repeated to come closer to the design specifications or goals. This iterative process proceeds until convergence occurs (Yassine and Braha 2003, Klein et al. 2006, Yassine et al. 2003).

Design iterations, which are the result of the PD network structure, might slow down the PD convergence or have a destabilizing effect on the system's behavior. This will delay the time required for product development, and thus compromise the effectiveness and efficiency of the PD process. For example, it is estimated that iteration costs about one-third of the whole PD time (Osborne 1993), while lost profits result when new products are delayed in development and shipped late (Clark 1989). Characterizing the real-world structure, and eventually the dynamics of complex PD networks, may lead to the development of guidelines for coping with complexity. It would also suggest ways for improving the decision-making process, and the search for innovative design solutions.

The last few years have witnessed substantial and dramatic new advances in understanding the large-scale structural properties of many real-world complex networks (Strogatz 2001, Albert and Barabási 2002, Newman 2003). The availability of large-scale empirical data on one hand and the advances in computing power and theoretical understanding have led to a series of discoveries that have uncovered statistical properties that are common to a variety of diverse real-world social, biological, and technological networks including the World-Wide Web (Albert et al. 1999), the Internet (Faloutsos et al. 1999), power grids (Watts and Strogatz 1998), metabolic and protein networks (Jeong et al. 2000, 2001), food webs (Montoya and Solé 2002), scientific collaboration networks (Amaral et al. 2000, Newman 2001), citation networks (de S. Price 1965), electronic circuits (Ferrer et al. 2001), and software architecture (Valverde et al. 2002). These studies have shown that many complex networks are sparse, that is, they have only a small fraction of the possible number of links. Despite being primarily locally connected, such networks exhibit the "small-world" property of short average path lengths between any two nodes. Studies also have found that complex networks are characterized by an inhomogeneous distribution of nodal degrees (the number of nodes a particular node is connected to), with this distribution often following a power law (termed "scale-free" networks in Barabási and Albert 1999). Scale-free networks have been shown to be robust to random failures of nodes, but vulnerable to failure of the highly connected nodes (Albert et al. 2000). A variety of network growth processes that might occur on real networks, and that lead to scale-free and small-world networks, have been proposed (Albert and Barabási 2002, Newman 2003). The dynamics of networks can be understood to be due to processes propagating through the network of connections (Bar-Yam and Epstein 2004); the range of dynamical processes include disease spreading and diffusion, search and random walks, synchronization, games, Boolean networks and cellular automata, and rumor propagation. Indeed, the raison d'être of complex network studies might be said to be the finding that topology provides direct information about the characteristics of network dynamics. In this paper, we study network topologies in the context of large-scale product development and discuss their relationship to the functional utility of the system, as well as to the dynamics of the underlying distributed design problem solving.

Planning techniques and analytical models that view the PD process as a network of interacting components have been proposed before (Braha and Maimon 1998, Yassine and Braha 2003, Klein et al. 2006, Yassine et al. 2003, Eppinger et al. 1994, Steward 1981, Mihm et al. 2003). However, others have not yet addressed the large-scale statistical properties of real-world PD task networks. In the research we report here, we study such networks. We show that task networks have properties (sparseness, small-world, scaling regimes) that are like those of other biological, social, and technological networks. We discover a distinctive asymmetry between the distributions of incoming and outgoing information flows (links) of PD networks, which has implications for their functionality, sensitivity, and robustness (error tolerance) properties.

We further present a model of PD dynamics embodying interactions through the network. Using analysis and simulation, we study its behavior to determine the conditions under which all the PD tasks are completed, and the rate of convergence to the completed state. We show that network topology provides key information about the characteristics of convergence, both whether and how rapidly convergence occurs. We find, quite reasonably, that the PD network dynamics will converge unless the total rate at which a task is affected by its neighboring tasks exceeds the "internal completion rate" of the task. Convergence is impeded by the existence of nodes that have high numbers of both incoming and outgoing information flows, i.e., convergence is controlled by the joint distribution of incoming and outgoing links. A more general result, which is presented in

Supplement 4 (provided in the e-companion),[1] shows that the characteristics of convergence depend on the incoming and outgoing information flows among multiple tasks.

This paper is organized as follows: In §2, we review the basic structural properties of real-world complex networks. In §3, we describe the PD data used in this paper. In §4, we present an analysis of the PD task networks, their small-world property, and node connectivity distributions. We demonstrate the distinct roles of incoming and outgoing information flows in distributed PD processes by analyzing the corresponding in-degree and out-degree link distributions. In §5, we present a dynamical model of PD processes on complex networks, and show analytically and numerically how the empirical structural properties bear on the PD dynamics. In §6, we present simulation results. In §7, we present our conclusions.

2. Structural Properties of Complex Networks

Complex networks can be defined formally in terms of a graph $G = (V, E)$, which is a pair of nodes $V = \{1, 2, \ldots, N\}$, and a set of lines $E = \{e_1, e_2, \ldots, e_L\}$ between pairs of nodes. If the line between two nodes is nondirectional, then the network is called *undirected*; otherwise, the network is called *directed*. A network is usually represented by a diagram, where nodes are drawn as points, undirected lines are drawn as edges, and directed lines as arcs connecting the corresponding two nodes. Three properties have been used to characterize real-world complex networks (Albert and Barabási 2002, Newman 2003). The first characteristic is the average distance (geodesic) between two nodes, where the distance $d(i, j)$ between nodes i and j is defined as the number of edges along the shortest path connecting them. The characteristic path length l is the average distance between any two vertices:

$$l = \frac{1}{N(N-1)} \sum_{i \neq j} d_{ij}. \tag{1}$$

The second characteristic measures the tendency of vertices to be locally interconnected or to cluster in dense modules. The clustering coefficient C_i of a vertex i is defined as follows: Let vertex i be connected to k_i neighbors. The total number of edges between these neighbors is at most $k_i(k_i - 1)/2$. If the actual number of edges between these k_i neighbors is n_i, then the clustering coefficient C_i of the vertex i is the ratio

$$C_i = \frac{2n_i}{k_i(k_i - 1)}. \tag{2}$$

The clustering coefficient of the graph, which is a measure of the network's potential modularity, is the average over all vertices,

$$C = \frac{1}{N} \sum_{i=1}^{N} C_i. \tag{3}$$

The third characteristic is the distribution of degrees of vertices. The degree of a vertex, denoted by k_i, is the number of nodes adjacent to it. The mean nodal degree is the average degree of the nodes in the network,

$$\langle k \rangle = \frac{\sum_{i=1}^{N} k_i}{N}. \tag{4}$$

If the network is directed, a distinction is made between the in-degree of a node and its out-degree. The in-degree of a node, $k_{\text{in}}(i)$, is the number of nodes that are adjacent to i. The out-degree of a node, $k_{\text{out}}(i)$, is the number of nodes adjacent from i.

Regular networks, where all the degrees of all the nodes are equal (such as circles, grids, and fully connected graphs), have been traditionally employed in modeling physical systems of atoms (Strogatz 2001). On the other hand, many real-world social, biological, and technological networks appear more random than regular (Strogatz 2001, Albert and Barabási 2002, Newman 2003). With the scarcity of large-scale empirical data on one hand, and the lack of computing power on the other hand, scientists have been led to model real-world networks as completely random graphs using the probabilistic graph models of Erdős and Rényi (1959).[2]

In their seminal paper on random graphs, Erdős and Rényi (1959) considered a model where N nodes are randomly connected with probability p. In this model, the average degree of the nodes in the network is $\langle k \rangle \cong pN$, and a Poisson distribution approximates the distribution of the nodal degree. In a Poisson random network, the probability of nodes with at least k edges decays rapidly for large values of k. Consequently, a typical Poisson random network is rather homogeneous, where most of the nodal degrees are concentrated around the mean. In particular, the average distance between any pair of nodes l_{random} scales with the number of nodes as $l_{\text{random}} \sim \ln(N)/\ln(\langle k \rangle)$. This feature of having a relatively short path between any two nodes, despite the often large graph size, is known as the *small-world effect* (de Sola Pool and Kochen 1978). In a Poisson random graph, the clustering coefficient is $C_{\text{random}} = p \cong \langle k \rangle / N$. Thus, while the average distance between any pair of nodes grows only logarithmically with N, the Poisson random graph is poorly clustered.

[1] An electronic companion to this paper is available as part of the online version that can be found at http://mansci.journal.informs.org/.

[2] For an earlier treatment of random graphs, see Solomonoff and Rapoport (1951).

Regular networks and random graphs serve as useful models for complex systems, yet many real networks are neither completely ordered nor completely random. Watts and Strogatz (1998) found that social, technological, and biological networks are much more highly clustered than a random graph with the same number of nodes and edges (i.e., $C_{\text{real}} \gg C_{\text{random}}$), while the characteristic path length l_{real} is close to the theoretically minimum distance obtained for a random graph with the same average connectivity. Small-world networks are a class of graphs that are highly clustered like regular graphs ($C_{\text{real}} \gg C_{\text{random}}$), but with a small characteristic path length like a random graph ($l_{\text{real}} \approx l_{\text{random}}$). Many real-world complex systems have been shown to be small-world networks, including power-line grids (Watts and Strogatz 1998), neuronal networks (Watts and Strogatz 1998), social networks (Amaral et al. 2000, Newman 2001), the World Wide Web (Albert et al. 1999), the Internet (Albert et al. 2000), food webs (Montoya and Solé 2002), and chemical-reaction networks (Jeong et al. 2000).

Another important characteristic of real-world networks is related to their nodal degree distribution. Unlike the bell-shaped Poisson distribution of random graphs, the degree distributions of many real-world networks have been documented to have power-law degree distribution,

$$p(k) \sim k^{-\gamma}, \tag{5}$$

where $p(k)$ is the probability that a node has k edges. Networks with power-law distributions are often referred to as *scale-free networks* (de S. Price 1965, Barabási and Albert 1999). The power-law distribution implies that there are a few nodes with many edges; in other words, the distribution of nodal degrees has a long right tail (resulting in an extremely large variance) of values that are far above the mean (as opposed to the fast decaying tail of a Poisson distribution, which results in a small variance). Power-law distributions of both the in-degree and out-degree of a node have also been observed in a variety of directed real-world networks (Albert and Barabási 2002, Newman 2003), including the World Wide Web, metabolic networks, networks of citations of scientific papers, and telephone call graphs. Although scale-free networks are prevalent, the power-law distribution is not universal. Empirical work shows that the total node degree distribution of a variety of real networks often has a scale-free regime with an exponential cutoff, i.e., $p(k) \sim k^{-\gamma} f(k/k^*)$, where k^* is the cutoff (Strogatz 2001, Amaral et al. 2000). The existence of a cutoff has been attributed to physical costs of adding links or limited capacity of a vertex (Amaral et al. 2000). In some networks, the power-law regime is not even present, and the nodal degree distribution is characterized by a distribution with a fast-decaying tail (Strogatz 2001, Amaral et al. 2000). It is also not clear that a scale-free network optimizes properties of network behavior, and alternatives have been proposed (Shargel et al. 2003).

The goal of this paper is to investigate the statistical properties of large-scale distributed product development networks. We show that large-scale PD networks, although of a different nature, have general properties that are shared by other social, technological, and biological networks.

3. Data

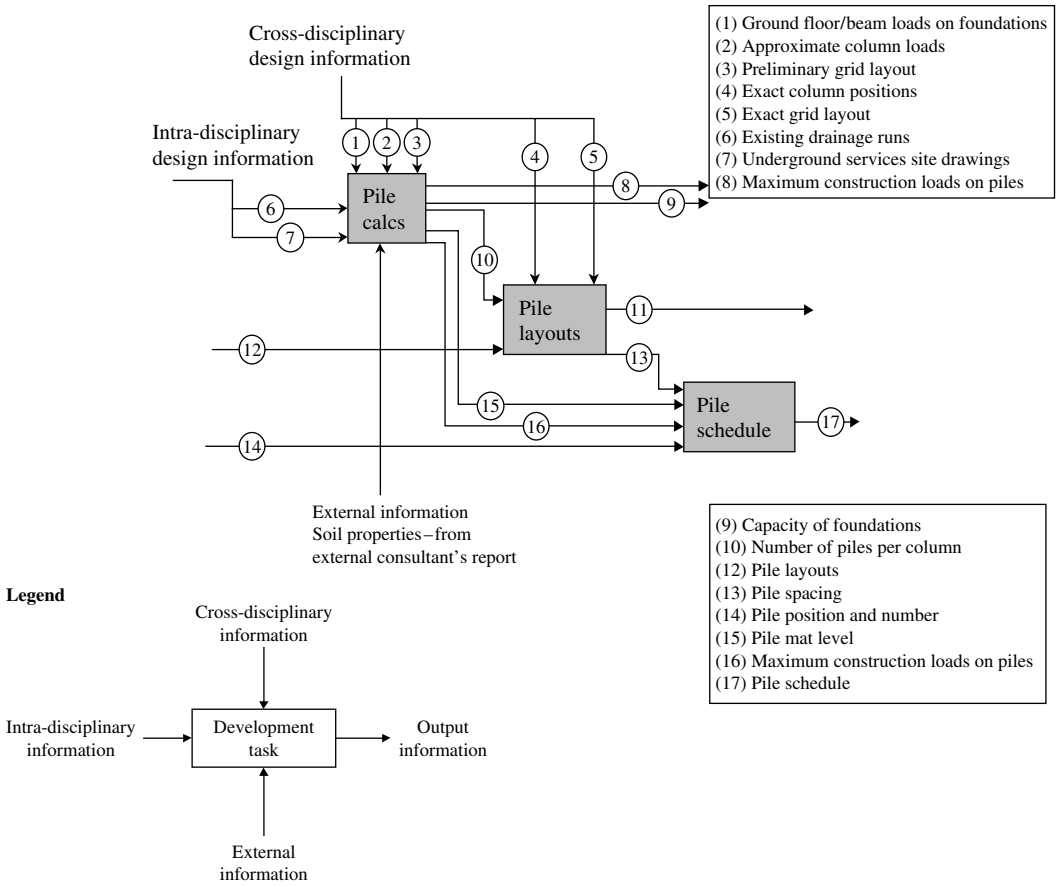
We analyzed distributed PD data of different large-scale organizations in the United States and England, involved in vehicle design (Cividanes 2002a), operating software design (Denker 2002), pharmaceutical facility design (Newton and Austin 2002), and a 16-story hospital facility design (Newton and Austin 2002). A PD-distributed network can be considered as a directed graph with N nodes and L arcs, where there is an arc from task v_i to task v_j if task v_i feeds information to task v_j. The documentation of the directed links between the tasks has been based on structured interviews with experienced engineers and design documentation data (design process models). In all cases, the repeated nature of the PD projects and the knowledgeable people involved in eliciting the information flow dependencies reduce the risk of error in the construction of the PD networks. More specifically, Cividanes obtained the vehicle development network by questioning in person at least one engineer from each task, "Where do the inputs for the task come from (e.g., another task)?" and "Where do the outputs generated by the task go to (e.g., another task)?" The answers to these questions were used by him to construct the network of information flows (Cividanes 2002b). The operating software development network was obtained from module/subsystems dependency diagrams compiled by Denker (2002); and both the pharmaceutical facility development and the hospital facility development networks were compiled by Newton and Austin (2002) from data flow diagrams and design-process model diagrams (Austin et al. 1999) deployed by the organizations. An example of a diagram from the pharmaceutical facility and 16-story hospital facility process models is shown in Figure 1.

