MICHIGAN
ROSS SCHOOL OF BUSINESS

Working Paper

Efficient Structures for Innovative Social Networks

William S. Lovejoy
Stephen M. Ross School of Business
University of Michigan

Amitabh Sinha
Stephen M. Ross School of Business
University of Michigan

Ross School of Business Working Paper
Working Paper No. 1136
November 2009

Efficient Structures for Innovative Social Networks*

William S. Lovejoy† Amitabh Sinha‡

November 12, 2009

*This paper is an earlier, expanded version of a paper that is under review at Management Science. The version under review has three main differences with respect to this manuscript. They are: 1. Group meetings (Section 9) in this paper are implemented in a different way in the version under review. This changes some, but not all, of the results regarding group meetings. 2. One of the main findings of this paper is the efficiency of what we call the Wheel-Star family of graphs. Although these graphs were discovered independently by us, we recognize that they are identical to the core-periphery graphs of Borgatti and Everett (1999). The later version recognizes this more explicitly. 3. This version has an expanded appendix; the later version only includes Appendices A and B.

†Ross School of Business, University of Michigan, Ann Arbor, MI 48109-1234. wlovejoy@umich.edu.

‡Ross School of Business, University of Michigan, Ann Arbor, MI 48109-1234. amitabh@umich.edu.

Abstract

What lines of communication among members of an organization are most conducive to the early, ideation phase of innovation? We investigate this question with a recombination and selection model of knowledge transfer operating through a social network. We measure cost in human time, and seek efficient social network structures in the time—total cost plane (minimize ideation time subject to an upper bound on total cost, or vice versa) and in the time—cost per period plane, with a similar interpretation. Our results suggest that efficiently innovative organizations look nothing like what one intuitively associates with standard formal organizations with strict and unchanging lines of communication, nor do they conform with what one might expect from static social network representations of communication patterns, and we offer variable support for current intuition regarding innovative network structures. We find that ideation is accelerated when people in the organization dynamically churn through a large (ideally the entire population) set of conversational partners over time, which naturally begets short path lengths and eliminates information bottlenecks. In organizations with these features group meetings do not help and can hurt the process, because many parallel conversations can achieve the same or better results as one-to-many communications. A family of networks called the complete wheel-stars emerges as an important family on the time-cost efficient frontier. Wheel-star graphs have a completely connected clique of agents at the center, with all other agents connected to the core but not to each other; the star and the complete graph are its extreme elements. We discuss the consequences of these results for organizations and sociometric analyses.

1 Introduction

This paper considers the ideation phase of innovation in an organization as a process of idea exchange through a social network. The research question is, what is the most efficient network structure? Efficiency here means generating great ideas at minimum organizational cost. Any model of a process as complex as innovation will of necessity be a stylized one, and it is important to understand its strengths and limitations. We address this first, before providing a more detailed model introduction, and our analysis and conclusions.

1.1 The process of innovation

Innovation is central to the enhancement of social welfare and, in the world of business, value-creation and firm survival. Yet, many features of the innovation process pose challenges for academic inquiry. Innovation is the result of complex individual psychological perception and sense-making processes combined with social behavior and communication processes. It is characterized by inherent randomness and path dependence, and (almost by definition) uniqueness. Current approaches to the study of innovation include anthropological studies, empirical studies, and mathematical analysis and simulation (references will be provided below). All three have strengths and weaknesses. Anthropological studies are valuable records of one or a few instances, but their deep rather than broad approach leaves generalizability uncertain. The uniqueness and variance inherent in innovation simultaneously reduce the number of replicates one can access for empirical investigation and increase the number necessary for statistical significance. In analytical models, the complex nature of the process results in an unwieldy number of possible degrees of freedom, requiring that such models focus on some specific aspects of the whole holding all else constant. All three approaches have merit in the general inquiry. Our model is in the third category. Here we consider the ideation phase of innovation in an organization as recombination and selection of ideas communicated through a social network. The research question is, what is the most efficient structural form for the social network? Efficiency here means generating great ideas at minimal organizational cost.

Innovation as recombination and selection through a social network

Much, some say most, innovation occurs by recombining existing things in a novel way rather than the immaculate conception of something totally new. The ideas being recombined must be sufficiently psychologically distant from each other that their recombination is not obvious, for otherwise the new synthesis would be neither distinctive nor innovative. Search or consideration beyond the obvious suggests very weak pre-screens about what is worth looking at. Indeed, lack of prejudgment is a core recommendation in brainstorming exercises, which are designed to encourage novel connections between seemingly disparate concepts. Without pre-judgments about what is productive to look at, the choice of focus among a universe of stimuli is essentially random. Individuals consider ideas selected randomly from the totality of things that he or she can experience, and combine new ideas with existing ones to form new combinations.

Recombination of ideas can create new syntheses but not all of these will be better than the status quo, so some selection mechanism is required to retain advantageous developments and reject disadvantageous ones. This, in turn, requires an objective, so that selection or rejection is based on perceived progress, or not, toward the desired end. A researcher may not know exactly where to look for good ideas, but he or she does have an overall objective in mind and a sense (rightly or wrongly) about whether or not a new idea serves that objective. The intellectual heritage for this model of innovation as search, recombination and selection goes back at least to Schumpeter (1939) and is documented with additional references in Aldrich (2000) and Fleming (2001). The practical heritage of this model goes back much further. Indeed, nature innovates via the random recombination and selection of genetic traits. Mathematically this model finds voice in "genetic algorithms" (GA's, c.f. Goldberg 1989) which are, literally, this natural search process in algorithmic form. GA's have been invoked to search of good solutions to large mathematical programs, but they are also an apt metaphor for the variation, selection and evolution of ideas.

The agents of recombination are people, and social network analysis has emerged as a productive way to bridge the micro behaviors of individuals and the macro performance of the populations of which they are a part. Innovation as the recombination and selection of ideas via communications through a social network is a familiar model (c.f. Liebeskind et al 1996, Powell et al 1996, Nahapiet and Ghoshal 1998, Burt 1997 and 2000, Kraatz 1998, Aldrich 2000, Ahuja 2000, Reagans and Zuckerman 2001, Fleming 2001, Nerkar and Paruchuri 2005 and references therein). In these, interactions are almost always dyadic (that is, between two individuals) conjuring up the image of one-to-one "hallway conversations" rather than one-to-many communication technologies (e.g. email broadcasts or formal meetings where one person can speak while many listen). The communications literature (c.f. Kaufer and Carley 1993), which we return to below, specifically considers the social effect of different communication technologies.

Our basic model is one of dyadic exchanges on random topics, between people matched through a social network, who retain (or not) communicated information based on their sense of whether or not it promotes a shared organizational objective. We derive our main results and intuition regarding the most efficient social network structures in this basic context. We then test the robustness of the results to changes in population size and initial knowledge endowments, extensions of the model to complex landscapes where “apparent” progress toward the objective might be illusory, and to “group meeting” communication technologies.

Existing intuition

There are extensive subliteratures within the study of social networks, communications, and diffusion processes that feature dyadic interactions between individuals in a social network aggregating across people and over time into population-level phenomena. Within these large subliteratures we will focus on papers which yield intuition regarding our research question, which is to identify the best structural form for the social network. Leavitt (1951) designed a laboratory experiment that is similar to our model in many ways. It involved cooperative action by a group of individuals who could only communicate via prescribed channels, and the research question was which pattern was the most effective. Like us, Leavitt defined effectiveness in two ways, the time to reach a solution and the number of messages required, although Leavitt’s “messages” are not controlled for complexity so may not be comparable one to the other. Leavitt assessed four communication patterns in his five-person experiments: the circle, path, star, and Y-shape (which later we will call a 0-2-2 broomstar network, or equivalently, a 5-node binary tree). The results were highly variable, so that statistically significant conclusions were scarce. He did find, however, that the star and the binary tree outperformed the path and circle. These networks had identifiable central agents. From this he concludes that for a problem where it is important to assemble distributed knowledge into one package, having a central depot for information is good. In his experiment, however, one time “period” was one collective passing of messages, regardless of how extensive or complex these were. So, the center of a star could process information from every other agent in the same “time” it takes for him or her to process a single message. If we count cost as person-hours, the star may be less effective, as we shall see.

Granovetter (1973) differentiates between strong interpersonal ties and weak ones, the former benefiting from higher frequency or more time, intimacy, or reciprocity. He identifies “bridges” between communities of individuals as the only path between them. Granovetter claims that a strong tie between two people is likely to be accompanied by at least some tie between their friends, so that no strong tie can be a bridge. This leads to the conclusion that the bridges between thought-worlds that are necessary for ideas to flow are likely to be weak and not strong ties. The network form that this suggests is what some now call a "small world" network (that term has several meanings in the literature), featuring dense clusters of people and just one or a few links between clusters.

Burt (1992) argues that frequently interacting people in dense clusters are likely to know and share common knowledge, so an outsider needs only one link into the cluster to access all of its information. Any additional links into the cluster are redundant. Since links indicate communications that take time and energy, redundant links are inefficient. This suggests that an efficiently innovative social network is either sparse or a small world (in the sense we adopted above). In much of his work, Burt focuses on the power or influence that accrues to an agent who can act as an information gatekeeper along the only link between two clusters, so that if that agent is removed the network breaks into two disconnected subgraphs. Such an agent is said to fill a "structural hole" in the network. Fillers of structural holes (firms or individuals) can themselves be innovative, as they can broker technology from one thought-world to another. Nerkar and Paruchuri (2005) analyze an R&D social network within a chemical and pharmaceutical firm with performance measured by co-patents. They find that central individuals, and filling structural holes, are important drivers of having one's knowledge used by others. We will see below that when we consider communication capacity (talking takes time) people filling structural holes can be information bottlenecks. They may personally benefit from their gatekeeping position, but the organization as a whole is not better off.