4. Results

4.1. Small-World Properties

An example of one of these distributed PD networks (operating software development) is shown in Figure 2. Here we consider the undirected version of the

Figure 1 Example of a Diagram from a Design Process Model



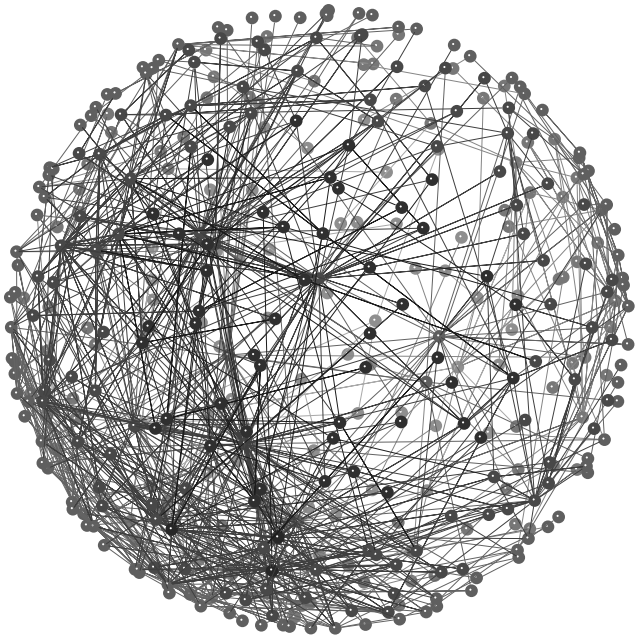
Note. Such diagrams were used to construct the pharmaceutical facility and the 16-story hospital facility networks (adapted from Austin et al. 2000).

network, where there is an edge between two tasks if they exchange information between them (not necessarily reciprocal communication). We see that this network is sparse ($2L/N(N-1) = 0.0114911$), with the average degree of each node only 5.34, which is small compared to the number of possible edges $N-1 = 465$. A clear deviation from a purely random graph is observed. We see that most of the nodes have low degree, while a few nodes have a very large degree. This is in contrast to the nodal degree homogeneity of purely random graphs, where most of the nodal degrees are concentrated around the mean. The software development network also illustrates the small-world property (see §2), which can be detected by measuring two basic statistical characteristics: (1) the average distance (geodesic) between two nodes, and (2) the clustering coefficient of the graph. Small-world networks are a class of graphs that are highly clustered like regular graphs ($C_{\text{real}} \gg C_{\text{random}}$), but with small characteristic path length like a random graph ($l_{\text{real}} \approx l_{\text{random}}$). For the software development network, the network is highly clustered, as measured by the clustering coefficient of the graph ($C_{\text{software}} = 0.327$) compared to a random graph with the same number of nodes and edges ($C_{\text{random}} = 0.021$), but with small characteristic path length like a random graph ($l_{\text{software}} = 3.700 \approx l_{\text{random}} = 3.448$).

In Table 1, we present the characteristic path length and clustering coefficient for the four distributed PD networks examined in this paper, and compare their values with random graphs having the same number of nodes and edges. In all cases, the empirical results display the small-world property ($C_{\text{real}} \gg C_{\text{random}}$ and $l_{\text{real}} \approx l_{\text{random}}$).

An interpretation of the functional significance of the architecture of PD networks must be based on recognition of the factors that such systems are optimizing. Shorter development times, improved product quality, and lower development costs are the key factors for successful complex PD processes. The existence of cycles in the PD networks, readily noted in the network architectures investigated, points to the seemingly undeniable truth that there is an inherent, iterative nature to the design process (Braha and Maimon 1998). Each iteration results in changes that must propagate through the PD network, requiring the rework of other reachable tasks. Consequently, late feedback and excessive rework should be minimized if shorter development time is required.

Figure 2 Network of Information Flows Between Tasks of an Operating System Development Process



Notes. This PD task network consists of 1,245 directed information flows between 466 development tasks. Each task is assigned to one or more actors ("design teams" or "engineers") who are responsible for it. Nodes with the same degree are colored the same.

The functional significance of the small-world property can be attributed to the fast information transfer throughout the network, which results in immediate response to the rework created by other tasks in the network. A high clustering coefficient is consistent with a modular organization, i.e., the organization of the PD process in clusters that contain most, if not all, of the interactions internally, and the interactions or links between separate clusters is minimized (Alexander 1964, Braha and Maimon 1998, Yassine and Braha 2003, Bar-Yam 1997). The dynamic model developed in Yassine et al. (2003) shows that a speed-up of the PD convergence to the design solution is obtained by reducing or ignoring some of the task dependencies (i.e., eliminating some of the arcs in the corresponding PD network). A modular architecture of the PD process is aligned with this strategy.

4.2. In-Degree and Out-Degree Distributions

We compared (see Figures 3(a)–3(d)) the cumulative probability distributions $P_{\text{in}}(k)$ and $P_{\text{out}}(k)$ that a task has more than k incoming and outgoing links, respectively.[3] For all four networks, we find that the in-degree and out-degree distributions can be described by power laws (the "scale-free" property), with cutoffs introduced at some characteristic scale k^*; $k^{-\gamma} f(k/k^*)$ (typically the function f corresponds to an exponential or Gaussian). More specifically, we find scaling regimes (i.e., straight-line regimes in the figure) for both $P_{\text{in}}(k)$ and $P_{\text{out}}(k)$. We note, however, that the cutoff k^* is lower by more than a factor of two for $P_{\text{in}}(k)$ than for $P_{\text{out}}(k)$.[4] It is worthwhile to note that the very low cutoff of the in-degree distribution exhibited by some networks (e.g., vehicle, software, and pharmaceutical) indicates that the in-degree distribution $k^{-\gamma} f(k/k^*)$ is broadly governed by the fast-decaying function f for $k > k^*$, implying that tasks with large incoming connectivity are practically absent. In this case, the in-degree distribution might be better fitted by an exponential as seen by the markedly curved-shaped behavior in Figure 3(c).

Table 1 Empirical Statistics of the Four Large-Scale PD Networks

Network	N	L	C	l	C_{random}	l_{random}
Vehicle	120	417	0.205	2.878	0.070	2.698
Operating software*	466	1,245	0.327	3.700	0.021	3.448
Pharmaceutical facility	582	4,123	0.449	2.628	0.023	2.771
16-story hospital facility*	889	8,178	0.274	3.118	0.024	2.583

*We restrict attention to the largest connected component of the graphs, which includes ∼82% of all tasks for the operating software network, and ∼92% of all tasks for the 16-story hospital facility network.

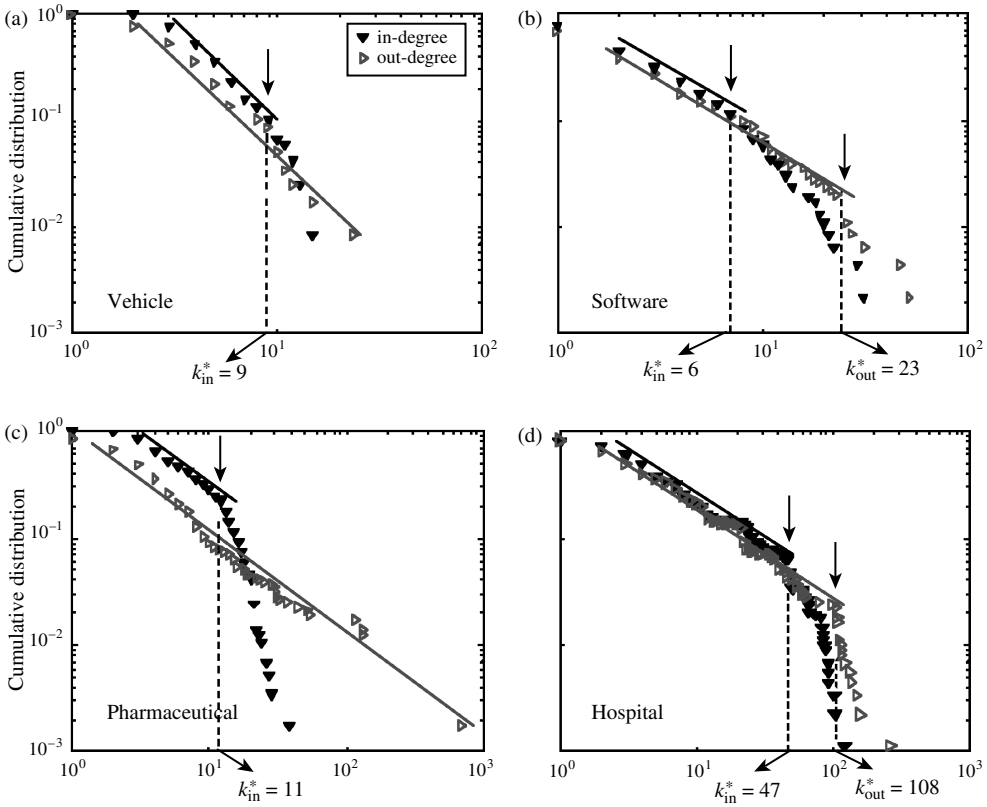
The scale-free property suggests that complex PD task networks are dominated by a few highly central tasks. This is in contrast to the bell-shaped Poisson distribution of random graphs, which leads to a fairly homogeneous network where each node has approximately the same number of links (and thus equally affecting the network behavior). The failure (e.g., excessive rework, lack of integration ability, or delays) of central PD tasks will likely affect the vulnerability of the overall PD process. Focusing engineering efforts and resources (e.g., funding and technology support) as well as developing appropriate control and management strategies for central PD tasks will likely maintain the sustainability and improve the performance of the PD process.

To analyze the structure of PD networks, it is important to study the relationships between the in-degree and out-degree of tasks. Thus, for example, we are interested in questions such as "Do tasks with high out-degree also have relatively high in-degree?" We address such questions by plotting the

[3] Note that a power-law distribution of the in-degree distribution (respectively, the out-degree distribution) $p_{\text{in}}(k) \sim k^{-\gamma_{\text{in}}}$ with exponent γ_{in} translates into a power-law distribution of the cumulative probability distribution $P_{\text{in}}(k) \sim \sum_{k'=k}^{\infty} k'^{-\gamma_{\text{in}}} \sim k^{-(\gamma_{\text{in}}-1)}$ with exponent $\gamma_{\text{in}} - 1$.

[4] This observation, verified for many real-world networks, has been previously reported in Braha and Bar-Yam (2004). See also Figure EC.4, which is provided as Supplement 5 in the e-companion, for an analysis of open source software and electronic circuit networks.

Figure 3 Degree Distributions for Four Distributed Problem-Solving Networks

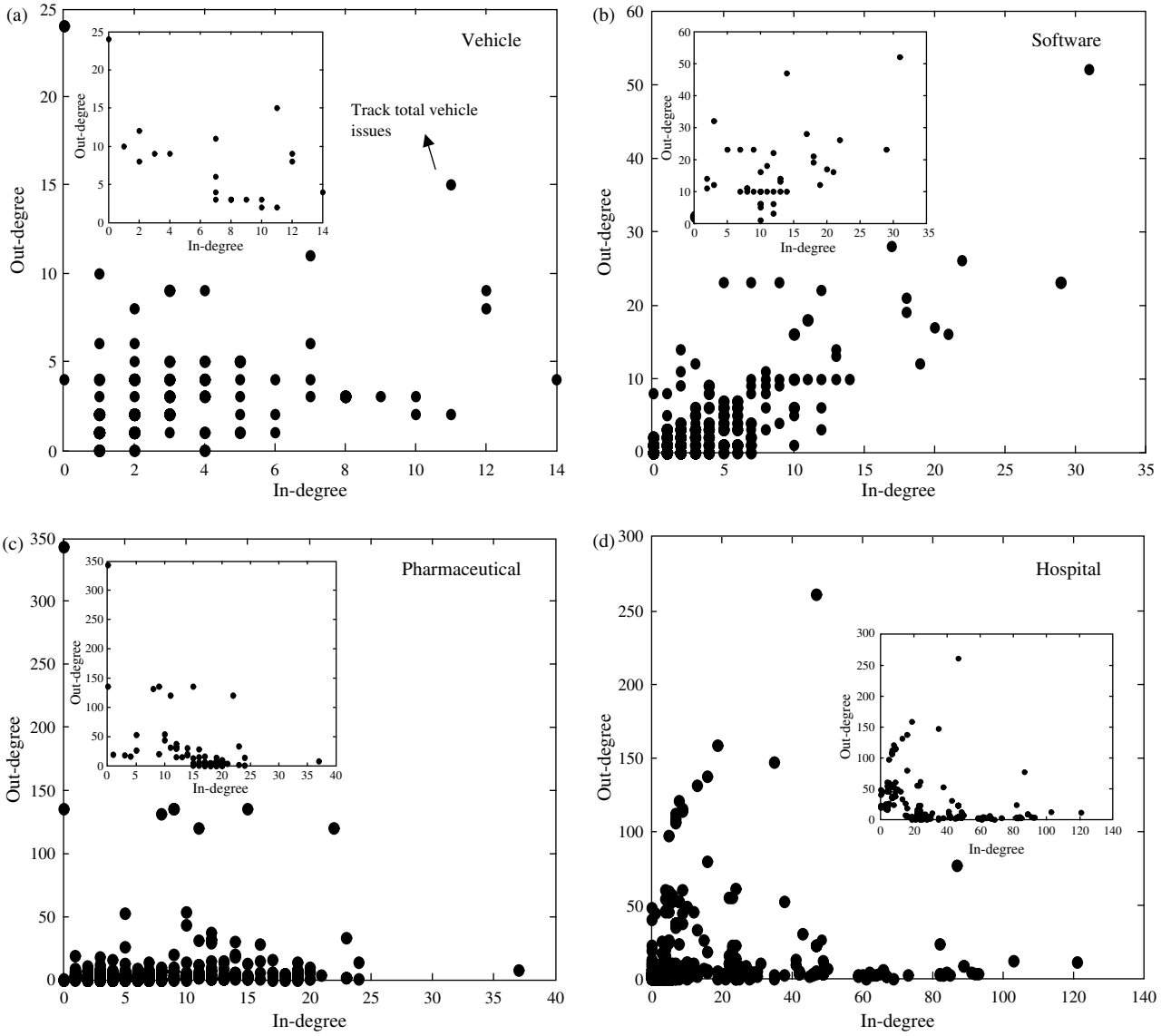


Notes. The log-log plots of the cumulative distributions of incoming and outgoing links show a power-law regime (Pearson coefficient $R > 0.98$, $p < 0.001$) with or without a fast-decaying tail in all cases. The in-degree distribution has a lower best visual fit cutoff k^*_{in} in each case. The very low cutoff of the in-degree distributions for the vehicle, software, and pharmaceutical networks suggests a single scale (characteristic of exponential decay) for the in-degree connectivities (see text). (a) Vehicle development with 120 tasks and 417 arcs. The exponents of the cumulative distributions are $\gamma^{in}_{vehicle} - 1$ and $\gamma^{out}_{vehicle} - 1$, where $\gamma^{in}_{vehicle} \approx 2.82 \pm 0.25$ and $\gamma^{out}_{vehicle} = 2.97 \pm 0.24$ denote the exponents of the associated probability density functions. (b) Software development with 466 tasks and 1,245 arcs, where $\gamma^{in}_{software} \approx 2.08 \pm 0.13$ and $\gamma^{out}_{software} \approx 2.25 \pm 0.15$. (c) Pharmaceutical facility development with 582 tasks and 4,123 arcs, where $\gamma^{in}_{pharmaceutical} \approx 1.92 \pm 0.07$ and $\gamma^{out}_{pharmaceutical} \approx 1.96 \pm 0.07$. (d) Hospital facility development with 889 tasks and 8,178 arcs, where $\gamma^{in}_{hospital} \approx 1.8 \pm 0.03$ and $\gamma^{out}_{hospital} \approx 1.95 \pm 0.03$.

relationship between the in-degree and out-degree of tasks (Figures 4(a)–4(d)). Figure 4 reveals that when considering the vehicle, pharmaceutical, and hospital PD networks, nodes with large in-degree generally have small out-degree, and vice versa. This kind of selective linking has been verified by calculating the Pearson coefficient between the in-degrees of tasks and their out-degrees; focusing on a reduced set of highly connected tasks for which their in-degree or out-degree exceed a certain threshold.[5] The calculation shows (see Figure 4) a noticeable negative correlation between in- and out-degrees for so-defined highly connected tasks. To further illustrate this finding, we present in Table 2 the top 10 tasks of the vehicle development network at General Motors' Research and Development Center, ranked according to their in-degree and out-degree centrality measures. We see that only two out of the 10 tasks (italicized in the table) appear both in the in-degree ranking and in the out-degree ranking. This finding implies that generally there is a clear distinction between large-scale generators of information (i.e., with high out-degree) and large-scale consumers (i.e., with high in-degree); a high generator of information could be a low consumer and vice versa. This further suggests that a distinction has to be made between in- and out-centrality as far as control and management strategies are concerned. Moreover, those tasks that have both high in- and out-centrality (e.g., "track total vehicle issues" at General Motors' vehicle design in Figure 4) are likely to play a unique role during the product design process. Interestingly enough, when considering the entire set of tasks (i.e., no filter on task connectivities is applied), the results for the vehicle, pharmaceutical, and hospital PD networks reveal almost no correlation between the in-degrees of tasks and their out-degrees. The dynamical model presented in §5 shows that the nature of in- and out-degree correlations has

[5] Somewhat arbitrarily, the threshold is defined as the maximum of the 90 percentiles associated with the in- and out-degree distributions.

Figure 4 In-Degree as a Function of Out-Degree for Four Distributed PD Networks



Notes. (a) Vehicle PD network. The main figure shows the correlation between the in-degrees of tasks and their out-degrees, when considering the entire set of tasks. Almost no correlation is observed (Pearson coefficient of 0.17). The Inset shows a significant negative correlation of k_{in} and k_{out}, when considering the reduced set of tasks with $k_{in} \geq 7$ or $k_{out} \geq 7$ (Pearson coefficient of -0.53). (b) Software PD network. The main figure shows a significant positive correlation for the full data set (Pearson coefficient of 0.76). For the reduced set with $k_{in} \geq 10$ or $k_{out} \geq 10$, a significant positive correlation is also observed (inset, Pearson coefficient of 0.43). (c) Pharmaceutical PD network. The main figure shows almost no correlation for the full data set (Pearson coefficient of 0.1). For the reduced set with $k_{in} \geq 15$ or $k_{out} \geq 15$, a significant negative correlation is observed (inset, Pearson coefficient of -0.52). (d) Hospital PD network. The main figure shows almost no correlation for the full data set (Pearson coefficient of 0.11). For the reduced set with $k_{in} \geq 20$ or $k_{out} \geq 20$, a significant negative correlation is observed (inset, Pearson coefficient of -0.34).

profound and subtle effects on the behavior of PD processes defined on top of complex networks. In particular, the model and simulation provides a theoretical explanation for the observed weak correlation at the level of all tasks, and anticorrelation at the level of highly connected tasks.

The presence of cutoffs in node degree distributions has been attributed to physical costs of adding links and limited capacity of a node (Amaral et al. 2000). Such networks may also arise if network formation occurs under conditions of preferential attachment with limited information (Mossa et al. 2002). As previously noted (Amaral et al. 2000, Mossa et al. 2002), the limited capacity of a node, or limited information-processing capability of a node, are similar to the so-called "bounded rationality" concept of Simon (1998).

We find that there is an *asymmetry* between the distributions of incoming and outgoing information flows. The narrower power-law regime for $P_{in}(k)$ sug-

Table 2 The Top 10 Tasks of the Vehicle Development Network at General Motors' Research and Development Center Ranked According to Their In-Degree and Out-Degree Centrality Measures

Task	In-degree	Task	Out-degree
Develop mainstream integrated concept vehicle model	14	Develop nine box summary	24
Maintain vehicle mainstream chart and update engineering product content sheet	12	*Track total vehicle issues*	15
		Set engineering target parameters (concept technical descriptors)	12
Conduct performance synthesis and analysis in quick study phase	12	Recommend final architecture	11
		Identify target architectures	10
Track total vehicle issues	11	Develop critical product characteristics/key voices	10
Review quick study deliverables	11	Develop engineering product content sheet	9
Assess risks in performance requirements	10	*Maintain vehicle mainstream chart and update engineering product content sheet*	9
Prepare program QRD matrix	10		
Follow up and maintain open issues—front compartment	9	Create initial visual surfaces	9
Follow up and maintain open issues—passenger/rear	8	Establish body BOM sharing strategy	9
Follow up and maintain open issues—chassis	8		

gests that the costs of adding incoming links and limited in-degree capacity of a task are higher than their counterpart out-degree links. We note that this is consistent with the realization that bounded rationality applies to incoming information, and to outgoing information only when it is different for each recipient, not when it is duplicated. This naturally leads to a weaker restriction on the out-degree distribution.

An additional functional significance of the asymmetric topology can be attributed to the distinct roles of incoming and outgoing links in distributed PD processes. The narrow scaling regime governing the information flowing into a task implies that tasks with large incoming connectivity are practically absent. This suggests that distributed PD networks limit conflicts by reducing the multiplicity of interactions that affect a single task, as reflected in the incoming links. Such architecture reduces the amount and range of potential revisions that occur in the dynamic PD process, and thus increases the likelihood of converging to a successful solution. Our empirical observation is found to be consistent with the dynamic PD model presented in the next section. There it is shown that additional rework might slow down the PD convergence or have a destabilizing effect on the system's behavior. As a general rule, the rate of problem solving has to be measured and controlled such that the total number of design problems being created is smaller than the total number of design problems being solved.

The scale-free nature of the outgoing communication links means that some tasks communicate their outcomes to many more tasks than others do, and may play the role of coordinators (or product integrators, see Yassine et al. 2003). Unlike the case of large numbers of incoming links, this may improve the integration and consistency of the problem-solving process, thus reducing the number of potential conflicts. Product integrators put the separate development tasks together to ensure fit and functionality. Because late changes in product design are highly expensive, product integrators continuously check unfinished component designs and provide feedback to a large number of tasks accordingly.

5. A Dynamical Model on Complex PD Networks

This section introduces a deliberately simple model of product development on complex directed networks, which captures important features of PD dynamics (e.g., see Yassine and Braha 2003). We characterize the model's behavior by using analysis and simulations performed on the empirically heterogeneous directed network topologies examined in §4 (rather than on simplified fully connected or lattice topologies). In our model, there is a network of interconnected nodes (elemental tasks); each can be in a "resolved" or "unresolved" state. Each node could be affected by those nodes that directly reach it, and could affect those nodes that are directly reachable from it. The rule by which a resolved node becomes unresolved depends stochastically on the number of unresolved contiguous incoming nodes—the higher the number of unresolved neighbors, the higher the probability of becoming unresolved. This rule reflects the repetition (rework) of tasks due to the availability of new information and input changes generated by other contiguous tasks. An unresolved node may be fully resolved with probability that depends on both its self-completion rate (internal problem-solving rate), and on the number of unresolved neighboring nodes. Incorporating the effect of task j on task i (which possibly differs between each pair of tasks) as well as including nonbinary states (e.g., the number of design problems or open issues associated with a task) can be readily done but does not offer additional understanding on the issues addressed here. Although the motivation is different, it is worthwhile to note that the model considered here is similar in

spirit to dynamic models that have been studied in the context of collective action, percolation, majority-vote cellular automata, self-organized criticality, spin-flip Ising dynamics, and epidemic spreading.

5.1. Model

We consider a network where each node (a task in the network) can be in one of two states, zero or one, representing unresolved or resolved states, respectively. We consider a dynamic process occurring at discrete times, $1, 2, \ldots, t$. Node states are updated synchronously, indicating a parallel mode of product development.[6] Let $s_i(t)$ be the state of node i at time t. We consider two cases:

Case 1. Node i is resolved at time t (i.e., $s_i(t) = 1$). Let k_i^{in} be the in-degree connectivity of node i, and let $k_i(t) = k_i^{\text{in}} - \sum_{j:\langle j,i\rangle \in E} s_j(t)$ be the number of neighboring unresolved nodes that are directly connected by directional links (arcs) to node i at time t. Node i changes its state according to the following stochastic rule:

$$s_i(t+1) = \begin{cases} 0 & \text{with probability } \tanh(\beta_i k_i(t)), \\ 1 & \text{with probability } 1 - \tanh(\beta_i k_i(t)), \end{cases} \tag{6}$$

where β_i is a parameter that reflects the sensitivity of the node i state to its neighboring unresolved nodes, and $\tanh(x)$ is the hyperbolic tangent function defined by

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{7}$$

The stochastic dynamic rule allows for node state realizations to vary over time even if the node has the same number of unresolved neighbors at different times. The parameter β_i captures the tendency of a node to be affected by its neighbors. For $\beta_i = 0$, the node's behavior is completely decoupled from its neighbors. A low β_i corresponds to the case where a node's behavior is not influenced much by the states of its neighbors. For $\beta_i \to \infty$, each node's behavior is completely dependent on its neighbors: Any nonzero number of unresolved neighbors will render the node unresolved at the next iteration.

Case 2. Node i is unresolved at time t (i.e., $s_i(t) = 0$). In this case, node i changes its state according to the following stochastic rule:

$$s_i(t+1) = \begin{cases} 0 & \text{with probability } 1 - r_i\ (1 - \tanh(\beta_i k_i(t))), \\ 1 & \text{with probability } r_i\ (1 - \tanh(\beta_i k_i(t))), \end{cases} \tag{8}$$

[6] Note that this assumption can be easily relaxed by randomly selecting, at each time point, a node for an update.

where r_i is a parameter that reflects the internal completion rate of task i ($0 \le r_i \le 1$). Here we assume that the node can be resolved if two events occur: (1) the node is not affected by its unresolved neighbors, and (2) the task is successfully completed internally in one unit of time with probability r_i.

Although the model presented above captures important aspects of PD dynamics, it is worthwhile to discuss some of the underlying assumptions:

The stochastic rule (Equations (6) and (8)) governing the response of each task could be modified in various ways. Still, it is easy to show that the qualitative results of the model will be preserved for a broad family of concave response functions that have continuous second derivatives, and guarantee probabilities in the $[0, 1]$ range. The concavity assumption of the response function gives a plausible shape for the marginal effects; that is, at relatively high values, a marginal change in the number of unresolved contiguous incoming nodes will give a relatively smaller modification in the probability of state change.

The model assumes that each task is equally influenced by its neighboring tasks, thus taking into account only the topological structure of the PD network. This assumption could be relaxed by having some tasks more influential than others. This could be formalized by assigning to each arc of the PD network a weight proportional to the dependency strength of the connections among the various tasks of the network. The total weight of a task's connectivities could then be incorporated into a response function. An interesting question here is, What is the dynamic effect of such weighted PD network architectures? Although the answer to this question requires further empirical and analytical study, it is plausible that the main findings reported here (e.g., cutoff asymmetry, and the robustness and sensitivity properties presented in §6) will remain valid for weighted PD networks as well. Evidence for this can be found in recent empirical studies that show some real-world networks (the worldwide airport network and scientific collaboration networks) exhibit strong correlations among weighted quantities and the underlying topological structure of the network (Barrat et al. 2004). More specifically, it has been shown that the strength of a node (i.e., the total weight of their connections) increases with the node degree, and that the functional behavior of the strength distribution exhibits similarities with the degree distribution (Barrat et al. 2004). Consequently, a node centrality measure based only on topological elements might provide useful information about the characteristics of network dynamics.