The structural intuition that derives from this social network research is that sparse or small-world networks are the most efficiently innovative. The reason is that they can access a wide range of information at minimum cost. That is, it is not that redundant arcs are bad things, it is just that they cost something to maintain and, being redundant from a knowledge and information perspective, are inefficient. This raises the issue of cost and efficiency in social networks, a topic that we pursue in greater detail. There is also some indication, in Leavitt (1951) and Nerkar and Parachuri (2005), that in addition to broadening the reach of ideas available to individuals in the network, some means for channeling these to a place of aggregation can be helpful. This would suggest a star-like network.

There are also authors who advocate for higher densities in social networks than would be suggested by the above intuition (c.f. Coleman 1990), noting that more frequent interactions with fewer individuals may not have the same intellectual reach, but the links that do exist will feature higher levels of trust and reciprocity. We omit these effects by assuming levels of trust to be uniform in the network.

In the communications literature Carley (1991) and Kaufer and Carley (1993) investigate the micro-macro synthesis from individuals to a population using a model nearly identical to our own. They investigate the time it takes a society to reach various population level thresholds, such as social stability defined as universally shared beliefs. Because of its similarity to our model, we will review this work more completely below, in the context of our model definition. Kaufer and Carley (1993) extend the basic, dyadic interaction model to investigate the effect of oral versus print communication technologies on their society-level statistics. Oral communication remains dyadic (one-to-one) and both individuals in a conversation can be changed by the interaction. In contrast, printed works cannot learn or change but can communicate to many in parallel (multiple copies of the same book can exist, and be accessed by many people in parallel). Their final verdict is that print technologies can, under the right conditions, speed diffusion, stability and consensus, but there are qualifications. Print speeds stability and consensus only when it circulates texts containing assimilated knowledge representative of the readers. Print containing avante-garde ideas, not shared by anybody initially, has no advantage over oral communication. In broad strokes, the advantage of print is its ability to be accessed by many people at the same time. The disadvantage is that the printed text does not learn from its interactions, and so cannot be an agent of integration, only communication of what is on its pages. Translated into our context, our restriction to dyadic conversations in our base case may or may not be restrictive. In section 9 we extend the base case to include group meetings, in which one (or several people in sequence) talk while many can listen. Each speaker engages in a one-to-many communication, like print, and based on Kaufer and Carley's results we would expect ambiguous returns for such meetings. Group meetings may help if the presenter has highly evolved knowledge (a very "learned" individual) relative to the rest of the assembly, so that a randomly chosen topic of conversation is likely to be instructive for all. But, they would not be helpful if he/she was differentiated only by isolated novel bits of information, because the probability of hitting exactly the right bit to improve the ideation process is low. Unless an individual speaker is more likely to move the process forward than a randomly chosen speaker, the group will invest a significant amount time (in person-hours) in a meeting, with a high probability of no compensating benefit relative to many parallel conversations. The perceived advantages of one-to-many broadcasts stem from diffusing ideas more rapidly en masse, but that intuition does not consider that the broadcaster may in fact have little of value to impart, yet he/she will dominate many person-hours of listeners' time. This practically familiar feature of meetings is a common theme of business-related humor. We repeat that our focus is on the ideation phase of innovation, and our results cannot be assumed to extend to post-ideation product development and commercialization. In the ideation phase organizational members are aware of broad objectives but have no good ideas yet about how best to achieve them. Once an actionable idea is embraced, downstream activities may feature consciously divided labor, targeted search, and group meetings featuring sub-team report-outs and planning processes. Our research does not shed light on how best to organize those downstream activities.

Another relevant existing literature studies the diffusion of innovation, technologies or diseases through populations (of firms or people). Again, the micro-level activity of dyadic interpersonal interaction drives, in aggregate, dynamic effects in the population. This literature typically takes the network structure as fixed (as is appropriate if speaking of a particular industry or social group) and analyzes the extent and rate of diffusion in the specified network. While this literature highlights some important structural dependencies it typically does not ask directly which structures are best (or worst). For example, Valente (1995) reviews existing research of this type and concludes that both relational (neighbors via direct ties) and structural (the overall structure of the network) characteristics affect diffusion, but the differential influence of each in isolation is not clear. Watts

(1999) provides simulation results for the spread of infection in a parameterized family of graphs that varies between highly structured ring-like graphs to randomly generated graphs. He shows that the time it takes a disease to spread, starting from a single infected individual, drops along this range. This is intuitive since (connected) random graphs tend to have short path lengths. However, the diffusion time reaches close to minimal values when the graphs are still in the small world range (short path lengths but highly clustered), well before the experiments reach random graphs. So we can infer from this that short path lengths are conducive to the rapid spread of disease, with or without high clustering. We can also infer that connected random graphs are likely to be effective in spreading disease (or ideas).

Below we show the advantages of randomly chosen conversational partners. It is important, however, to distinguish between randomly chosen partners and randomly generated graphs. The former means, essentially, that there is no structure to who talks to whom. The latter is a specific graph generated by randomly adding edges. So, the rapidity of propagation through randomly generated graphs is different than the advantages of randomly chosen partners each time period. However, the diffusion literature does suggest that short path lengths are key structural features.

In summary, these literatures suggest that efficient innovative social networks will have a sparse or small-world structure, and potentially feature some central agent or group in which knowledge accumulates. Short path lengths are good for the rapid dissemination of ideas, and there are ambiguous returns for one-to-many communication technologies. Our work contradicts some of these tentative conclusions, and supports others, as discussed in the final section of the paper.

2 Base Case Model

In our model we assume that an organization is a purposeful social unit and that each individual shares a desire to fulfill the purpose of the collective. People interact though the social network, exchanging ideas in conversation and retaining ideas that they believe are productive vis-á-vis the organizational goals. The choice of topics is random, reflecting lack of prejudgment over what ideas might be most productive. We abstract away from the cognitive and relational aspects of communications to focus on the structure of the social network. We assume that a great idea exists, and can be assembled by combining a series of individual idea bits that, ex-ante, are distributed throughout the organization. Conversations take time, and therefore have an opportunity cost. We seek efficient network structures, those that can generate great ideas at minimal cost.

2.1 Knowledge evolution

In our model agents have knowledge, beliefs or world-views represented by binary strings that are altered in conversations with others. Here we describe our base case model, which contains assumptions that will be relaxed in section 8 when we test the robustness of our base case results.

Agents begin with knowledge strings of 1's and 0's representing their current beliefs or world-view. They meet along communication lines (determined by the social network structure) and converse over a randomly chosen bit in their knowledge string. They exchange perspectives on the chosen topic, represented by a 1 or 0 in that bit, and either agent can adopt the views of the other if he/she feels it is better. In this way, indvidual beliefs change over time as a result of interactions with other agents.

We assume the existence of one best solution or "great idea," which without loss of generality we assign to be the 1-vector (a string of all 1's). Agents cannot anticipate the nature of the 1-vector solution ex-ante, but in the base case they can recognize progress toward that goal. If any agent does realize the 1-vector (puts all of the pieces together) there is an "a-ha" moment when he/she recognizes the idea as great (c.f Shilling 2005), and a good basis for more formal and specific development. Hence, in our base case, the ideation process stops when one person in the population assembles the 1-vector in their mind. In section 8, we consider a more sophisticated model of world-views and knowledge, and obtain qualitatitively similar results.

If two people meet in conversation and they both have a "1" or both have a "0" in this bit location, they both agree on that topic and leave the conversation unaltered. However, if one agent has a "1" in the string and the other has a "0", then only one of them (the one with the "0") will recognize the other's idea as an improvement over their current beliefs, and will upgrade her beliefs accordingly. So, exactly one agent will leave with an upgraded string and knowledge, and the other will leave unaltered. The situation is illustrated in figure 1, where there are four agents networked into a ring, with their initial (5-bit) belief strings as shown. If agents 1 and 2 converse and discuss the first bit, they leave unaltered since they agree on that dimension of the problem. If they converse and discuss the second bit then agent 1 will consider changing from 10100 to 11100. That is she will consider the idea proposed by agent 2. Likewise agent 2 will consider changing from 11000 to 10000. Since both can recognize progress, agent 1 will change to 11100 and agent 2 will remain at 11000. Since conversational bits are chosen at random and each knowledge string has 5 bits of information, there is a $2/5$ probability that agent 3 will learn something in conversation with agent 1, and a $1/5$ probability with agent 4. Agent 1 has only a $1/5$ probability of learning something from agent 2, and no chance of learning from agent 3.

Figure 1 near here.

This model has some intuitively attractive features as a representation of the evolution of knowledge in a group. First, after a sufficient number of encounters all agents in the ring will share the common beliefs 11100, that is a group perspective will evolve in a closed group after many conversations. Because selection always moves the group forward (in the base case), their equilibrium knowledge will maximize the group's performance within the limits of their collective imagination. But, this group will always maintain 00 as the final two bits, because all members come in with those preconceptions and so they will never be changed in conversation. This is the darker side of group-wide perceptions (group-think in the pejorative); if all individuals come in with a common set of assumptions these will never be challenged. For this group to move further toward the best solution a new member or members with new ideas (1's in one of the last two dimensions) will have to be introduced and listened to.

For most of this paper we assume that agents match randomly within the network structure (some strategic matching policies will be analyzed in section 6), using the following process. An agent is chosen at random and that agent contacts one of her network neighbors at random, and if that neighbor is not already engaged the two converse. Those two agents are then removed from consideration and the process is repeated, until all agents have been considered for that period.

Matchings are independent across time periods. We refer to this matching process as "random" matching (there are various definitions of "random" matching in the literature; we mean the process just described).

We do not explicitly include some realistic influences that complicate actions and interactions in actual organizations, including opportunism linking unilateral gains to losses for others, personality conflicts, explicit pressures for conformity, or explicit resistance to changing one's belief system. Some of these influences are there indirectly (for example, conformity of beliefs arises in our model in a closed network through the exchange process), but in general we use a parsimonious model of interaction that is broadly consistent with existing models of organizational learning yet simple enough to focus on the effect of the organizational structure without being confounded by a host of other considerations.

2.2 Organizational Performance

Consistent with managerial custom we will assess organizational designs on an efficiency basis. Efficiency means that a given level of performance is achieved at minimum cost. The cost of an idea to the firm is the time its human assets invest in the conversations that lead to it. A network structure is efficient if it achieves a given level of performance at minimum cost.

In contrast with most of the social network literature, we explicitly model people's finite capacity for interactions. That is, due to the fact that interactions take time and there are only a limited number of hours in a day, people can only have a finite number of interactions with colleagues in any finite unit of time, such as a day. Most social network analyses do not include this capacity feature of human interaction, yet it is an important aspect of management in real systems and is just now gaining attention. See, for example, Sosa et al (2007) for empirical validation of human capacity effects on design teams. In contrast, the communications and diffusion literatures explicitly model time in their analyses (c.f. Kaufer and Carley 1993 and Newman et al 2006).

We assume conversations are of uniform length and use that duration as our unit of time, so that any individual can have at most one conversation per time period. So, each time period we match people through the social network structure, these matched pairs interact, and we move to the next time period. We denote by TTF (for "time to finish") the total time it takes to reach the solution. In practice TTF will contribute to time-to-market. We measure organizational cost in two ways, as an absolute cost and as a cost rate. The first is the total number of conversations required to reach the solution. This reflects the opportunity cost of agents' time, because while they are in conversation they cannot be working on other tasks. We call this $NCTF$ for "number of conversations to finish." The second cost metric is the average number of conversations per period (CPP) that the network structure implies. This is a cost per period metric, which would be operable in a firm with only a fixed amount of spare human resource capacity per time period.

We should contrast our model of the cost of a network with intuition standard in the social networks literature, where an arc represents some investment of time or energy in a relationship (c.f. Granovetter 1973, Uzzi 1996, Kraatz 1998, Reagans and Zuckerman 2001) so "number of arcs" is a proxy for the cost of (or energy required to maintain) a sociomatrix. The implied investment is active interaction, and when constructing sociomatrices researchers draw arcs based on questionnaires or data that reveal who interacts with whom on a regular basis (c.f. Baker 2000). If interactions are sufficiently regular or intense there is an arc and if not there is not. But, the real energy is in the interactions. That is, "number of arcs" is only a rough representation for the more fundamental investment that is the time and effort to interact. We measure that fundamental investment directly.

We assume that both time to reach a solution and the cost to reach the solution are important performance metrics for the firm, and we seek an efficient frontier of network types in the time-cost plane. We will do this for each of the two cost concepts ($NCTF$ and CPP). It is well-known (c.f. Cohen 1978) that the efficient frontier provides the set of solutions to a series of optimal design problems, for example minimizing TTF subject to $NCTF$ being less than or equal to a specified upper bound or minimizing $NCTF$ subject to TTF being less than or equal to a bound. As we alter the bounds, we generate the efficient frontier of networks in the $TTF - NCTF$ plane. We will also construct the frontier for TTF versus CPP, with a similar interpretation.

In our base case we initialize each network in a distributed knowledge configuration. If there are n agents in the network and m bits in a knowledge string, a distributed knowledge network has

$n \geq m$, m agents starting with a different unit vector, and $n - m$ agents start with all 0's. With this as a starting point all of the knowledge required to discover the solution (1-vector) exists in a distributed fashion somewhere in the organization, embedded in their belief strings. Arriving at the solution will require combining the knowledge of m different agents in the network. In section 8 we test the robustness of our conclusions with more varied initial knowledge endowments.

In some innovative contexts it may be sufficient for just one member of the organization to find the solution. In others, it may be necessary for all members to realize the solution before we declare that they are finished. For $1 \leq x \leq n$ define $TTF(x)$ to be the total time until x members of the organization have attained the optimal belief string. We will use $TTF(all)$ to represent $TTF(n)$ to emphasize the fact that all members must be on board before we stop. We define $NCTF(x)$ analogously. We focus on $TTF(1)$, under the assumption that once a great idea is recognized by an individual it can be communicated to others more efficiently than through random exchanges of partial ideas. However, because it is costless to collect and because there is a lot of consistency in the $TTF(x)$ data, we will occasionally report on $TTF(all)$ and the intermediate value $TTF(3)$ if the results differ among them. We note that $TTF(all)$ is comparable to Kaufer and Carley's (1993) time to social stability, because once all agents share the 1-vector no further changes in belief strings will occur.

3 Analytical Method

Because agents choose randomly among their network neighbors, the knowledge trajectory, TTF and $NCTF$ are random processes. Deriving our two efficient frontiers is, in theory, aided by the provable fact (see appendix D) that $E[NCTF] = CPP \times E[TTF]$. With random matching, CPP is an intrinsic feature of the network structure, like its density or diameter. Analytical expressions for CPP are available for some graphs (c.f. appendix B) and can be computed for others. So, computation of $E(TTF)$ is sufficient to compute $E(NCTF)$. Unfortunately, analytical expressions for $E(TTF)$ are unavailable for all but the simplest of graphs. For example, suppose agent i has a 1 in bit location j, and we are interested in the time it would take to transmit this 1 to agent k along a path between i and k. If i and k are neighbors, denote the probability that they converse in any time period by p_{ik}. If they converse, the desired bit is selected with probability $1/m$ (recall m is the number of bits in a knowledge string). So, the probability that the desired bit of knowledge is transmitted in any time period is p_{ik}/m, and the time until transmission is geometrically distributed. This has some intuitive properties, in that the more frequently people talk (higher p_{ik}) and the narrower the intellectual landscape open to exploration (smaller m) the faster (stochastically) the specified idea will be communicated. These natural qualitative features extend to multi-party paths of conversations in networks, but are of little use analytically.

If i and k are not neighbors the time until the bit of knowledge is transmitted along any specific path beween i and k is the sum of geometric random variables (one for each arc along the path), which has a negative binomial distribution if the geometric random variables are independent and identically distributed (i.i.d) but has no analytical expression otherwise. To be identically distributed each arc along the path must have the same probability of being active in conversation each period (the p_{ik}'s must be equal), which is only true for very few network types (e.g. stars, complete graphs, cycles). Even if the path between any two nodes is unique (that is, the graph is a tree) the time for an agent to become completely learned (reach the 1-vector knowledge string) is in general the maximum of a set of random variables (one for each required bit of knowledge), each of which is the sum of non-identical geometric random variables, which has no known analytical expression or close approximation. We note that the star is the only tree network in which all arcs have the same probability of being chosen for conversation each period. All of the other examples are not trees, and further the path between any two nodes need not be unique which compounds the analytical problems. For example, in a complete 10-node graph there are $\sum_{k=0}^{8} \binom{8}{k} k! = 109,601$ distinct paths between any two nodes, and each of these paths may be responsible for a bit of knowledge propagating from one agent to another. Analytically computing TTF or $NCTF$ in general graphs is unavailable with current technology.

There are some approximation methods that can be useful for specific graph types. Grabner and Prodinger (1997) present an approximation for the maximum of a set of i.i.d. negative binomial random variables, but the i.i.d. requirement reduces the utility of this result in our setting. The evolution of knowledge in a network can be cast as a Markov chain, but with 2^{nm} states these are too large for meaningful analysis. Even in small, highly structured examples the analysis is essentially numerical and not particularly insightful. In general, numerical simulation is the only technology that allows us to accurately compute network performances for a set of graphs sufficiently large to span the space of interest for our research question.

Calibration runs suggested that we could generate margins of error (corresponding to 95% confidence intervals) for our metrics on the order of 1% by using 1000 replicates for each experiment (some representative 95% confidence intervals are reported in table 1 below). This allows us to order the performance of alternative networks with statistical confidence. For the remainder of this paper, this is the technology we employ.

4 A naive intuition

It is instructive to look at some preliminary results on standard graphs to begin building an intuition. We investigated the learning rates for different organizations by simulating one thousand replicates of each of five standard 10-agent test networks: a complete graph, cycle, path, binary tree, and star (all graphs used in this paper are described and illustrated in appendix A). Other than the complete graph, the other 4 are 10-node versions of the 4 networks studied by Bavelas (1950) and Leavitt (1951). We took data on $TTF(1)$, $TTF(3)$ and $TTF(10) = TTF(all)$. The results were consistent, significant and unambiguous. For all stopping criteria the TTF for the test networks were ordered as follows:

$$\text{Complete} < \text{Cycle} < \text{Path} < \text{Binary Tree} < \text{Star.}$$

The differences in mean TTF's were extremely significant (p-values close to zero for pairwise t-tests). The star was significantly worse than the others. The TTF distributions were skewed and heavy-tailed to the right, as one might expect since they are truncated below at zero. But, these features were preserved even for the longest $TTF(all)$ times, which were bounded well away from zero.

	mean	stdev	min	max	95% CI	CPP
Star	199.7	75.3	50	545	[195.0, 204.3]	1.00
Binary tree	135.0	45.0	49	380	[132.2, 137.8]	3.30
Path	105.5	27.2	40	199	[103.8, 107.2]	4.26
Cycle	76.4	13.8	37	126	[75.5, 77.2]	4.40
Complete	36.1	6.7	16	70	[35.7, 36.5]	5.00

Table 1: $TTF(1)$ statistics for some standard networks

The variances increased with the means so the above TTF ordering works for the variance of TTF also. In most cases these differences can be confirmed statistically (using robust variance tests, c.f. Levene 1960 and Brown and Forsythe 1974). In addition to being significantly different from each other, the absolute level of variability was large, consistent with intuition regarding the variability of creative exercises. The statistics for $TTF(1)$ for these graphs are shown in table 1 (the results for $TTF(3)$ and $TTF(all)$ are similar).