In our model, we assume that the resource usage intensity required to accomplish the internal completion rate of the various tasks is uniform throughout the project. Adjustments in effort levels via resource

allocation and workload policies could be taken into account by introducing internal completion rates that are time based. Finally, in our model, we assume that the product content, performance, variable cost, investment, quality, and other project attributes do not change over the duration of the project. Project changes could be quantified by introducing an "external field" into the model.

5.2. Analytic Results for Random (Erdős-Rényi) Networks

The relaxation of the system to the uniformly resolved state (i.e., $s_i(t) = 1$ for all tasks) depends on the free parameters β_i, r_i, the initial state of the network, and the PD network topology. Although there is no theorem guaranteeing the relaxation of the network to the uniformly resolved state, we apply a mean-field approximation (Marro and Dickman 1999) to the stochastic model we have defined to gain insight about the convergent final state. We derive a rate equation for the density of unresolved tasks at time t,

$$\alpha(t) = 1 - \sum_i s_i(t)/N.$$

We assume that for every task i, $r_i = r$, and $\beta_i = \beta$. We also make the following homogeneity condition, which holds particularly well for a completely random graph: For every task i, its number of unresolved neighbors is approximately $k_i(t) \cong \langle k_{\text{in}}\rangle \alpha(t)$, where $\langle k_{\text{in}}\rangle$ denotes the average in-degree of a task in the network. The global density of unresolved tasks $\alpha(t)$ evolves according to the rate equation

$$\frac{d\alpha(t)}{dt} = (1-\alpha(t))\tanh(\beta\langle k_{\text{in}}\rangle\alpha(t)) - \alpha(t)r(1-\tanh(\beta\langle k_{\text{in}}\rangle\alpha(t))). \quad (9)$$

After substitution of the hyperbolic tangent function and $\bar{\beta} = \beta\langle k_{\text{in}}\rangle$, we obtain

$$\frac{d\alpha(t)}{dt} = (1-\alpha(t))\frac{e^{\bar{\beta}\alpha(t)} - e^{-\bar{\beta}\alpha(t)}}{e^{\bar{\beta}\alpha(t)} + e^{-\bar{\beta}\alpha(t)}} - \alpha(t)r\frac{2e^{-\bar{\beta}\alpha(t)}}{e^{\bar{\beta}\alpha(t)} + e^{-\bar{\beta}\alpha(t)}}.$$

At an equilibrium $d\alpha(t)/dt = 0$; thus, we obtain a single equation to be solved for $\alpha(t) = \alpha$:

$$\alpha = \frac{e^{\bar{\beta}\alpha} - e^{-\bar{\beta}\alpha}}{e^{\bar{\beta}\alpha} - e^{-\bar{\beta}\alpha} + 2re^{-\bar{\beta}\alpha}} = f(\alpha). \quad (10)$$

We conclude that the only stable fixed point of this equation is $\alpha^* = 0$ if $f'(\alpha)|_{\alpha=0} < 1$, and has a nonzero solution if $f'(\alpha)|_{\alpha=0} > 1$. Thus, for $\bar{\beta} < r$, the global density of unresolved tasks at equilibrium is $\alpha^* = 0$ (i.e., all tasks are successfully completed). This result has a simple intuitive interpretation: If the task's internal completion rate r exceeds the average total sensitivity of the task to its unresolved incoming neighbors $\bar{\beta} = \beta\langle k_{\text{in}}\rangle$, the PD process will converge to the uniformly resolved state; otherwise, it is quite likely that the PD process will converge to a state where a nonzero fraction $\alpha^* > 0$ of the tasks remains unresolved.[7]

When the fraction of unresolved tasks at equilibrium is very small ($\alpha^* \ll 1$), we can find a closed-form expression for α^* by expanding $f(\alpha)$ in Equation (10) to the second order in α and solving for α^*:

$$f(\alpha) = \frac{\bar{\beta}}{r}\alpha + \frac{\bar{\beta}^2(r-1)}{r^2}\alpha^2 + O(\alpha^3). \quad (11)$$

Hence,

$$\alpha^* \approx \frac{r(\bar{\beta}-r)}{\bar{\beta}^2(1-r)}. \quad (12)$$

To gain further insight regarding the rate of convergence to the fixed point of Equation (9), we solve it approximately as follows. For small values of α, Taylor's expansion yields

$$\tanh(\bar{\beta}\alpha) = \bar{\beta}\alpha + O(\alpha^3).$$

Thus, the differential Equation (9) is approximated by

$$\frac{d\alpha}{dt} = (1-\alpha)\bar{\beta}\alpha - \alpha r(1-\bar{\beta}\alpha). \quad (13)$$

The solution of Equation (13) is

$$\alpha(t) = \frac{\bar{\beta}-r}{re^{(r-\bar{\beta})t}(\bar{\beta}-1) + \bar{\beta}(1-r)}. \quad (14)$$

For $\bar{\beta} < r$, $\lim_{t\to\infty}\alpha(t) = 0$ and the system converges exponentially to the uniformly resolved state. For $\bar{\beta} > r$, the fraction of unresolved tasks decays at an exponential rate and eventually saturates at a nonzero fraction of unresolved tasks

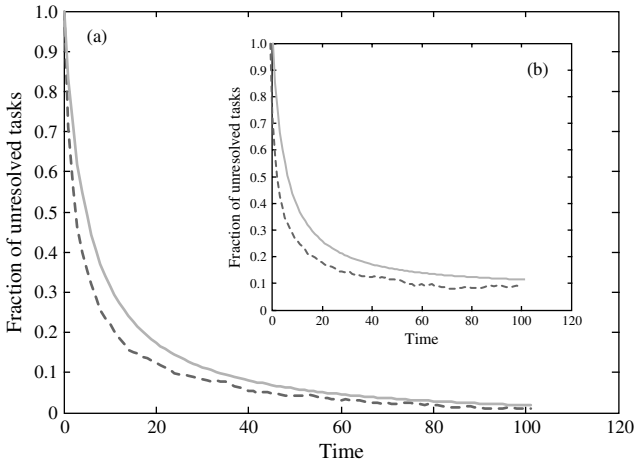
$$\alpha^* = \lim_{t\to\infty}\alpha(t) = \frac{\bar{\beta}-r}{\bar{\beta}(1-r)}$$

(note that when α^* is small, as assumed, $\bar{\beta}/r \approx 1$, and thus the prediction of Equation (12) is consistent with the estimate above).

The deterministic analysis of the model has involved a number of assumptions that can be tested by simulation. First, a directed random graph with a prescribed average in-degree of tasks (and same average out-degree) has been generated, and all tasks have been initially selected to be unresolved. The graph contains 10^5 tasks with connectivity $\langle k_{\text{in}}\rangle = \langle k_{\text{out}}\rangle = 12$. We have simulated the model using a synchronous discrete-event implementation. Figure 5 compares a typical simulation run to the corresponding deterministic solution (14). The simulation run has followed

[7] In other words, a threshold behavior occurs at $\beta\langle k_{\text{in}}\rangle/r = 1$.

Figure 5 **Comparison Between Average Fraction of Unresolved Tasks vs. Time as Predicted by Deterministic Theory (Solid Curve) and a Typical Simulation Run (Broken Curve) on a Randomly Generated Graph with 10^5 Nodes**



Notes. The average number of incoming arcs connected to a node is $\langle k_{\text{in}} \rangle = 12.019$. In (a), the time evolution of $\alpha(t)$ when the sensitivity and the internal completion rates of tasks are $\beta = 0.061$ and $r = 0.75$, respectively. In this case, $\beta\langle k_{\text{in}} \rangle < r$, and the simulation run converges to the uniformly resolved state as predicted by theory. In (b), the sensitivity and the internal completion rates are $\beta = 0.064$ and $r = 0.75$, respectively. In equilibrium, the average fraction of unresolved tasks between $t = 65$ and $t = 100$ (a stationary regime) is $\alpha^* = 0.087$, which agrees reasonably well with the prediction $\alpha^* = 0.090$ given in Equation (12).

the deterministic solution quite well. Performing multiple independent simulation runs using the same parameters has shown that the variation in the equilibrium obtained across different simulation runs has been quite small.

5.3. Analytic Results for Correlated PD Networks

The extreme heterogeneity of the connectivity distributions of undirected scale-free networks significantly affects the dynamical processes that propagate through these networks. In particular, it was shown that the large fluctuations, $\langle k^2 \rangle$, of power-law connectivity distributions cannot be neglected as far as system dynamics is concerned, even for finite-size systems (Anderson and May 1992). For directed PD networks, we show below that the first-order joint moment of the joint in-degree and out-degree distributions (i.e., $\langle k_{\text{in}} k_{\text{out}} \rangle$) plays an important role in determining the PD dynamics.

To take into account the extreme heterogeneity (related to the degree distributions) of real-world directed PD networks (as shown in §4), we modify the mean-field analysis presented for random (Erdős-Rényi) networks by writing the rate equations governing the time evolution of $\alpha_k(t)$, where $\alpha_k(t)$ is the relative density of unresolved tasks with given in-degree connectivity k (recall that for Erdős-Rényi networks, $\alpha_k(t) \approx \alpha(t)$). A general analysis, provided in the appendix (see Supplement 4 provided in the e-companion), indicates that the characteristics of PD dynamics are related to the degree correlations among neighboring tasks. For directed PD networks, the degree correlations can be measured by four possible linear (Pearson) correlation coefficients between sets of degrees (in- and out-degrees) for all tasks i and j at either ends of a directed edge in the network. More specifically, we consider $\rho(k^i_{\text{in}}, k^j_{\text{in}})$, $\rho(k^i_{\text{in}}, k^j_{\text{out}})$, $\rho(k^i_{\text{out}}, k^j_{\text{in}})$, and $\rho(k^i_{\text{out}}, k^j_{\text{out}})$, where the index i indicates the source node of the directed edge, and j refers to the destination node.[8] These measures reflect the tendency of tasks of similar degrees to be connected to one another. In particular, we show in Supplement 4 that the PD dynamics is directly determined by the correlation coefficient $\rho(k^i_{\text{in}}, k^j_{\text{in}})$. However, we further show that if neighboring tasks are weakly correlated (or weakly coupled, in which case $\rho(k^i_{\text{in}}, k^j_{\text{in}}) \approx 0$) a simplified analysis can be obtained. To this end, we have tested the correlation coefficients $\rho(k^i_{\text{in}}, k^j_{\text{in}})$ for each of the PD networks studied. These have been found to be 0.0943 (vehicle), −0.0644 (software), 0.2452 (pharmaceutical), and −0.0750 (hospital), indicating that $\rho(k^i_{\text{in}}, k^j_{\text{in}})$ is indeed small. Consequently, we assume that the degree correlations among neighboring tasks can be safely neglected, and proceed with a simplified analysis.

The dynamical mean-field rate equations now become

$$\frac{d\alpha_k(t)}{dt} = (1 - \alpha_k(t)) \tanh(\beta k \theta(t)) - \alpha_k(t) r (1 - \tanh(\beta k \theta(t))) \quad \forall k, \tag{15}$$

where $\theta(t)$ is the probability that any given incoming link (arc) to a task originates from an unresolved task. At an equilibrium $d\alpha_k(t)/dt = 0$, $\theta(t) = \theta$, and thus we obtain a single equation to be solved for $\alpha_k(t) = \alpha_k$:

$$\alpha_k = \frac{e^{\beta k \theta} - e^{-\beta k \theta}}{e^{\beta k \theta} - e^{-\beta k \theta} + 2re^{-\beta k \theta}}. \tag{16}$$

To solve the equations in (16), we need to derive an expression for θ. First, we define the probability q_m that an incoming link to a task originates from another task with m outgoing links. Because it is more likely that a randomly chosen link originates from a node with high out-degree connectivity, the probability q_m is proportional to $mP_{\text{out}}(m)$ and the normalized distribution is given by

$$q_m = \frac{mP_{\text{out}}(m)}{\sum_s sP_{\text{out}}(s)} = \frac{mP_{\text{out}}(m)}{\langle k_{\text{out}} \rangle}. \tag{17}$$

[8] For undirected networks, these measures are reduced to the linear (Pearson) correlation $\rho(k_i, k_j)$ between the sets $\{k_i\}$ and $\{k_j\}$ of total degrees for all tasks i and j at either ends of an edge in the network.

We conclude that θ is given by

$$\theta = \sum_m \sum_k \frac{mP_{\text{out}}(m)P(k_{\text{in}} = k \mid k_{\text{out}} = m)\alpha_k}{\langle k_{\text{out}} \rangle}$$
$$= \sum_m \sum_k \frac{mP(m,k)}{\langle k_{\text{out}} \rangle} \left(\frac{e^{\beta k \theta} - e^{-\beta k \theta}}{e^{\beta k \theta} - e^{-\beta k \theta} + 2re^{-\beta k \theta}} \right)$$
$$= f(\theta), \tag{18}$$

where $P(k_{\text{in}}, k_{\text{out}})$ is the joint probability distribution of k_{in} and k_{out}. Consequently, Equation (18) yields a consistency equation for θ. After solving Equation (18) for θ, the average density of unresolved tasks in the system at equilibrium is evaluated by the relation $\alpha = \sum_k P_{\text{in}}(k)\alpha_k$.