What drives this ordering? Intuitively, since knowledge flows via conversations the time to completion should be significantly affected by the number of parallel conversations allowed each time period by the organizational design. When conversational capacity is considered, a star allows only one conversation per period ($CPP = 1$), whereas a complete network with an even number of agents (n) has $CPP = n/2$ as every agent talks to somebody. It is shown in appendix B that with $n = 10$ agents the theoretical average number of conversations per period for these networks are as shown in table 1. It is tempting to conclude that TTF is driven by CPP, which would lead to simple organizational design guidelines.

This intuition is misleading, however. Figure 2 shows two 9-agent networks, a “spider” and the complete network. The CPP for both of these networks is 4, yet they will have very different performance characteristics.

Figure 2 near here.

Returning to our 10-agent networks, a spider graph has the CPP of a cycle ($CPP = 4.4$) but has much worse performance ($TTF(1) = 160.4$), even worse than a binary tree . In contrast, a "wheel-star 17" graph (the wheel-star family of graphs will be described below and in appendix A) has $CPP = 1.98$, lower than a binary tree, but $TTF(1) = 96.6$, better than a path. Clearly more is going on than just the number of conversations per period. Just as clearly, we would have missed these counterexamples had we restricted our attention to standard network types. In what follows we will want to generate a sufficiently large and diverse set of networks to be sure we are covering the landscape of graphs, so that our conclusions are robust. The next section describes how we generate our test set of networks, and details of the simulations.

5 The test landscape of graphs

One way to test hypotheses in simulation is to generate large numbers of random graphs (in this discussion we are referring to traditional "Bernoulli" random graphs, introduced by Erdös and Renyi 1960) as the test set . Random graphs, while having advantages in estimation and hypothesis testing, are not sufficient for our task. Random graphs can congregate at levels strictly interior to theoretical bounds on common network statistics, yet it may be the extreme graphs that exhibit optimal learning performance. If we generate each of the $\binom{n}{2}$ possible arcs in an n-agent graph with equal probability p, we tend to get disconnected graphs for p low and relatively dense graphs for p high. The sort of sparse, connected networks that conventional wisdom suggests we want for innovation are not likely to arise randomly. For example with 10 agents the probability of generating a simple star is about 1.7 in a billion, and of generating a $WS17$ graph is about 5 in a trillion, as discussed in appendix A.

In recent years, a rich literature has developed on other models of randomly generated graphs which mirror certain structural properties of social networks. For instance, Watts and Strogatz (1998) propose a model of small world graphs (an alternative, and more specific definition than we invoked above), where path lengths are much smaller than Bernoulli random graphs despite a similar numbers of edges. Barabási and Albert (1999) proposed a different model of random graph generation which displays the power-law property of nodal degree, and called these "preferential

Characteristic	Min	Max	Min example(s)	Max example(s)
CPL	1	$\frac{2+(n-1)(n+2)}{3n}$	complete graph	path
diameter	1	$n-1$	complete graph	path
Density	$\frac{2}{n}$	1	star or any tree	complete graph
Clustering coefficient	0	1	star or any tree	complete graph
Degree centrality	0	1	cycle	star
CPP	1	5	star	complete graph

Table 2: Graph metrics

attachment" networks. A comprehensive survey of large classes of random graphs, and the properties of the real social networks that they model, appears in Albert and Barabási (2002). We include a total of 30 random graphs from all three families (Bernoulli, small world, and preferential attachments; as described in appendix A) in our test set for completeness. However, we augment these with a set of graphs deliberately constructed to span the theoretical space.

First, we generated graphs that span the feasible range of commonly invoked characteristics: critical path length (CPL), diameter, density, clustering coefficient, and degree centrality (c.f. Wasserman and Faust 1999 for definitions of these features). As a result of our preliminary tests we also included CPP in this set. The theoretical bounds on these characteristics in a graph of size n are shown in table 2 (c.f. Wasserman and Faust 1999, Lovejoy and Loch 2003), along with some standard graphs that achieve these bounds.

We generated a test set of graphs to occupy, to the extent possible, a continuous range between these extreme values. Complete descriptions and data are provided in appendix A. In addition to spanning these ranges, we wanted to specifically include graphs with theoretical interest. These include "clique-paths" (these maximize CPL, c.f. Lovejoy and Loch 2003, and have a complete graph and a path as their extreme forms), "brooms" (chosen to combine stars and paths), "spiders" (these can maximize CPP without being dense), "wheel-stars" (constructed to minimize CPP for a given number of arcs), and other graph types deliberately designed to express a range of network characteristics. The resulting test set of 108 different graphs, along with their statistics, is shown in appendix A. We note that our random graphs fall strictly interior to some important boundaries of our landscape. Indeed, as will be seen below, we would have missed the efficient frontier had we relied on random graphs alone in this investigation, as the required graph types would be very unlikely to appear even in large sample sizes.

All of our graphs have 10 agents each with a 9-bit knowledge string. For each of our 108 graph types we ran 1000 simulated ideation trajectories, keeping track of the statistics necessary to generate two efficient frontiers: TTF versus $NCTF$ and TTF versus CPP.

6 $TTF - NCTF$ efficient networks

Recall that the efficient frontier in the $TTF-NCTF$ plane is sufficient to identify optimal network structures for minimizing TTF (time to reach a great idea) subject to any upper bound on $NCTF$ (total person-hours) and/or minimizing $NCTF$ subject to an upper bound on TTF. Figure 3 shows our test graphs in the $TTF(1) - NCTF(1)$ plane. The most striking feature of this plot is that there really is no "frontier." Rather, the set of graphs comes to an efficient point, which is the complete graph, which minimizes time *and* cost. This is true regardless of the criterion ($TTF(1)$, $TTF(3)$, or $TTF(all)$).

Figure 3 near here.

Why is the complete graph efficient? Although the complete graph maximizes the number of conversations per period, this is not the reason it dominates all other network structures, as noted above. The advantage of the complete graph is that agents have many different potential conversational partners and move among them randomly over time. This avoids spending too much time talking to the same person, which would promote alignment of thinking but less potential for learning. The spider, for example, has poor performance because most of its conversations are between a fixed pairing of agents, so that these pairs will quickly come into alignment (after which conversations between them are unproductive).

Our preliminary conclusion, prior to robustness tests, is that in the ideation phase of an innovative effort *no* stable communication structure is optimal when minimizing either time or cost.

Rather people should change conversational partners frequently among the entire set of people in the organization. So, the conventional sociometric questionnaire resulting in a stable set of arcs is not adequate. Rather, the key feature one would look for is instability in the arcs over time. Also, if talking to the same partner repeatedly is bad, and changing partners often is good, we may be able to recommend a non-random conversational policy that out-performs the complete graph. In the next section we look for theoretical and empirical justification for such a policy.

6.1 Deliberate (non-random) learning strategies

We have assumed that agents choose conversational partners randomly among their neighbors in the network design, and concluded that the complete graph is efficient in the $TTF - NCTF$ plane because it combines high CPP with no constraints on one's choice set, reducing the probability of talking to the same person repeatedly. The resulting active chatter and churning of partners is productive in the ideation phase of innovation if the required knowledge bits are distributed throughout the organization. This conclusion would be reinforced if deliberate churning (for example, never talk to the same person twice in a row) improves performance still further. Can a non-random conversational matching policy out-perform a random one? A conscious matching policy must designate a strategy for who to talk to next in the organization, as a function of only those things that are observable, which for each agent is "Who did I talk to?" and "Was the conversation productive?" (did I learn anything?).

Sequential decision problems where decision makers only observe noisy signals of a more detailed underlying stochastic process can be cast as Partially Observed Markov Decision Problems (POMDP's), a well-known but very difficult class of problems (c.f. Smallwood and Sondik 1973, Lovejoy 1991) that is largely intractable except for problems of special structure or small size. However, as described in appendix C, there are approximations to the POMDP representation of the problem of who to talk to next in an innovative social network that fall into a more tractable problem class, called Bandit Problems (c.f. Gittins and Jones 1974, Gittins 1979, Whittle 1998). The strategy space for the general problem is too large to be manageable, but in appendix C we use the bandit approximations to identify some specific conversational strategies that might outperform random matching in the complete graph. Here we present these results intuitively.

First, if there is very little left to learn (which an agent cannot know but would be more true later than earlier in the process) then it is effective to talk to the person you have not spoken to for the longest time. We call this the "Longest Elapsed Time" policy or LET. The reason is that if, say, there is just one piece to the puzzle missing and a conversation with your current partner supplies that piece, then the ideation process ends because you have just realized the great idea. If the process continues, then your conversation with your current partner must have failed which decreases your estimate that they have something to offer. Meanwhile, if others in the organization are talking amongst themselves they may be gaining that one critical knowledge bit, so they will be preferred as your next partner over anybody you have recently talked to unproductively. Of course, agents cannot know what they need to complete the idea, nor what other agents know, but they do not have to. In an expected value sense, as long as the process is ongoing it is optimal to always switch partners to somebody you have not talked to in a long time. One way to implement this deliberate churning is via LET.

In contrast, if there is a lot to learn (again, the agent cannot know this but the situation is more likely early in the process), then learning something in conversation does not end the process, and further it is a potential signal that your current partner has a wealth of knowledge that you can benefit from (after all, a random conversation turned out to be productive). In this case, you want to continue with the same partner, a strategy we call "stick with a winner" or SWW. Combining these two, we propose that the decision maker begin by choosing a neighbor at random, and then sticking with that partner as long as he or she learns something in conversation. Upon the first occasion of non-learning, she should choose an alternative partner using the LET logic, and then stick with that partner until failure, etc. We will call this the SWW-LET policy. We tested both the LET and SWW-LET policies for the complete graph in our simulation. The average results are shown in table 3 .

LET is statistically significantly better performing than random matching, but SWW-LET is either worse or statistically indistinguishable from random matching. There are several possible explanations for this. First, the combination of diminishing probabilities for learning from repeat

	$TTF(1)$	$TTF(3)$	$TTF(all)$
Random	36.07	43.97	72.26
LET	35.49^{*}	43.30^{*}	71.09^{*}
SWW-LET	36.67^{*}	43.20	72.19

* in LET row, p-value for (LET $<$ random) $< 5\%$

* in SWW-LET row p-value for (random $<$ SWW-LET) $< 5\%$

Table 3: TTF values for strategic conversational strategies

conversations and the parallel learning opportunities others (not in the current conversation) enjoy combine to make churning conversational partners better than "sticking with a winner." Second, the SWW strategy may not, in a network setting, be all that different from random matching. Each time a conversation results in learning, one partner will want to repeat with the same partner, but the other does not. When people call partners in random sequence, an SWW strategy may not translate into a lot of repeat conversations anyway. Evidence that the former explanation is more operative than the latter is that, while random matching and SWW-LET perform similarly, LET significantly outperforms SWW-LET.

Another contingency that might influence the difference between LET and SWW-LET is the balance of knowledge throughout the network. Recall that the set-up for the analysis of the SWW policy included the assumption that one agent has a lot to learn and her neighbors have a lot to offer. This implies a very asymmetrical distribution of learning endowments. All of our base case experiments begin with roughly equal, mimimal, knowledge endowments.

To test these results for varied levels and distributions of starting knowledge endowments, we ran a matched triplet of experiments testing the three conversational strategies (random, LET, and SWW-LET) on randomly generated starting belief strings. Specifically, we generated starting knowledge strings by letting each bit for each agent be a 1 with probability p and 0 with probability $1 - p$, and let $p = .3$, $.5$, and $.8$. The resulting starting beliefs exhibited asymmetrical knowledge, with bit sums ranging from 0 to 4 across agents when $p = .3$, ranging from 2 to 6 when $p = .5$, and ranging from 6 to 9 when $p = .8$. Starting with each of these three belief scenarios we ran parallel experiments using the three conversational strategies. The results of these experiments consistently support the hypotheses that LET dominates random matching, and random matching dominates SWW-LET (in that all differences that were statistically significant at 5% satisfied these inequalities). This was true for all criteria $TTF(1)$, $TTF(3)$, $TTF(all)$. We repeated the experiments with $NCTF(1)$, $NCTF(3)$, and $NCTF(all)$, with identical results. So deliberate churning of partners is strictly better than random matching in minimizing both TTF and $NCTF$, confirming that it is not CPP per se, but the constant changing of conversational partners, that is the key feature driving rapid ideation.

6.2 Statistical tests

In our base case the complete graph minimized *both* $TTF(1)$ and $NCTF(1)$ simultaneously, making it the unique efficient graph. However, there is a cluster of graphs near the efficient point, some of which might be statistically indistinguishable from the complete graph. We tested the null hypothesis that each of these alternative graphs was better than the complete graph on either of these dimensions, so that rejection of the null is strong evidence that the complete graph is the unique efficient graph. For $TTF(1)$, we can reject the null hypothesis (all tests featuring p-values below .003) for all alternative graphs in the landscape except $WS44$, which is the complete graph minus one arc. So, the complete graph or something very similar is the unique minimizer of ideation time. There are several graphs for which we cannot reject the analogous null hypothesis (at 5% level of significance) for $NCTF(1)$, leaving open the possibility that some graphs may feature higher $TTF(1)$ but lower $NCTF(1)$ than the complete graph. All of the potential alternative graphs, however, are again almost complete in that they are missing but a few arcs and share most of their essential features with the complete graph. This is not surprising, since for any graph family that converges to the complete graph as we add arcs, there will be a neighborhood of the complete graph containing multiple graphs that look and perform similarly. Specifically, at a 5% level of significance we cannot reject the null hypothesis that the following graphs have mean $NCTF$ less than that of the complete graph: WS44, WS43, SW9, WS42, and WS39. All of these feature high densities, high CPP, low CPL and diameters, high minimum nodal degree and have

Graph	$TTF(1)$	$NCTF(1)$	CPP	CPL	Density	Dia.	Min. deg.
Complete	36.07	360.65	5.00	1.00	1.00	1	9
WS44	36.25	360.81	4.98	1.02	0.98	2	8
WS43	36.93	365.67	4.95	1.04	0.96	2	7
SW9	36.96	362.41	4.90	1.11	0.89	2	7
WS42	37.27	365.72	4.91	1.07	0.93	2	7
WS39	38.48	363.97	4.73	1.13	0.87	2	6

Table 4: Close-to-optimal graphs for minimizing $TTF(1)$ and $NCTF(1)$

no information bottlenecks (see table 4).

6.3 Summary of the base case in the $TTF - NCTF$ plane

We conclude from these results that in the $TTF(1) - NCTF(1)$ plane we should seek complete-graph-like qualities in the social organization, specifically a lot of parallel conversations per period (high CPP), high churning of conversational partners (each person talking to a lot of other people in the organization, reflected in high densities and nodal degrees), low CPL and diameter, and no information bottlenecks. This suggests that no static network structure is optimal, and social network empiricists might wish to look for (and encourage) unstable networks, at least in the upstream ideation phase of an innovation process. This intuition is reinforced by the enhanced performance of the LET policy, which deliberately churns conversational partners rather than leaving that to chance.

7 Minimizing TTF subject to a constraint on CPP

The second efficient frontier that we want to investigate is in the TTF-CPP plane. Such a frontier would allow us to identify networks that minimize time (TTF) subject to an upper bound on cost per period (CPP), or vice versa. Figure 4 displays the complete test set of networks in the TTF-

CPP plane, the efficient frontier being its lower convex hull. The efficient frontier is completely determined by a special subset of a single family of graphs, the "wheel-stars." These networks are described below and in appendix A.

Figure 4 near here.

Moving from low to high CPP the efficient frontier is traced by (these graphs will be described shortly) the $WS9$ (which is a star), $WS17$, $WS23$, $WS30$, $WS33$, and $WS36$ followed by a denser set of graphs culminating $WS45$, which is the complete graph. The "WS" graphs are "wheel-stars," a family of graphs deliberately constructed to add conversational options without greatly increasing CPP. We do this by beginning with a star graph (the only graph with $CPP = 1$) and then sequentially adding arcs to agents who are already highly connected. Since an agent can only speak to one person at a time, adding an arc to a highly connected agent increases their conversational options without significantly increasing the number of conversations they will have. This can simultaneously achieve high churning of partners with low CPP. But not all wheel-star graphs are on the $TTF - CPP$ efficient frontier. Rather, this frontier is defined by graphs that are either in or very close to a specific subset of that family, which are "complete" wheel-stars. These are described next.

We use the label "WSk" to refer to a wheel-star graph with k arcs. For a fixed number of agents k uniquely identifies the wheel-star graph structure. To generate the wheel-star family we start with a star ($WS9$ for 10-agent graphs) and then successively add an arc from any non-fully-connected agent to the available node with the highest nodal degree (breaking ties randomly). For example, $WS9$, $WS10$ and $WS17$ are shown in figure 5. In that figure, we would go from $WS10$ to $WS11$ by adding an arc from a member of the set $\{4, 5, ..., 9, 10\}$ to a member of the set $\{2, 3\}$, because the latter set contains the available (not already fully connected) agents with highest nodal degree.

Figure 5 near here.

The "complete" wheel-stars are those graphs with a completely connected clique of agents in the center with links to all agents in the population, and with all other agents arrayed around the

Network	$TTF(1)$	CPP
Star (=WS9)	199.66	1.00
WS10	182.36	1.88
WS17	96.63	1.98
WS24	65.32	2.90
WS30	49.49	3.72
Complete (=WS45)	36.07	5.00

Table 5: $TTF(1)$ and CPP statistics for some wheel-stars

perimeter connected to each member of the central clique but not to each other. For example, in figure 5 the star $WS9$ is a complete wheel-star with a central clique of 1 agent, and $WS17$ is complete with a central clique of 2 agents. The other complete wheel-stars (in a 10-agent graph) are $WS24$, $WS30$, $WS35$, $WS39$, $WS42$, $WS44$ and $WS45$, the last being the complete graph. The complete wheel-stars are identical to the "idealized core/periphery" networks as defined by Borgatti and Everett (1999), while the incomplete wheel-stars display a high degree of core/periphery structure but are not "idealized". In our base case experiments the complete wheel-stars are either unambiguously on the efficient frontier or statistically indistinguishable from it, and this is true whether we are looking at CPP versus $TTF(1)$, $TTF(3)$, or $TTF(all)$.