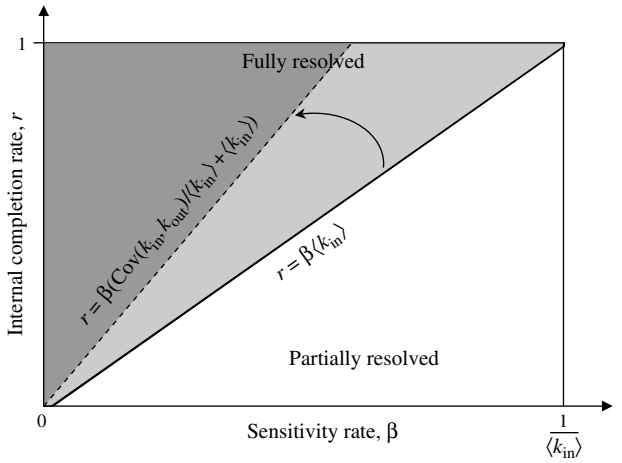
Finding an exact solution of Equation (18) can be difficult, depending on the particular form of $P(k_{\text{in}}, k_{\text{out}})$. Fortunately, we do not have to solve Equation (18) explicitly to gain a qualitative understanding of the underlying PD dynamics. Following the same argument as in Equation (10), we obtain that

$$f'(\theta)|_{\theta=0} = \sum_m \sum_k \frac{mP(m,k)}{\langle k_{\text{out}} \rangle} \left(\frac{\beta k}{r} \right) = \frac{\beta \langle k_{\text{in}} k_{\text{out}} \rangle}{r \langle k_{\text{out}} \rangle}$$
$$= \frac{\beta}{r} \left(\frac{\text{Cov}(k_{\text{in}}, k_{\text{out}})}{\hat{k}} + \hat{k} \right), \tag{19}$$

where $\langle k_{\text{in}} k_{\text{out}} \rangle$ denotes the first-order joint moment of the joint probability distribution $P(k_{\text{in}}, k_{\text{out}})$, $\text{Cov}(k_{\text{in}}, k_{\text{out}})$ denotes the covariance[9] of the two random variables k_{in} and k_{out}, and $\hat{k} = \langle k_{\text{in}} \rangle = \langle k_{\text{out}} \rangle$. Strikingly, the factors $\text{Cov}(k_{\text{in}}, k_{\text{out}})$ or $\langle k_{\text{in}} k_{\text{out}} \rangle$ in Equation (19) depend solely on the topology of the network; thus, showing how the underlying network topology provides direct information about the characteristics of the network dynamics. For the general case considered here, the model exhibits a threshold behavior at $f'(\theta)|_{\theta=0} = 1$, which implies that an initial seed of unresolved tasks would lead, at equilibrium, to the uniformly resolved state if $f'(\theta)|_{\theta=0} < 1$, i.e., for $\beta \langle k_{\text{in}} k_{\text{out}} \rangle < r \langle k_{\text{out}} \rangle$. We also note that if k_{in} and k_{out} are uncorrelated, then $\langle k_{\text{in}} k_{\text{out}} \rangle = \langle k_{\text{in}} \rangle \langle k_{\text{out}} \rangle$, from which we recover the condition $\beta \langle k_{\text{in}} \rangle / r < 1$ obtained for homogeneous random networks (see §5.2).

The above analysis suggests that the dynamic model does exhibit a threshold behavior, as for homogeneous networks. However, for positive covariance between the two random variables k_{in} and k_{out}, the range of $r - \beta$ values for which the system converges to the uniformly resolved state is more constrained compared with the corresponding homogeneous situation (see Figure 6). It is concluded that high covariance values exhibited in PD networks must be compensated for by either reducing the sensitivity of tasks to their neighbors (reflected by the parameter β) or by increasing their internal problem-solving rates (reflected by the parameter r) if the project is expected to converge to the uniformly resolved state. As mentioned in §4.2, the observed low correlation between the in-degrees and out-degrees of nodes for some PD networks implies that $\text{Cov}(k_{\text{in}}, k_{\text{out}})/\hat{k} \ll 1$ in Equation (19), and thus the predicted threshold behavior at $\beta \langle k_{\text{in}} \rangle / r = 1$ (as for random homogeneous networks) could still be a good approximation (this is confirmed by simulations).

Figure 6 Dynamical Behavior of a PD Network with Uncorrelated and Correlated Topologies



Notes. For a fixed network topology (i.e., $\langle k_{\text{in}} \rangle$ and $\text{Cov}(k_{\text{in}}, k_{\text{out}})$ are known values), the dynamics are characterized in terms of the parameters r and β. The parameter space $r - \beta$ is divided into two distinct regimes: fully resolved (gray) and partially resolved (white). The solid line represents the transition between these two regimes for a completely random (uncorrelated) network, while the dashed line represents the transition between these two regimes for a positively correlated network (i.e., $\text{Cov}(k_{\text{in}}, k_{\text{out}}) \geq 0$).

The above deterministic analysis has been tested by simulating the model on the software network described in §3 with $\hat{k} = 3.163$. The software network indicates a high degree of interdependence or covariance between the two random variables k_{in} and k_{out} (i.e. $\text{Cov}(k_{\text{in}}, k_{\text{out}}) \approx 5.59$ and correlation ≈ 0.76). For example, for internal completion rates $r_i = 0.75 \ \forall i$, a threshold behavior is predicted at a value of $\beta \approx 0.086$, for which

$$\frac{\beta}{r} \left(\frac{\text{Cov}(k_{\text{in}}, k_{\text{out}})}{\hat{k}} + \hat{k} \right) = 1.$$

It is instructive to compare the threshold thus obtained with the prediction for an uncorrelated random network, $\beta \approx 0.237$. The actual measurement of the threshold has been found at $\beta \approx 0.103$, smaller by

[9] We have shown elsewhere that some directed networks used to model the topology of the World Wide Web and e-mail networks yield very large covariance values that go to infinity as the number of nodes in the network goes to infinity.

a factor of 2.3 than the value corresponding to uncorrelated random networks ($\beta \approx 0.237$), and in quite good agreement with the prediction for correlated networks ($\beta \approx 0.086$). The actual threshold appears to be a bit higher than the prediction for correlated networks because the mean-field approximation used to derive the deterministic theory neglected the density correlations among the different tasks (see the appendix in Supplement 4) and because the size of the network is small.

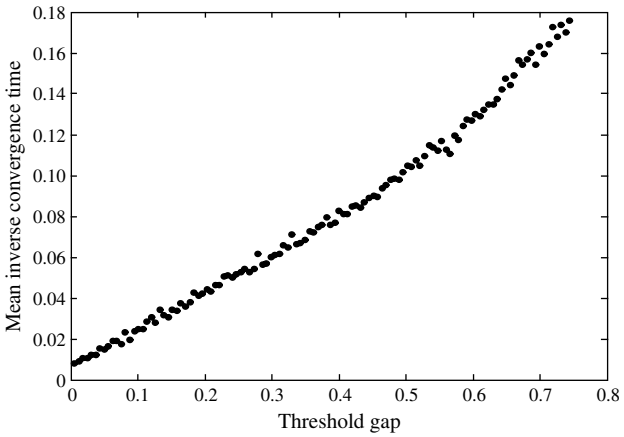
6. Simulation Results: Sensitivity and Robustness of PD

One of the most practical aspects of a product development process is whether the total number of design problems (e.g., unresolved tasks) being solved remains bounded as the project evolves over time, and eventually falls below an acceptable threshold within a specified time frame (Yassine and Braha 2003, Yassine et al. 2003). To analyze (from a global perspective) the performance of product development, we measure it by the time T^* it takes for the PD process to reach the uniformly resolved state.[10] Because the characterization of the PD systemwide behavior depends on the distribution of β_is and r_is, we consider the special case where the sensitivity and the internal completion rates are identical across tasks ($\beta_i = \beta$ and $r_i = r \ \forall i$).

We confirmed through simulations that the time it takes for the PD system to reach the uniformly resolved state increases sharply as the sensitivity and completion rate parameters, β and r, increase toward the threshold regime (e.g., the solid line in Figure 6). Indeed, we note that the exponent in the denominator of Equation (14) begins to dominate the other factors for values of t for which $(r - \bar{\beta})t \approx 1$. Thus, we expect the inverse of the mean convergence time, $1/T^*$, to grow linearly with the scaled parameter $\delta = r - \bar{\beta} = r - \beta\langle k_{\text{in}}\rangle$—the "threshold gap." This is illustrated in Figure 7 for the pharmaceutical network, where the inverse of the convergence time $1/T^*$ is plotted against the threshold gap δ, which verifies the predicted linear dependence of $1/T^*$ on δ over a large range of threshold gap values.

We further examine the dynamics of the PD process by analyzing the sensitivity as well as robustness (error tolerance) of the PD network topology with respect to internal and external perturbations such as planned and unplanned design changes. We demonstrate two important properties of complex PD networks: (1) their dynamic behavior is highly *insensitive* (error tolerant) to random perturbations, yet highly *sensitive* (responsive) to perturbations that are targeted at specific tasks; and (2) if wisely exploited, the sensitivity of PD complex networks to targeted perturbations can yield great benefits with minimal effort, yet the sensitivity characteristic may also result in detrimental effects if not properly controlled.

Figure 7 **The Inverse of the Mean Convergence Time, $1/T^*$, for Different Values of the Threshold Gap δ**



Notes. The average number of in coming arcs connected to a node is $\langle k_{\text{in}}\rangle = 6.37$, and the internal completion rates are $r_i = 0.75 \ \forall i$. The values of β go from 0 to 0.117 in increments of 0.001, and convergence times were averaged over 100 independent simulation runs. The plot shows a linear relationship (Pearson coefficient $R > 0.98$, $p < 0.001$).

In the following, perturbations are considered as either *planned* or *unplanned* task modifications that could affect the performance[11] of the PD process. Planned task modifications are defined as deliberate improvements of task parameters, and include:[12] (1) decreasing the value of sensitivity rates β_i, or (2) increasing the value of internal completion rates r_i. To simulate unplanned task changes, we modified tasks by impairing their sensitivity or completion-rate parameters.

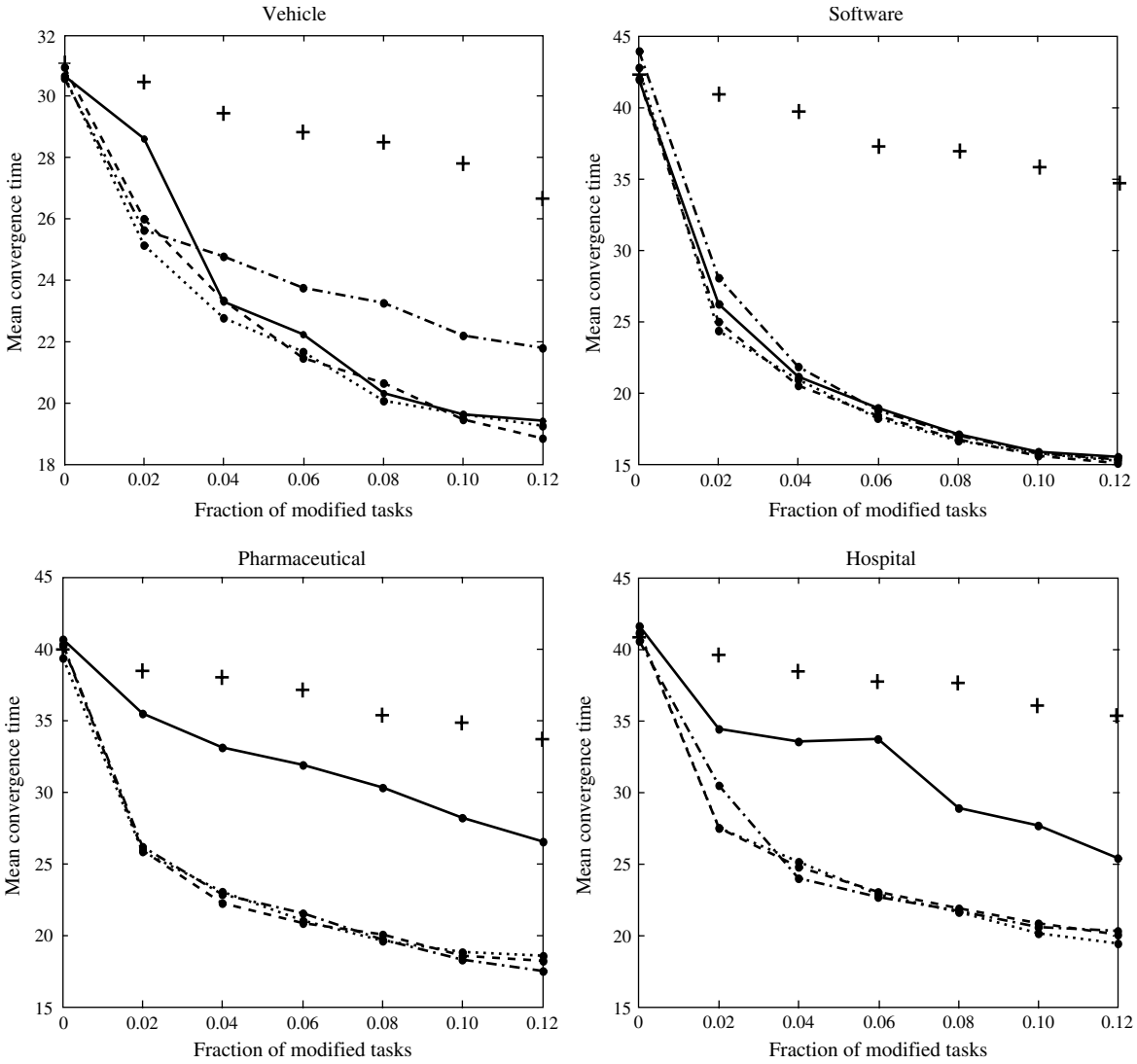
We begin by analyzing the effect of planned changes of tasks on the PD performance. Effective improvement of tasks will preferentially direct resources to the most "important" tasks (rather than selecting tasks randomly). Hence, we consider the following five task modification policies: (1) *Information-generating policy*: First, modify the task with the highest out-degree, and continue selecting and modifying tasks in decreasing order of their out-degree connectivity k_{out}. (2) *Information-consuming policy*: Same as in 1, but

[10] Here we guarantee convergence by proper selection of parameters. In general, we could use a performance measure T_p, where T_p is the earliest time at which the fraction of resolved tasks is greater or equal to p.

[11] As before, the performance is measured as the time it takes for the PD process to reach the uniformly resolved state.

[12] Restructuring (redesigning) the task connectivities is another means for improving performance. Here, we assume a fixed network topology.

Figure 8 Product Development Performance vs. Fraction of Modified Tasks for Which Internal Completion Rates Are Improved



Notes. Comparison between five task-modification policies: multiplicative (dotted line), additive (dashed line), information-generating (dashed-dotted line), information-consuming (solid line), and random (+). For the nonrandom modification policies, each data point is the average of 1,000 realizations. For the random modification policy, each point is the average of 30 different modified task selections, performed for 100 independent runs. The average in-degree, sensitivity rate, internal completion rate *prior to modification*, and modified internal completion rate are, respectively, as follows: Vehicle: $\langle k_{in}\rangle = 3.475$, $\beta = 0.135$, $r = 0.5$, $r^+ = 1$; Software: $\langle k_{in}\rangle = 3.163$, $\beta = 0.06$, $r = 0.5$, $r^+ = 1$; Pharmaceutical: $\langle k_{in}\rangle = 6.371$, $\beta = 0.065$, $r = 0.5$, $r^+ = 1$; Hospital: $\langle k_{in}\rangle = 9.741$, $\beta = 0.045$, $r = 0.5$, $r^+ = 1$.

modify tasks according to their in-degree k_{in}. (3) *Multiplicative policy*: Same as in 1, but modify tasks according to the product of their in-degree and out-degree $k_{in}k_{out}$. (4) *Additive policy*: Same as in 1, but modify tasks according to the sum of their in-degree and out-degree $k_{in} + k_{out}$. (5) *Random policy*: Tasks are selected randomly, and modified accordingly. The latter scheme reflects an uninformed modification strategy.