Why are the wheel-stars efficient network designs? Table 5 shows the $TTF(1)$ and CPP values for some wheel-stars. The dynamics of $WS9$ (a standard star) are well understood; at each time period, the star center talks to one other agent, and each of the nine leaves have an equal probability of being chosen. Now consider $WS10$ and an arbitrary time period. There are only two possible conversational pairings: (i) Agent 1 talks to one of agents 4, 5, $\ldots$, 10 and agents 2 and 3 talk to each other, and (ii) Agent 1 talks to either agent 2 or agent 3, and no one else talks (c.f. figure 5). The first case occurs approximately 88% of the time, and in that case agent 1 has a relatively high chance of learning or disseminating new information, because of the diversity of agents she talks to. Agents 2 and 3, on the other hand, exchange relatively little information, because most of their time is spent talking to each other. The network is in a sense dichotomous, with agents 2 and 3 in one partition and everyone else in the other. Agent 1 is the only conduit by which information can flow between these two partitions, but converses across this bridge only 12% of the time. Thus although there are an average of 1.88 conversations per period (almost twice as many conversations per period as the star) the additional conversations are not very helpful because many are confined to agents 2 and 3 talking to each other. Therefore, $TTF(1)$ decreases some but not too much, despite CPP almost doubling. As we add more arcs between $WS10$ and $WS17$ we do not greatly affect CPP but we do increase the available conversational partners, which decreases TTF. So, it is efficient to increase the number of arcs in this range. This ends at the next complete wheel-star, $WS17$.

$WS17$ consists of two star centers, agents 1 and 2, with each leaf connected to both star centers. In this network, once again, there are two primary conversational outcomes in any time period: (i) Agent 1 talks to one of agents $3, 4, \ldots, 10$, and agent 2 talks to another of agents $3, 4, \ldots, 10$, and (ii) Agents 1 and 2 talk to each other. The first case occurs with probability 98% and the second case with 2% probability. In the first case, we really have two stars running in parallel, with commensurate information exchange gains. Since every leaf is connected to both star centers, there is no information bottleneck (no partition of agents or information) and free flow via the leaves takes place, along with free flow between the two star centers when those conversations occur (the second case). The two star centers essentially learn in parallel, and periodically compare notes. This explains why the $TTF(1)$ for this network is just slightly better than half of the $TTF(1)$ for a star (96.6 vs. 199.7). When we consider $WS18$, however, we effectively start a third star center. The third star center is almost always forced to talk to the only leaf she is connected to, while the other 2 star centers operate like distinct stars running in parallel. Thus $WS18$ performs only marginally better than $WS17$ in $TTF(1)$, while expending considerably more energy (CPP), repeating an earlier story. Continuing in this fashion, as we add arcs toward the next complete wheel-star ($WS24$), we get increasingly better performance for only marginal increases in CPP because we add conversational options without increasing the number of conversations. That is, we increase churning.

The wheel-star family of graphs limits CPP but within that limit achieves low CPL, low diameter, high conversational churning, and no information bottlenecks (the only complete wheel-star with a bottleneck is the star, which is very slow but the only graph with $CPP = 1$). Thus, the intuition we developed from the $TTF - NCTF$ data remains intact. The advantages of the complete wheel-star structure are related to discouraging repeat conversations with the same partner, as described. This advantage dissipates as densities increase. As we continue to add arcs we reach a point (roughly, when half the agents are in the wheel-star core) when CPP is already close to its maximum (5 in a 10-agent graph) and beyond that point adding an arc to a complete wheel-star does not dramatically increase CPP, but does add a conversational option to somebody who already has several options. At higher levels of CPP some incomplete wheel-stars and other types of graphs are competitive, as described below. This is not surprising since the limiting wheel-star is the complete graph and at higher densities there are many "almost complete" graphs with similar performance. The intuitive story is similar to that for the $NCTF$ efficient frontier. In networks where every agent has something to contribute to the total solution, mixing conversational partners is a good thing to do, which leads to shortened path lengths and reduces information bottlenecks. When we limit CPP, we need to encourage these features without increasing the total number of conversations per period. The wheel-stars do exactly that.

7.1 Statistical tests

In the $TTF(1) - CPP$ plane the base case reveals an efficient frontier, rather than a single efficient graph. Specifically the frontier was composed of the $WS45$ (the complete graph), $WS44$, $WS39$, $WS36$, $WS33$, $WS30$, $WS23$, $WS17$, and $WS9$ (a star). For these wheel-stars to populate the frontier given our comprehensive landscape of structured and random graphs unambiguously identifies the wheel-stars as an important family of graphs. To see what other graphs might be close to efficient we constructed a piece-wise linear approximation to the frontier using these graphs, and then tested the null hypothesis that the mean vector $(CPP, TTF(1))$ for alternative graphs falls below the frontier. The only graphs for which we were unable to reject that null hypothesis at 5% level of significance were $SW9$ and $WS42$, both at the higher density end of the landscape (see table 4).

7.2 Summary of base case results

The principal base case findings, then, are that complete or almost-complete graphs minimize $TTF(1)$ and only graphs with similar characteristics are efficient in the $TTF(1) - NCTF(1)$ plane. The salient desirable characteristics are many parallel conversations per period, high churning of conversational partners throughout the organization, low CPL and diameter, and no information bottlenecks. In fact, deliberate churning of partners can outperform random matching. These features are simultaneously time and total cost minimizing, and are also desirable with high amounts of discretionary time per period for hallway-style conversations (high CPP).

At lower levels of discretionary time per period efficient structures will strive for these attributes while keeping CPP bounded. One structure that does this is the complete wheel-star structure, featuring a core of completely connected agents (all of whom talk to everybody in the organization), with all others connected only to the complete core. That is, non-core agents do not talk to each other, but do talk to all members of the core, or close to it (no non-core agent should connect to just one core member, which would create an information bottleneck). Modest departures from this design can perform similarly, provided non-core agents do not spend the majority of their time speaking to the same small set of people. As more discretionary time per period become available these distinctions become less critical, because the core grows in size, the number of non-core agents declines and they have more conversational options, and the wheel-stars converge to the complete graph. Among our broad landscape of graphs, the wheel-stars distinguished themselves as an important efficient family. Regardless of what limit we place on CPP, a wheel-star is an efficient structure.

We also observe that graphs that do not have the characteristics we have identified above perform very poorly. For instance, consider the performance of the path, binary tree, cycle, and BS(4-1-4) networks as shown in figures 3 and 4: all four perform poorly relative to the efficient frontiers. The path maximizes CPL and diameter, and pays the price for this despite high CPP. The cycle also has high CPP, but its performance suffers because each agent has a choice of only 2 possible agents to talk to, so the cycle (also the star) does not see much churning of conversational partners. The path, binary tree, and BS(4-1-4) are all trees and have significant information bottlenecks. BS(4-1-4) has a relatively low CPP, which, combined with the presence of bottlenecks and lack of diversity among conversational partners, makes it the poorest-performing network in our set.

In summary, organizational structures that minimize time to ideation will feature complete graph qualities: high CPP and high levels of conversational churning, leading to low path lengths and diameter and no information bottlenecks. Structures that minimize time when CPP is bounded will strive for these features within the constraints of that bound. The wheel-star family achieves this, and distinguishes itself from the landscape of graphs by its presence along the efficient frontier.

8 Robustness tests

All of the experiments reported above were conducted on 10-person networks, with a sparse and uniform initial distribution of knowledge, and in a context where a person can recognize a productive concept when they hear it, and so can always advance (never retreat) toward the best idea. This last point refers to the preference function an agent uses to embrace, or not, an idea. In our base case model, in all bit locations a “1” was, and was perceived to be, better than a “0” by all agents, independently of what knowledge existed in other bit locations, and the 1-vector was the best of all. This represents a unimodal preference function with a unique global maximum and no local maxima. In reality it can be the case that an idea that seems good with a given state of knowledge will in fact be recognized as poor with a more evolved state of knowledge. That is, we can embrace ideas that turn out, in the end, to be bad ones, or reject ideas that would have turned out to be good ones. The efficacy of a given piece of information can change depending on what other bits of information are already in place. This would represent a “complex landscape” with interdependencies among bits. Reality may also feature local maxima at which ideation can get stuck, far from the best idea. Here we report on the robustness of our conclusions to changes in network size, initial knowledge distribution, and preference landscape complexity.

Because of the extensive computational burden of performing these tests over all 108 networks, we selected 14 representative networks for these robustness tests. The 14 networks selected, and the reasons for selecting them, are as follows (again, appendix A contains a description of each graph). We include the complete graph which was uniquely efficient in the $TTF-NCTF$ plane; the star which minimizes CPP, the intermedidate complete and near-complete wheel-stars ($WS17$, $WS24$, $WS30$ and $WS33$) which populate or are very close to the efficient frontier in the $TTF-CPP$ planes. We included a $PA5$, $SW5$ and $ER5$ graph to include one representative of each family of random graphs. We included a cycle and binary tree because these are common in the literature. Finally, we included a $PA1$, $BS(4-1-4)$ and $Spider$ graph because these three networks were among the most inefficient networks in our initial experiments, and including them allows us to get an idea of the landscape of feasible solution values in the robustness experiments.

For perspective, the position of these graphs in the $TTF-NCTF$ and $TTF-CPP$ planes in our original (base case) experiments are shown in Figure 6. As described in the next section, we also tested the robustness of our result on the complete set of all possible graphs that can be constructed with 5 agents.

Figure 6 near here.

8.1 Robustness with respect to network size

We repeated the construction of the efficient frontiers in the $TTF-NCTF$ and $TTF-CPP$ planes using our robustness test networks with 8 and 12 agents to augment our base case of networks with 10 agents. For some of our test network structures (cycle, complete graph, star, spider, binary tree) the corresponding networks with 8 or 12 nodes are unambiguous. For others, we had to define the corresponding networks. For example, with 10 agents the complete wheel-star with two core agents is a $WS17$, whereas with 8 agents this is a $WS13$ graph. See appendix A for details.

Figures 7 and 8 near here.

The graphs for the 8-node networks are shown in Figure 7, while those for the 12-node networks are in Figure 8. The complete graph continues to minimize ideation time and $NCTF$, and the complete wheel-stars continue to populate the efficient frontier in the $TTF - CPP$ plane. While larger networks become computationally intractable, these experiments provide evidence that our results hold for at least modest variations in network size around the base case. Networks with 5 nodes admit only 21 different possible graphs (in contrast with networks with 10 nodes for which there are more than 11.7 million graphs, c.f. Sloane 1973). So, with 5 agents we could tractably generate and test *all* possible graphs and not just a representative sample. The results, shown in Figure 9, are also consistent with our base case findings. The linear alignment in that figure is due to the fact that with a moderate number of arcs many 5-agent graphs have the same CPP (and recall $NCTF$ is linear in TTF for constant CPP).

Figure 9 near here.

8.2 Robustness with respect to initial knowledge distribution

In our base case each knowledge bit was held (at time 0) by only one agent, and no agent started with more than one knowledge bit. This represents an organization far from the synthesis required for a great idea, and with a relatively symmetrical distribution of relevant information within the organization. We tested the robustness of our findings with respect to more asymmetrical and uneven initial knowledge endowments as follows. For a fixed $p \in (0, 1)$, we set each (time = 0) bit in each agent's belief string be 1 with probability p and 0 with probability $(1 - p)$, independent of all other bits and agent belief strings. In each trial, a new set of belief strings was generated, and the simulation allowed to run as before. If the belief strings were such that for some bit position no agent had a 1, then the belief string was re-generated using the same process until for every bit position, at least one agent in the network had a 1. We tested two values of p, $p = 1/3$ and $p = 2/3$. $p = 1/3$ means that the expected starting knowledge endowment for an agent would feature 3 bits equal to 1, and $p = 2/3$ means this expected value is 6 bits (out of 9 total, so 6 bits is well advanced in terms of accumulated knowledge). Statisically the number of bits per agent is binomially distributed, and with $p = 1/3$ there is a 33% chance that somebody in the network begins with at least 5 bits equal to 1, halfway to the great idea, at time 0. When $p = 2/3$ there is a 79% chance that at least one agent begins with 7 bits equal 1, and a 23% chance of somebody beginning with 8 out of the 9 bits already in place. These initial endowments start the exercise much closer to completion than our base case, and also inject significant asymmetry into the knowledge distribution among individuals in the organization. The results of these experiments are shown in figures 10 and 11.

Figures 10 and 11 near here.

The complete wheel-stars continue to define the efficient frontier in the $TTF-CPP$ plane and the complete graph continues to be the time minimizing structure, as before. However, Figure 10 shows something new in the $TTF - NCTF$ plane. There, the complete graph is no longer time *and* cost minimizing. That is, at lower levels of allowable $NCTF$, other members of the complete wheel-star family may be better. Comparing figures 6, 10 and 11 reveals that the lower envelope (efficient frontier) of graphs slants upward (the complete graph is uniquely efficient) when knowledge is sparse and evenly distributed (the very front end of the ideation process). When there is more, and more asymmetrically distributed, knowledge ($p = 1/3$) the lower envelope is relatively flat, so the complete graph is joined by the complete wheel-stars as being efficient. With even higher levels of starting knowledge and asymmetries ($p = 2/3$), the lower envelope slants downward, so the complete wheel-stars define the frontier, ranging from the star to the complete graph for different levels of allowable cost.

This result can be understood by considering two phases of the ideation stage, exchanging enough information to put somebody "close" to a great idea, and then "closing the deal" by adding the last few missing bits of information to that person or persons. The complete graph is better at putting somebody close to completion, but other wheel-stars economize on conversations once such a person exists. As an extreme example, consider the star graph. The center will be the first to learn, and the center will be included in all conversations and no alternative conversations are allowed. This prevents conversations among people who may gain individual knowledge, but will not be the first to the great idea and hence will not matter much for $TTF(1)$. Hence, in a star no conversations are "wasted" en route to $TTF(1)$ and the star can be better, although much slower, than the complete graph for extremely low levels of allowed cost. We note that this logic does not extend to $TTF(all)$. In contrast to most of our results for which $TTF(1)$ and $TTF(all)$ behave similarly, in this robustness test the complete graph remains uniquely efficient in the $TTF(all) - NCTF(all)$ plane for all initial knowledge distributions.

These robustness tests suggest that the relatively rapid and democratic knowledge generated by the complete graph makes it the best configuration when speed dominates cost as a consideration, and early in the ideation process when there is a lot to learn by everybody. But the closer one gets to an agent "near" the solution and the more one must restrict costs, more structured networks that strive for conversational churning, no information bottlenecks, and low path lengths while keeping conversations per period bounded can emerge as efficient in that they can be slower, but less costly, in generating a great idea. The complete wheel-stars have these features.

8.3 Robustness with respect to preference landscape

Here we test whether our conclusions remain valid if agents interact in a complex preference landscape, meaning there can be local maxima and interdependence among the information bits in an agent's knowledge string. In a complex landscape, agents may embrace an idea that seems productive at the time but in fact is counterproductive relative to the best available idea. Also, due to interdependencies among the bits of information in their knowledge string, agents may embrace an idea with one particular endowment of knowledge in other areas, but reject that same idea with a different endowment. Since endowments can change over time, so can the relative preference an agent has for any individual bit of information. We implement this robustness test by using preference functions generated by the Nk model of Kauffman and Levin (1987), which is described in apppendix E (more details appear in Kauffman and Levin 1987, and Evans and Steinsaltz 2002). We use the latter's specification of the Nk models so we can appeal to their analytical results. In this model, the parameter k controls the complexity of the preference function: $k = 0$ is equivalent to our base case of unimodal functions where all local improvements lead to the global optimum, while higher k translates to more complex functions characterized by several local optima so that an agent changing bits by following local improvements may never get to the global optimum. As described in appendix E, we tested three values of k: 0, 1 and 3.

In a complex landscape agents can adopt beliefs in a way that traps them at a local maximum, so the globally best idea may never be found. Therefore, we cannot use TTF or $NCTF$ as performance metrics because some trajectories will never reach “finish.” Instead, we measure performance by evaluating the endowment of the most knowledgeable agent at pre-specified times. Specifically, for any landscape we first find the globally optimal preference value in the landscape (the preference value of the best idea available to the group), which we call F^*, by exhaustively searching all $2^9 = 512$ possibilities. Then, after assignment of the initial knowledge bits we compute $F(0)$, the highest preference value in the population at time $t = 0$. We then perform the experiment and record the fractional progress up to several pre-specified time points t, where $F(t)$ below is the highest preference value in the population at time t:

$$P(t) = \frac{F(t) - F(0)}{F^* - F(0)}.$$

For each of our 14 test networks and 3 values of k, we ran 1000 simulation trials and recorded the $P(t)$ values. A new random preference function was generated for each trial, and the initial bit endowment of each agent was also randomly chosen with each bit in each belief string set to 1 with probability 0.5, independently of all others. This injects considerable randomness into the initial endowment and the symmetry of that endowment, and considerable ruggedness into the preference landscape. We then plotted the results for varying levels of CPP (recall $NCTF$ is not defined with rugged landscapes).

Figure 12 near here.

Figure 12 shows the results for $t = 25$ and $k = 0$ (our smooth base-case landscape). Since better performance now translates to higher % progress (as opposed to lower TTF or $NCTF$), the efficient frontier is now the lower right portion of the chart. The scatter plot shows the efficient frontier is composed of the complete wheel-stars, as expected.

Figure 13 near here.

The scatter plots for $k = 1$ and $k = 3$ are shown in Figure 13. The figures confirm our previous results that the complete graph is the best when judged on speed alone, and the complete wheel-stars are on the efficient cost-time frontier.

8.4 Summary for dyadic conversations

The existing literature on innovation and social networks suggests that efficient structures will be sparse or small-world (clusters connected by weak bridges), may include agents filling structural holes, and may also feature some central agent or core group in which knowledge accumulates. There is also evidence that short path lengths are good for the rapid dissemination of ideas. We confirm the advantages of short path lengths, but otherwise generate a different intuition. Agents filling structural holes may enjoy individual benefits, but they are organizationally undesirable. Such agents act as information bottlenecks, and slow the ideation process. They are only recommended when severe limitations on conversations per period (CPP) force a star structure (the only graph with $CPP = 1$).

The efficacy of a highly interactive core, first suggested by Levitt's (1951) experiments, is partially supported with some contingencies. A core is one way to keep path lengths low, but not the only way. Also, although the efficient wheel-star family features a core, the more costs can be tolerated the bigger the core becomes, eventually converging to the complete graph which is all core (that is, no distinguishable core). So, an identifiable core actually slows the ideation process but might be efficient in an organization that will accept longer ideation times to keep costs per period low.

We also recommend a different lens on social network analyses of organizations, one that explicitly considers the dynamic nature of interactions within the organization. When one explicitly considers human capacity (people cannot hold more than one conversation at a time, and each conversation takes time) an agent holding one conversation per period but changing partners frequently incurs no more cost than holding one conversation per period with the same partner. However, we have shown that the former is significantly more efficient. Having more parallel conversations per period, indicated by a graph's CPP value, is good if those conversations are among constantly changing partners. Otherwise, high CPP can incur cost with little benefit. The complete graph, which maximizes both CPP and conversational churning throughout the organization, is the best for rapid ideation. At lower levels of allowable CPP or total cost, different members of the complete wheel-star family of graphs (which range from a star to the complete graph) are efficient. These graphs maximize conversational churning while keeping cost per period under control, and feature short path lengths and minimal bottlenecks.

The complete graph and complete wheel-stars, which emerge so prominently in our study, feature the following ideation-friendly characteristics: high conversational churning, short path lengths, and minimal information bottlenecks for any allowable level of CPP. Other graphs that achieve these ends can also be competitive. In particular, there are many graphs at the high density end that are "almost complete" in that they are missing just a few arcs and they behave very similarly to the complete graph. These results are robust to variations in initial knowledge endowment, the complexity of the preference landscape, and to modest variations in network size.