We examine the sensitivity and robustness of PD networks with respect to perturbations by studying how the performance is being affected when a small fraction, f, of the tasks is modified according to the task modification policies specified above. In general, as seen in Figure 8 and Figure EC.1 (Supplement 1 provided in the e-companion), planned task modifications tend to increase the performance of the PD process. However, while the PD performance increases slowly with f when the random modification scheme is applied, a drastically different behavior is observed when the deliberate modification schemes are utilized. When tasks are modified preferentially (by either one of the above modification policies), the performance of the PD network increases rapidly, becoming about twice as large as its original value, even if only 6% of the tasks are modified. This sensitivity to deliberate perturbations is deeply ingrained in the inhomogeneity property of the in-degree and

out-degree connectivity distributions of PD networks as indicated by their long right tails and extremely large variances (see §4). More specifically, the inhomogeneity property related to the out-degree connectivity means that the PD network is dominated by a few tasks that generate information to a large number of other neighboring tasks. Similarly, the inhomogeneity associated with the in-degree distribution implies that the PD network is dominated by a few tasks that consume information from a large number of neighboring tasks. Consequently, improvement efforts that are channeled towards these dominating tasks (e.g., increasing their internal completion rates) are expected to *drastically* alter the overall network's performance.

The aggregate-based policies (multiplicative and additive) seem to generally outperform the single-based policies (information-generating and information-consuming). This result is rooted in the nature of the directed information flows forming the links among the tasks. While not uniquely affected by either the in-degree or out-degree connectivity distributions alone, both distributions are needed to understand the dynamics of the PD process. Tasks with large in- and out-degrees have both significant internal complexity associated with assembling the information of several other tasks, and significant external dependability upon which others rely. Thus, it is plausible to expect that tasks with large in- and out-degrees could hamper the PD process.

Figures 8 and EC.1 show that for the vehicle, pharmaceutical, and hospital networks the performance of the information-consuming policy (based on in-degree connectivities) is poor relative to the other policies. As observed in §4, these networks have the following properties: (1) the correlation between the in-degree and out-degree of tasks is small, and (2) the in-degree distribution has a cutoff that is significantly lower than the corresponding out-degree cutoff. This suggests that other networks that satisfy these properties and utilize the information-consuming policy might also perform less effectively. Indeed, an early cutoff of the in-degree distribution (relative to the out-degree cutoff) implies that tasks with large incoming connectivities are practically absent. Also, a lack of degree correlation implies that it is unlikely that a highly information-generating task (i.e., with large out-degree connectivity) is also highly information consuming (i.e., with high in-degree connectivity). Consequently, the PD dynamics are generally expected to be more responsive to modifications that focus on tasks with high out-degree connectivities.

Finally, we observe that, for the software network, all the "nonrandom" policies perform similarly. This might be expected for networks for which the in-degree and out-degree connectivities are highly correlated (e.g., for the software network, the correlation is 0.76).

Next, we analyze the effect of *unplanned* changes of tasks (where task parameters are impaired) on the PD performance. As seen in Figures EC.2 and EC.3 (Supplements 2 and 3, provided in the e-companion), the PD performance decreases slowly with f when tasks are changed randomly. On the other hand, a drastically different behavior is observed if unplanned changes are targeted at central tasks. When tasks are modified preferentially (by either one of the above modification policies), the performance of the PD network decreases rapidly, becoming about twice as low as its original value even if only 6% of the tasks are modified.

Overall, Figures 8 and EC.1–EC.4 illustrate the double-faceted characteristic of PD sensitivity—if wisely planned, the sensitivity of PD complex networks to targeted perturbations can yield great benefits with minimal effort (Figures 8 and EC.1), yet the sensitivity characteristic may also result in detrimental effects if not properly controlled (Figures EC.2 and EC.3).

7. Summary and Conclusions

In the last few years, the study of complex network topologies has become a rapidly advancing area of research across many fields of science and technology (Strogatz 2001, Albert and Barabási 2002, Newman 2003). One of the key areas of research is understanding the network properties that are optimized by specific network architectures (Amaral et al. 2000, Valverde et al. 2002, Cancho and Solé 2003, Mossa et al. 2002, Shargel et al. 2003). Here, we have analyzed the statistical properties of real-world networks of people engaged in product development activities. We have shown that complex PD networks display similar statistical patterns to other real-world networks of different origins, and have shown how the underlying network topologies provide direct information about the characteristics of PD dynamics. In particular:

- PD complex networks exhibit the small-world property, which means that they react rapidly to changes in design status;
- PD complex networks are characterized by inhomogeneous distributions of incoming and outgoing information flows of tasks. Consequently, PD task networks are dominated by a few highly central information-consuming and information-generating tasks;
- PD networks exhibit a noticeable asymmetry (related to the cutoffs) between the distributions of incoming and outgoing information flows, suggesting that the incoming capacities of tasks are much more limited than their counterpart outgoing capacities. The cutoffs observed in the in-degree and out-degree

distributions might reflect Herbert Simon's notion of bounded rationality (Simon 1998) and its extension to group-level information processing.

- Focusing engineering and management efforts on central information-consuming and information-generating PD tasks will likely improve the performance of the overall PD process;
- Failure of central PD tasks affects the vulnerability of the overall PD process;
- Positive correlation between the in-degree and out-degree of a task tends to limit the range of the parameters' values for which the system converges to the uniformly resolved state;
- PD dynamics is highly error tolerant, yet highly responsive to perturbations that are targeted at specific tasks.

In the context of product development, what is the meaning of these patterns? How do they come to be what they are? We propose several explanations for these patterns. Successful PD processes in competitive environments are often characterized by short time to market, high product performance, and low development costs (Clark 1989). In many high-technology industries, an important trade-off exists between minimizing time-to-market and development costs and maximizing the product performance. In PD task networks, accelerating the PD process can be achieved by cutting out some of the links between the tasks (Yassine et al. 2003). Although the elimination of some arcs should result in more rapid PD convergence, this might worsen the performance of the end system. Consequently, a trade-off exists between the elimination of task dependencies (speeding up the process) and the desire to improve the system's performance through the incorporation of additional task dependencies. PD networks are likely to be highly optimized when both PD completion time and product performance are accounted for. Recent studies have shown that an evolutionary algorithm involving minimization of link density and average distance between any pair of nodes can lead to nontrivial types of networks, including truncated scale-free networks, i.e., $p(k) = k^{-\gamma} f(k/k^*)$ (Valverde et al. 2002, Cancho and Solé 2003). This might suggest that an evolutionary process that incorporates similar generic optimization mechanisms (e.g., minimizing a weighted sum of development time and product quality losses) might lead to the formation of a PD network structure with the small-world and truncated scale-free properties.

Another explanation for the characteristic patterns of PD networks might be related to the close interplay between the design structure (product architecture) and the related organization of tasks involved in the design process. It has been observed that in many technical systems design tasks are commonly organized around the architecture of the product (Eppinger et al. 1994). Consequently, there is a strong association between the information flows underlying the PD task network and the design network composed of the physical (or logical) components of the product and the interfaces between them. If the task network is a mirror image of the related design network, it is reasonable that their large-scale statistical properties might be similar. Evidence for this can be found in recent empirical studies that show some design networks (electronic circuits, Ferrer et al. 2001, and software architectures, Valverde et al. 2002) exhibit small-world and scaling properties. The scale-free structure of design networks, in turn, might reflect the strategy adopted by many firms of reusing existing modules together with newly developed modules in future product architectures (Braha and Maimon 1998). Thus, the highly connected nodes of the scale-free design network tend to be the most reusable modules. Reusing modules at the product architecture level also has a direct effect on the task level of product development; it allows firms to reduce the complexity and scope of the product development project by exploiting the knowledge embedded in reused modules, and thus significantly reduce the product development time.

Of greatest significance for the analysis of generic network architectures, we demonstrated a previously unreported difference between the distribution of incoming and outgoing links in a complex network (see Figure EC.4, which is provided as Supplement 5 in the e-companion, for an analysis of open source software and electronic circuit networks). Specifically, we find that the distribution of incoming communication links always has a cutoff, while outgoing communication links are scale-free with or without a cutoff. In the cases studied, when both distributions have cutoffs, the incoming distribution has a cutoff that is significantly lower by more than a factor of two. From a product development viewpoint, the functional significance of this asymmetric topology has been explained by considering a bounded-rationality argument originally put forward by Simon in the context of human interactions (Simon 1998). Accordingly, this asymmetry could be interpreted as indicating a limitation on the actor's capacity to process information provided by others, rather than the ability to transmit information over the network. In the latter case, boundedness is less apparent because the capacity required to transmit information over a network is often less constrained, especially when it is replicated (e.g., many actors can receive the same information from a single actor by broadcast). In light of this observation, we expect a distinct cutoff distribution for in-degree as opposed to out-degree distributions

when the network reflects communication of information between human beings as a natural and direct outcome of Simon's bounded-rationality argument. It would be interesting to see whether this property can be found more generally in other directed human or nonhuman networks. It seems reasonable to propose that the asymmetric link distribution is likely to hold for such networks when nodes represent information processing elements.

The paper analyzes an intraorganizational network where PD tasks are nodes. It would be interesting to see if the statistical patterns uncovered for intraorganizational networks remain invariant when moving to the interorganizational level, where enterprises form the nodes (e.g., supply chain networks; see Reitman 1997, Nishiguchi and Beaudet 1998). We conjecture that the level of abstraction will not significantly change the qualitative structure of the network's topology; but may change the embedded parameters underlying the network's characteristics (e.g., coefficients and cutoffs of the power-law distributions). We have identified two generic categories of network nodes: information-consuming and information-generating. We believe that this categorization could be expanded by at least three methods: (1) considering other unit centrality measures (e.g., closeness and betweenness centrality; see Wasserman and Faust 1999); (2) analyzing the structure of subgraphs (building blocks) embedded in the networks; and (3) assigning richer data structures that more naturally describe a PD; e.g., adding characteristics to each task or adding information bandwidth (weights) to links. Finally, it would be interesting to see (by direct observations) if the group-level information-processing capacity reflected by the distributions' cutoffs can be extended; e.g., by redesigning the structure or topology of the network or by incorporating sophisticated information technologies and transaction protocols.

8. Electronic Companion

An electronic companion to this paper is available as part of the online version that can be found at http://mansci.journal.informs.org/.

Acknowledgments

This paper is a major extension of Braha and Bar-Yam (2004). The authors thank Alberto Cividanes of the Massachusetts Institute of Technology for providing the vehicle data at General Motors' Research and Development Center, Stephen Denker of the Business Process Architects for providing the operating software data at a major telecommunication company, and Simon Austin and Andy Newton of Loughborough University and ADePT Management Limited for providing the pharmaceutical facility and 16-story hospital facility data. The authors also thank the referees for their useful comments.

References

Albert, R., A.-L. Barabási. 2002. Statistical mechanics of complex networks. *Rev. Modern Phys.* **74** 47–97.

Albert, R., H. Jeong, A.-L. Barabási. 1999. Diameter of the World Wide Web. *Nature* **401** 130–131.

Albert, R., H. Jeong, A.-L. Barabási. 2000. Error and attack tolerance in complex networks. *Nature* **406** 378–382.

Alexander, C. 1964. *Notes on the Synthesis of Form*. Harvard University Press, Cambridge, MA.

Amaral, L. A. N., A. Scala, M. Barthélémy, H. E. Stanley. 2000. Classes of small-world networks. *Proc. Natl. Acad. Sci. USA* **97** 11149–11152.

Anderson, R. M., R. M. May. 1992. *Infectious Diseases of Humans: Dynamics and Control*. Oxford University Press, New York.

Austin, S., A. Baldwin, B. Li, P. Waskett. 1999. Analytical design planning technique: A model of the detailed building design process. *Design Stud.* **20**(3) 279–296.

Austin, S., A. Baldwin, B. Li, P. Waskett. 2000. Integrating design in the project process. *Proc. Institution Civil Engineers* **138**(4) 177–182.

Barabási, A.-L., R. Albert. 1999. Emergence of scaling in random networks. *Science* **286** 509–512.

Barrat, A., M. Barthélémy, R. Pastor-Satorras, A. Vespignani. 2004. The architecture of complex weighted networks. *Proc. Natl. Acad. Sci. USA* **101** 3747–3752.

Bar-Yam, Y. 1997. *Dynamics of Complex Systems*. Perseus Books, Reading, MA.

Bar-Yam, Y., I. R. Epstein. 2004. Response of complex networks to stimuli. *Proc. Natl. Acad. Sci. USA* **101** 4341–4345.

Braha, D., Y. Bar-Yam. 2004. Topology of large-scale engineering problem-solving networks. *Physical Rev. E.* **69**(1) 016113-1–016113-7.

Braha, D., O. Maimon. 1998. *A Mathematical Theory of Design: Foundations, Algorithms, and Applications*. Kluwer Academic Publishers, Boston, MA.

Cancho, R. F., R. V. Solé. 2003. Optimization in complex networks. *Lecture Notes in Physics*, Vol. 625. Springer, Berlin, Germany, 114–125.

Cividanes, A. 2002a. Optimal scheduling of design reviews in product development. M.Sc. thesis, Mechanical Engineering Department, Massachusetts Institute of Technology, Cambridge, MA.

Cividanes, A. 2002b. Private communication. A complete description of the tasks, the list of interviewees, and the result of the survey at GM's Research and Development Center are available at http://necsi.org/projects/braha/largescaleengineering.html.

Clark, K. B. 1989. Project scope and project performance: The effect of parts strategy and supplier involvement on product development. *Management Sci.* **35**(10) 1247–1263.

Denker, S. 2002. Private communication. Available at http://necsi.org/projects/braha/largescaleengineering.html.

de S. Price, D. J. 1965. Networks of scientific papers. *Science* **149** 510–515.

de Sola Pool, I., M. Kochen. 1978. Contacts and influence. *Soc. Networks* **1**(1) 5–51.

Eppinger, S. D., D. E. Whitney, R. P. Smith, D. A. Gebala. 1994. A model-based method for organizing tasks in product development. *Res. Engrg. Design* **6**(1) 1–13.

Erdős, P., A. Rényi. 1959. On random graphs. *Publicationes Mathematicae* **6**(1) 290–297.

Faloutsos, M., P. Faloutsos, C. Faloutsos. 1999. On power-law relationships of the Internet topology. *Comp. Comm. Rev.* **29** 251–262.

Ferrer, R., C. Janssen, R. V. Solé. 2001. Topology of technology graphs: Small world patterns in electronic circuits. *Phys. Rev. E* **64**(4) 046119-1–046119-5.

Granovetter, M. 1973. The strength of weak ties. *Amer. J. Sociol.* **78** 1360–1380.

Jeong, H., S. Mason, A.-L. Barabási, Z. N. Oltvai. 2001. Lethality and centrality in protein networks. *Nature* **411** 41–42.

Jeong, H., B. Tombor, R. Albert, Z. N. Oltavi, A.-L. Barabási. 2000. The large-scale organization of metabolic networks. *Nature* **407** 651–654.

Klein, M., H. Sayama, P. Faratin, Y. Bar-Yam. 2006. The dynamics of collaborative design: Insights from complex systems and negotiation research. D. Braha, A. Minai, Y. Bar-Yam, eds. *Complex Engineered Systems: Science Meets Technology*. Springer, New York, 158–174.

Krackhardt, D., J. R. Hanson. 1993. Informal networks: The company behind the chart. *Harvard Bus. Rev.* **71**(4) 104–111.

Marro, J., R. Dickman. 1999. *Nonequilibrium Phase Transitions in Lattice Models*. Cambridge University Press, Cambridge, UK.

Mihm, J., C. H. Loch, A. Huchzermeier. 2003. Problem-solving oscillations in complex projects. *Management Sci.* **49**(6) 733–750.

Montoya, J. M., R. V. Solé. 2002. Small world patterns in food webs. *J. Theoret. Biol.* **214** 405–412.

Mossa, S., M. Barthélémy, H. E. Stanley, L. A. N. Amaral. 2002. Truncation of power law behavior in "scale-free" network models due to information filtering. *Phys. Rev. Lett.* **88**(13) 138701-1–138701-4.

Newman, M. E. J. 2001. The structure of scientific collaboration networks. *Proc. Natl. Acad. Sci. USA* **98** 404–409.

Newman, M. E. J. 2003. The structure and function of complex networks. *SIAM Rev.* **45** 167–256.

Newton, A., S. Austin. 2002. Private communication. Available at http://necsi.org/projects/braha/largescaleengineering.html.

Nishiguchi, T., A. Beaudet. 1998. The Toyota group and the Aisin fire. *Sloan Management Rev.* (Fall).

Osborne, S. M. 1993. Product development cycle time characterization through modeling of process iteration. M.Sc. thesis, Massachusetts Institute of Technology, Cambridge, MA.

Reitman, V. 1997. Toyota's fast rebound. *Wall Street J.* (May 8) A1.

Shargel, B., H. Sayama, I. R. Epstein, Y. Bar-Yam. 2003. Optimization of robustness and connectivity in complex networks. *Phys. Rev. Lett.* **90**(6) 068701-1–068701-7.

Simon, H. A. 1998. *The Sciences of the Artificial*. MIT Press, Cambridge, MA.

Solomonoff, R., A. Rapoport. 1951. Connectivity of random nets. *Bull. Math. Biophysics* **13** 107–117.

Steward, D. V. 1981. The design structure system: A method for managing the design of complex systems. *IEEE Trans. Engrg. Management* **28** 71–74.

Strogatz, S. H. 2001. Exploring complex networks. *Nature* **410** 268–276.

Valverde, S., R. F. Cancho, R. V. Solé. 2002. Scale free networks from optimal design. *Europhysics Lett.* **60** 512–517.

Wasserman, S., K. Faust. 1999. *Social Network Analysis*. Cambridge University Press, Cambridge, UK.

Watts, D. J., S. H. Strogatz. 1998. Collective dynamics of "small-world" networks. *Nature* **393** 440–442.

Yassine, A., D. Braha. 2003. Complex concurrent engineering and the design structure matrix method. *Concurrent Engrg.* **11**(3) 165–176.

Yassine, A., N. Joglekar, D. Braha, S. Eppinger, D. Whitney. 2003. Information hiding in product development: The design churn effect. *Res. Engrg. Design* **14**(3) 131–144.

MANAGEMENT SCIENCE
DOI 10.1287/mnsc.1060.0617ec
pp. ec1–ec5

e-companion

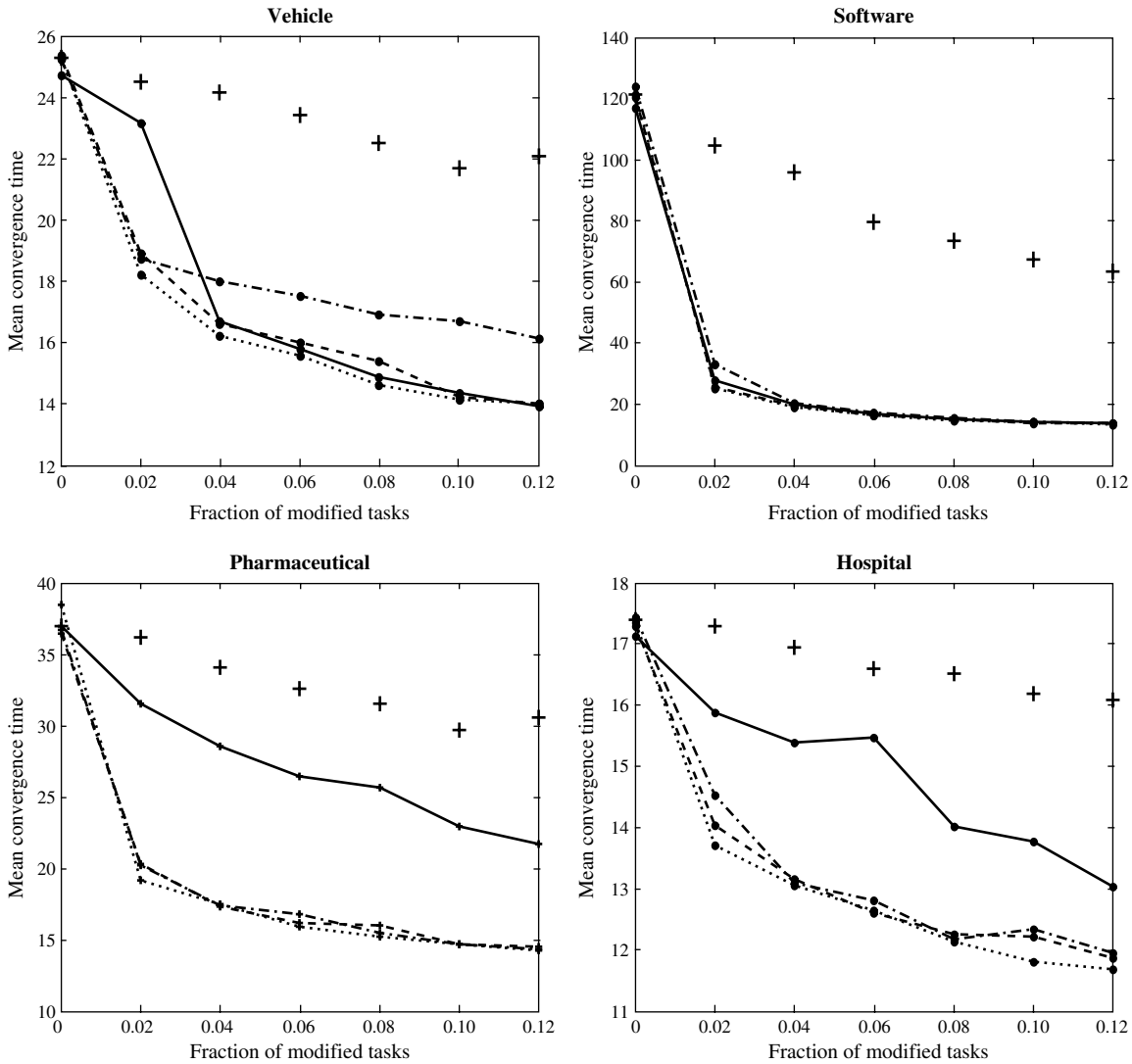
ONLY AVAILABLE IN ELECTRONIC FORM

Electronic Companion—"The Statistical Mechanics of Complex Product Development: Empirical and Analytical Results" by Dan Braha and Yaneer Bar-Yam, *Management Science* 2007, 53(7) 1127–1145.

Online Supplements

Supplement 1. Figure EC.1

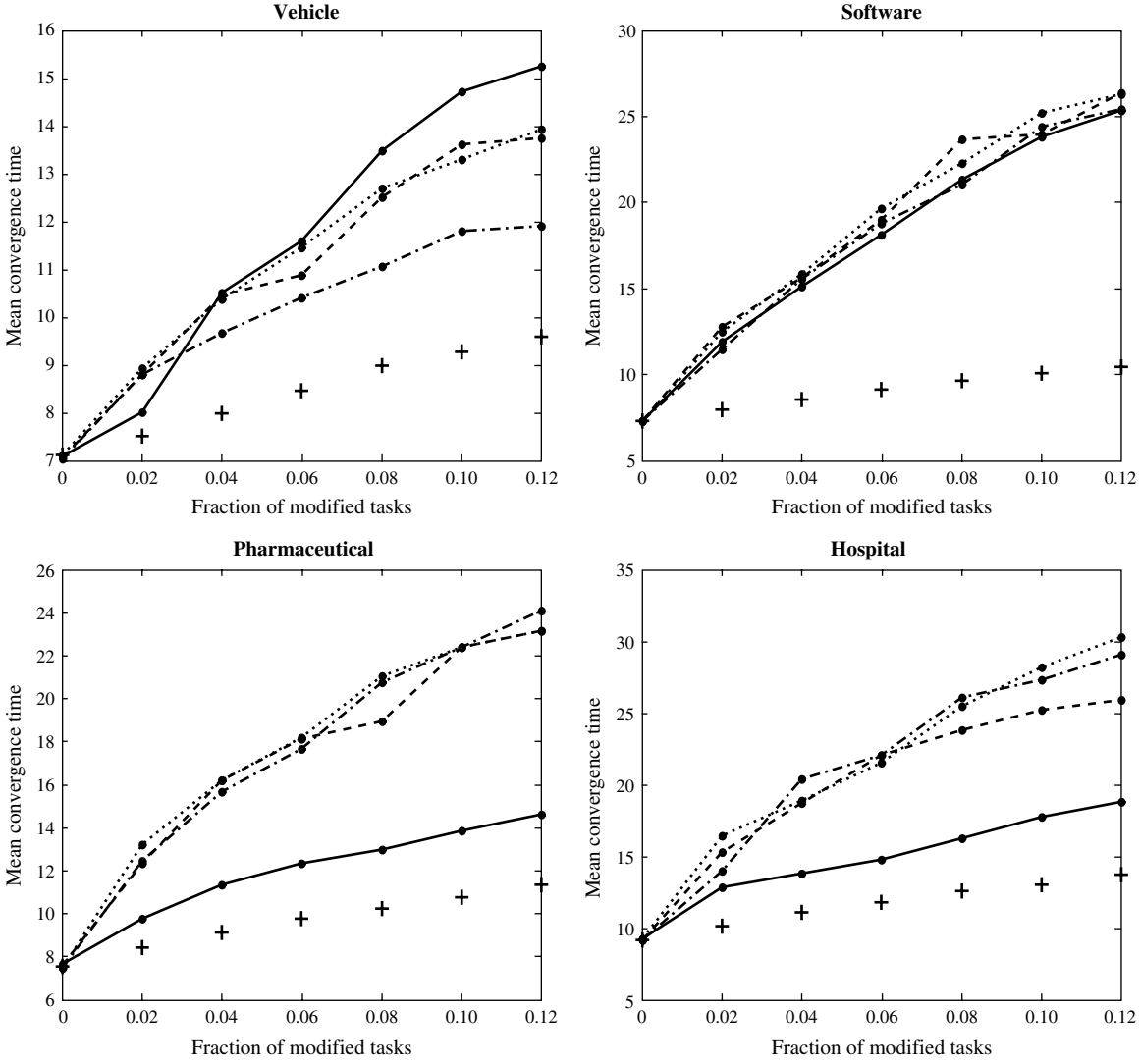
Figure EC.1 Product Development Performance vs. Fraction of Modified Tasks for Which Sensitivity Rates Are Improved



Notes. Comparison between five-task modification policies: Multiplicative (dotted line), additive (dashed line), information-generating (dashed-dotted line), information-consuming (solid line), and random (+). The average in-degree, internal completion rate, sensitivity rate *prior to modification*, and modified sensitivity rate are, respectively, as follows: Vehicle: $\langle k_{\text{in}} \rangle = 3.475$, $r = 0.75$, $\beta = 0.2$, $\beta^- = 0.1$; Software: $\langle k_{\text{in}} \rangle = 3.163$, $r = 0.75$, $\beta = 0.1$, $\beta^- = 0.05$; Pharmaceutical: $\langle k_{\text{in}} \rangle = 6.371$, $r = 0.75$, $\beta = 0.1$, $\beta^- = 0.05$; Hospital: $\langle k_{\text{in}} \rangle = 9.741$, $r = 0.75$, $\beta = 0.05$, $\beta^- = 0.025$.

Supplement 2. Figure EC.2

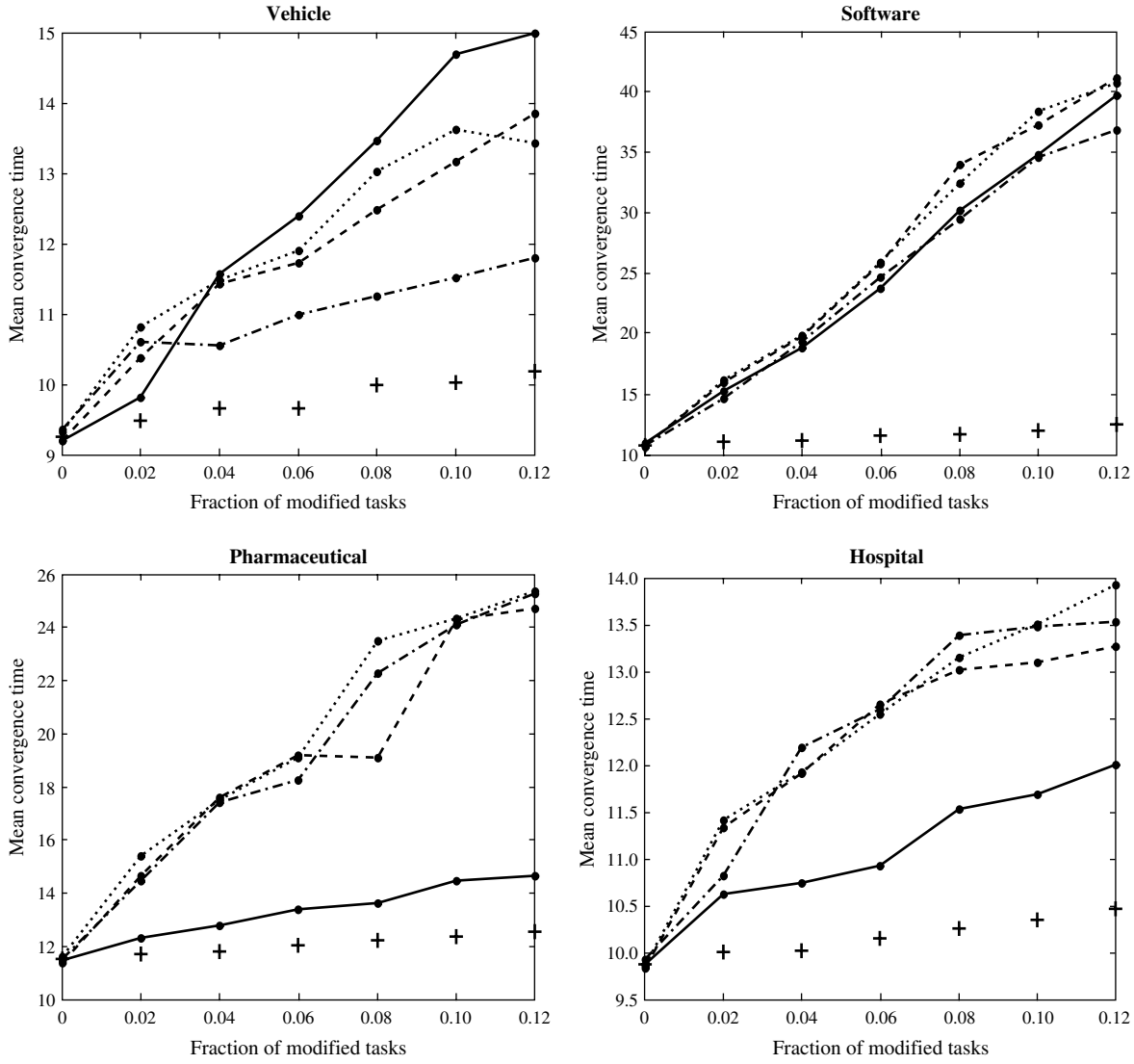
Figure EC.2 Product Development Performance vs. Fraction of Modified Tasks for Which Completion Rates Are Impaired



Notes. Comparison between five-task modification policies: Multiplicative (dotted line), additive (dashed line), information-generating (dashed-dotted line), information-consuming (solid line), and random (+). The average in-degree, sensitivity rate, internal completion rate *prior to modification*, and modified internal completion rate are, respectively, as follows: Vehicle: $\langle k_{in} \rangle = 3.475$, $\beta = 0.135$, $r = 1$, $r^- = 0.5$; Software: $\langle k_{in} \rangle = 3.163$, $\beta = 0.06$, $r = 1$, $r^- = 0.5$; Pharmaceutical: $\langle k_{in} \rangle = 6.371$, $\beta = 0.06$, $r = 1$, $r^- = 0.5$; Hospital: $\langle k_{in} \rangle = 9.741$, $\beta = 0.05$, $r = 1$, $r^- = 0.5$.

Supplement 3. Figure EC.3

Figure EC.3 Product Development Performance vs. Fraction of Modified Tasks for Which Sensitivity Rates Are Impaired



Notes. Comparison between five-task modification policies: Multiplicative (dotted line), additive (dashed line), information-generating (dashed-dotted line), information-consuming (solid line), and random (+). The average in-degree, internal completion rate, sensitivity rate *prior to modification*, and modified sensitivity rate are, respectively, as follows: Vehicle: $\langle k_{\text{in}} \rangle = 3.475$, $r = 0.75$, $\beta = 0.1$, $\beta^+ = 0.2$; Software: $\langle k_{\text{in}} \rangle = 3.163$, $r = 0.75$, $\beta = 0.05$, $\beta^+ = 0.1$; Pharmaceutical: $\langle k_{\text{in}} \rangle = 6.371$, $r = 0.75$, $\beta = 0.05$, $\beta^+ = 0.1$; Hospital: $\langle k_{\text{in}} \rangle = 9.741$, $r = 0.75$, $\beta = 0.025$, $\beta^+ = 0.05$.

Supplement 4. Appendix

The analysis presented in §5.3 neglected the correlations *among* different tasks' connectivities. For completeness, we present in this appendix a brief analysis of the case where there are explicit correlations between *pairs* of neighboring tasks. First, we modify the dynamical mean-field rate equations as follows:

$$\frac{d\alpha_k(t)}{dt} = (1-\alpha_k(t))\tanh(\beta k\theta_k(t)) - \alpha_k(t)r(1-\tanh(\beta k\theta_k(t))) \quad \forall k, \tag{EC1}$$

where $\theta_k(t)$ is the probability that any given incoming link (arc) to a task with in-degree k originates from an unresolved task.

It is easy to see that θ_k is given by

$$\theta_k(t) = \sum_j P(\tilde{k}_{\text{in}} = j \mid k_{\text{in}} = k)\alpha_j(t), \tag{EC2}$$

where $P(\tilde{k}_{\text{in}} = j \mid k_{\text{in}} = k)$ is the probability that an incoming link to a task with in-degree $k_{\text{in}} = k$ originates from another (neighboring) task with in-degree $\tilde{k}_{\text{in}} = j$. By plugging the expressions for $\theta_k(t)$ in (EC1), we obtain a first-order nonlinear system of differential equations,

$$\frac{d\boldsymbol{\alpha}}{dt} = \mathbf{f}(\boldsymbol{\alpha}). \tag{EC3}$$

We will analyze the stability of the uniformly resolved state $\alpha_k = 0 \ \forall k$ by using a linearization technique. More specifically, the uniformly resolved state $\boldsymbol{\alpha} = 0$ will be unstable if the Jacobian matrix of $\mathbf{f}(\boldsymbol{\alpha})$ at the fixed point $\boldsymbol{\alpha} = 0$ has positive eigenvalues. The Jacobian matrix of $\mathbf{f}(\boldsymbol{\alpha})$ at the fixed point $\alpha = 0$ can be shown to be

$$J(\boldsymbol{\alpha})_{\boldsymbol{\alpha}=0} = \begin{pmatrix} \frac{\partial f_1}{\partial \alpha_1} & \frac{\partial f_1}{\partial \alpha_2} & \cdots & \frac{\partial f_1}{\partial \alpha_{n-1}} & \frac{\partial f_1}{\partial \alpha_n} \\ \frac{\partial f_2}{\partial \alpha_1} & \frac{\partial f_2}{\partial \alpha_2} & \ddots & \frac{\partial f_2}{\partial \alpha_{n-1}} & \frac{\partial f_2}{\partial \alpha_n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \frac{\partial f_{n-1}}{\partial \alpha_1} & \frac{\partial f_{n-1}}{\partial \alpha_2} & \ddots & \frac{\partial f_{n-1}}{\partial \alpha_{n-1}} & \frac{\partial f_{n-1}}{\partial \alpha_n} \\ \frac{\partial f_n}{\partial \alpha_1} & \frac{\partial f_n}{\partial \alpha_2} & \cdots & \frac{\partial f_n}{\partial \alpha_{n-1}} & \frac{\partial f_n}{\partial \alpha_n} \end{pmatrix}_{\boldsymbol{\alpha}=0} = \beta A - rI,$$

where $A = (a_{kj})$ is a matrix for which $a_{kj} = kP(\tilde{k}_{\text{in}} = j \mid k_{\text{in}} = k)$, and I is the identity matrix. Let λ^* be the largest eigenvalue of the matrix A. Thus, the largest eigenvalue of the Jacobian $J(\boldsymbol{\alpha})_{\boldsymbol{\alpha}=0}$ is $\beta\lambda^* - r$, and we conclude that the uniformly resolved state $\boldsymbol{\alpha} = 0$ is unstable if $\beta > (r/\lambda^*)$.

Finally, it is shown that if the correlations among different tasks' connectivities are neglected we recover the conditions derived in §5.3. If neighboring tasks are uncorrelated—that is, $P(\tilde{k}_{\text{in}} = j \mid k_{\text{in}} = k) \approx P(\tilde{k}_{\text{in}} = j)$—we obtain

$$a_{kj} = kP(\tilde{k}_{\text{in}} = j \mid k_{\text{in}} = k) = \sum_m kP(\tilde{k}_{\text{in}} = j \mid \tilde{k}_{\text{out}} = m, k_{\text{in}} = k)P(\tilde{k}_{\text{out}} = m \mid k_{\text{in}} = k)$$

$$\approx \sum_m kP(\tilde{k}_{\text{in}} = j \mid \tilde{k}_{\text{out}} = m)\frac{mP(\tilde{k}_{\text{out}} = m)}{\langle \tilde{k}_{\text{out}} \rangle} = \sum_m mP(\tilde{k}_{\text{in}} = j, \tilde{k}_{\text{out}} = m)\frac{k}{\langle \tilde{k}_{\text{out}} \rangle}.$$

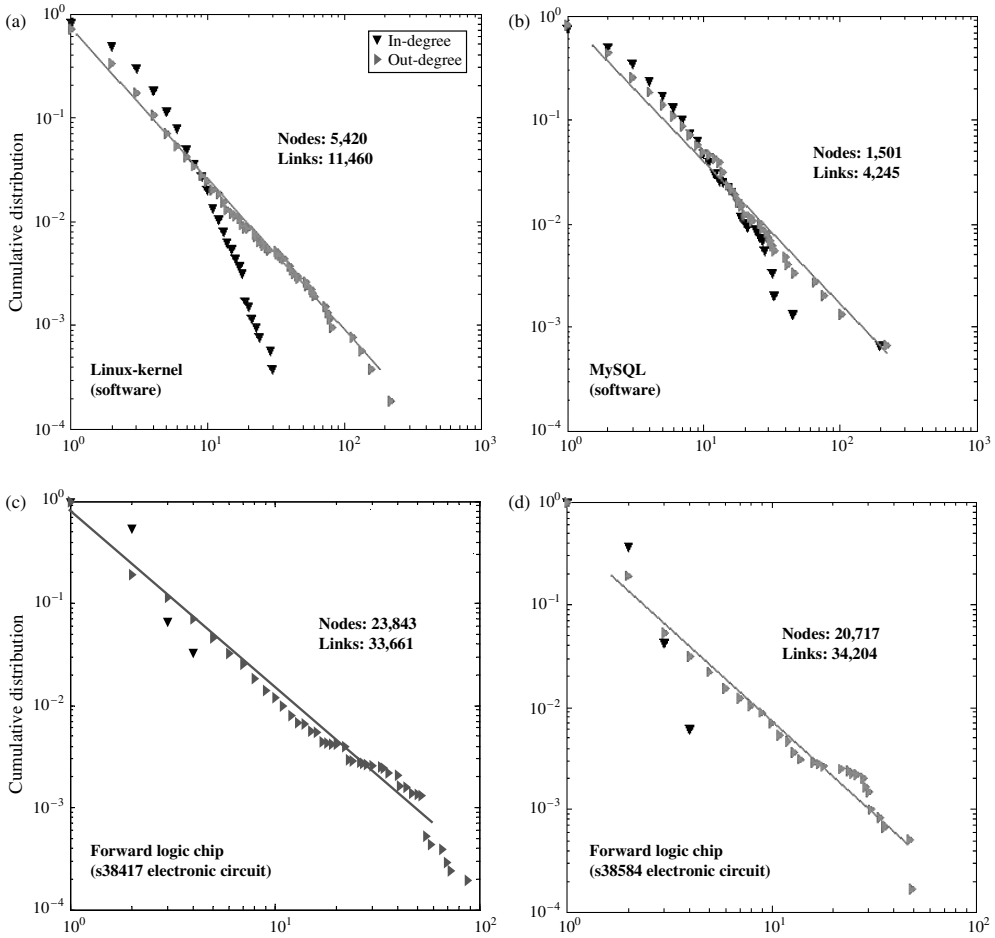
A direct calculation shows that the matrix A thus obtained has the eigenvalue $\lambda^* = \langle k_{\text{in}}k_{\text{out}} \rangle / \langle k_{\text{out}} \rangle$ with the corresponding eigenvector $\mathbf{v} = (1, 2, \ldots, n-1, n)$. Moreover, it is easy to check that λ^* is the unique and thus largest eigenvalue of A. Consequently, we conclude that the uniformly resolved state $\boldsymbol{\alpha} = 0$ is unstable if

$$\beta > \frac{r}{\lambda^*} = \frac{r\langle k_{\text{out}} \rangle}{\langle k_{\text{in}}k_{\text{out}} \rangle}$$

recovering the result established in §5.3.

Supplement 5. Figure EC.4

Figure EC.4 Degree Distributions for Open Source Software and Electronic Circuit Networks



Notes. (a and b) Open source software systems: The software system networks were generated from the call graphs of the Linux operating system kernel, and the MySQL relational database system (version 2.4.19 and version 3.23.32 respectively, data courtesy of Chris Myers, Cornell University). A call graph is a directed graph that represents calling relationship among subroutines. (c and d) Electronic circuits: The electronic circuit networks were generated from the ISCAS89 benchmark set of sequential logic electronic circuits. The nodes represent logic gates and flip-flops (data courtesy of Ron Milo, Weizmann Institute). The log-log plots of the cumulative distributions of incoming and outgoing links show a power law regime for the out-degree distributions with a fast decaying tail for the in-degree distributions. Similar to the product development networks (see Figure 3 in main text), the product design networks exhibit a noticeable asymmetry (related to the cut-offs) between the distributions of incoming and outgoing information flows, suggesting that the incoming capacities of "components" are much more limited than their counterpart outgoing capacities. These product design networks also exhibit a low correlation between the in-degrees and out-degrees of nodes as observed for the PD networks.