Thus, efficient innovative organizations look nothing like what one intuitively associates with standard formal organizations with strict and unchanging lines of communication; nor do they conform with what one might expect from static social network representations of communication patterns. From a sociometry perspective, the graph-specific parameter CPP should join the list of features we routinely compute for a network, to augment "number of arcs" as a proxy for the network's cost in personnel time. Also, sociometric surveys related to ideation should not look at the structure of stable conversational partners, but rather look specifically for churning of partners among a wide set of people in the organization, along with short path lengths and no information bottlenecks. The development of the appropriate metrics that balance theory with practical data collection, and testing those metrics in the lab or field, is the subject of future work.

The complete wheel-stars heuristically resemble the core and extended team structure familiar in new product development efforts. In these a highly interactive core team connects to a changing cast of extended team members as the development process progresses through its stages. The extended team may be dominated by market research during the concept generation phase, then technical R&D for development, then operations, sales and other standard business functions for launch. The differences with our model are important however. We model the ideation process upstream in the fuzzy front end of an innovative effort, where people have very weak priors on what knowledge will be useful, and hence engage in random rather than targeted conversations. The downstream stages of a new product development effort will likely feature more prescribed topics of conversation. Our model of dyadic interactions is more reflective of serendipitous "water cooler" conversations than what one typically thinks of as a team meeting. Overall, our model is intuitively better suited to inform the informal patterns of communication in an organization rather than formal teams or project structures.

Still, the complete wheel-stars feature a highly interactive core group that heuristically suggests that investigating some form of group meeting might be useful. In the following section we introduce group meetings in the form of several sequential one-to-many communications, in which one person talks to many listeners simultaneously. From existing results in communication theory (c.f. Kaufer and Carley 1993) it is not clear ex ante whether this will help or hurt ideation.

9 Group meetings

Dyadic, or bilateral, conversations and group meetings are different forms of communication, and we have to be careful to define what we mean by each so that comparisons can be meaningful. In bilateral conversations there are always two people, so the cost of a conversation in "person-time" is just a constant scaling of the cost in chronological time, justifying the use of $NCTF$ and CPP as cost parameters. However, when varying numbers of people can be involved in exchanges, we need to compute costs in person time. We do this as follows. In a single bilateral conversation two people converse on a topic, each understanding the other's perspective and incorporating that perspective if they feel it improves their knowledge. To do this we must have each person explaining their perspective, and potentially answering some clarifying questions from the other. We assume that it takes 1 time unit for this process of communicating a perspective from one person to another and for the other to comprehend it fully, so 2 people invest 1 time unit each in that activity. The other person then offers their opinion on the topic, which is comprehended and potentially incorporated by the other, taking another unit of chronological time. So, a bilateral conversation will involve two people and take 2 time units, costing 4 “person time units” (PTU’s), and as a result of this investment either party may (or may not) update their perspective. If the organization engaged in only bilateral conversations, the organizational cost per period would be $(4 \times CPP)$ PTU’s.

Now, however, we will also allow one-to-many broadcasts of information, where one person talks to many. We assume that the speaker states their position on a topic, and may admit up to one clarifying question which we assume will suffice for all, taking a total of 1 time unit (the same as a one-way communication in a bilateral conversation). The speaker does not yield the podium to other speakers to state their cases. So, in 1 time unit a single speaker can make their position known simultaneously to all people in attendance, who comprehend, consider and potentially embrace the content of that communication, but the speaker will not learn from the audience. If there are n people involved (including the speaker) the organizational cost for this single broadcast is n PTUs.

The advantage of one-to-many broadcasts over face-to-face bilateral communication is that one single communication can impact many listeners simultaneously. The advantages of face-to-face communication over one-to-many broadcasts is that many bilateral conversations can take place in parallel, and in each either party can learn from the other. By comparing these two modes of communication we are, essentially, comparing the relative merits of these features. In actual practice, of course, conversations and one-to-many communications can take many forms, forming a complex continuum of possibilities. Bilateral communication can, depending on personality types, be more like one-sided broadcasts and speakers to a group may engage a single audience member in spirited debate, excluding the rest of the audience and reducing the interaction to more of a bilateral form. Different forms of one-to-many broadcasts (e.g. email) have different characteristics that might affect the dynamics, and the probability of attention and/or accurate comprehension. We chose our two archetypes for comparison because they capture some significant differences between bilateral and one-to-many communication, and because the relative merits of these differences for ideation are not clear ex ante.

We make the further assumption that a “group meeting” involves a series of people speaking in turn, taking one time period to broadcast his or her opinion on a randomly chosen topic to the group, with the listeners comprehending and considering that opinion for incorporation into their personal beliefs. The speaker does not update his or her opinion on that specific topic at that time, but might later if he or she is a listener while somebody else in the meeting holds forth. This, likewise, is a pure form of dynamics that in practice can take on practically limitless levels of complexity.

We test the efficacy of one-to-many broadcasts versus parallel bilateral communications by assuming that for the last τ out of every T time periods the agents gather for group meetings, and in between meetings they converse bilaterally as before, matched through a particular social network. In the meeting, at each of the τ time periods, a randomly chosen agent broadcasts his or her opinion on a randomly chosen topic to the group. All people in the group simultaneously comprehend the content of the broadcast and consider incorporating it into their beliefs. We call such a situation a "$\{$Social network $\}$ T/τ" system where we insert the name of the social network in brackets. For example, in a "Binary tree 100/10" system the agents spend 90 time periods out of 100 communicating through a binary tree as usual, and then the last 10 time periods out of every 100 in meetings as described. In our experiments, we test $\tau = 10$ with $T = 20, 50, 100$.

To account for the different organizational costs of these two forms of communication, we report $TUF(1)$ (Time-Units to Finish, for at least one agent to be fully learned) instead of $TTF(1)$, where we count a period of bilateral conversations as 2 time units (because 2 agents in turn are speaking in each conversation), and a group meeting of length τ as τ time units (because a single agent broadcasts 1 bit in 1 time unit). We report $PTUF$ (Person Time-Units to Finish) instead of $NCTF$, where for one time period in a meeting with n people we charge the organization n PTUs, and when the organization is engaged in k bilateral conversations we charge $4k$ PTUs. For example, a meeting with 10 agents that meets for 1 time period = 10 PTUs and allows one person to broadcast their opinion. Those same ten agents communicating through a social network that results in k bilateral conversations will incur an organizational cost of $4k$ PTUs. We also define and report $BAPP$ (Busy Agents Per Period) instead of CPP, where $BAPP$ is equal to $PTUF$ divided by TUF and indicates the average number of agents engaged in some form of communication per time period. If there were no group meetings, $BAPP$ would equal $2CPP$.

We collect data for our 14 robustness test networks and also collect data for a “10/10” scenario (i.e. only group meetings) for comparison. Note that since there are no bilateral conversations in a “10/10” structure, the name of the network is not necessary. We test the efficacy of group meetings in both smooth and complex landscapes.

9.1 Results for group meetings with a unimodal preference function

$TUF(1)$, the chronological time for the first agent to arrive at the best idea, for our 14 robustness test networks and for each meeting scenario are shown in Figure 14. In this picture we also show the expected $TUF(1)$ for the $10/10$ structure (only group meetings) which equals 164.6 (derived in appendix F).

Figure 14 near here.

Clearly, “good” organizational structures can be hurt by group meetings, but “bad” ones can be helped. In particular, for the networks to the left on the horizontal axis which have a “no-meetings” $TUF(1) < 164.6$, performance is not helped (statistically not differentiated at 5%) as group meetings are added within the test range, but they (statistically significantly) outperform $TUF(1) = 164.6$ so we would expect to see significant degradation in performance as meetings increase in frequency toward the “only meetings” scenario. This was confirmed with test runs for WS30, using increasing meeting frequencies in a range from 20/10 to 21/20 (appendix F). In summary, for these graphs meetings don’t help at any frequency, and can hurt at high frequency.

In contrast, for networks with “no-meetings” $TUF(1) >$ 164.6 the addition of group meetings (statistically significantly) decreases TUF(1) towards the limiting value of 164.6 as meeting frequency increases. For example, for the broom-star $BS(4-1-4)$ network the benefit of meetings is very dramatic, going from $TUF(1) = 764$ with no meetings to 195 with 20/10 meetings. Of course, the broom-star is a poorly performing network that one hopes is never operational in an organization. But, if it is, group meetings can help significantly.

There is a key structural difference between the networks for which group meetings significantly improves performance and those for which they do not. The last 5 networks in Figure 14 (rightmost on the horizontal axis) all have an information bottleneck, an individual filling a "structural hole" who, if removed, disconnects the graph into two disjoint sets of people. In contrast, the first 6 networks in Figure 14 (where group meetings cannot help but can hurt performance) have significant redudancy in information channels, in that one would have to remove 3 or more agents to disconnect the graph. We caution that this finding is limited to our robustness test set of 14 networks, and further study on the interdependency of group meetings and information bottlenecks on ideation time is warranted. However, the intuition has face validity. Information bottlenecks can retard ideation, which is dependent on the synthesis of diverse thought, and group meetings can short-circuit the information bottlenecks. Structures that naturally disallow bottlenecks do not need group meetings in the ideation phase.

Figure 15 near here.

Figure 15 shows the average number of busy agents per period ($BAPP$) as a function of underlying network and group meetings. As expected, $BAPP$ increases with meeting frequency because everybody is busy in a meeting, whereas some agents may not be in conversation each time period when matching through a social network. With 10 agents in a T/τ regime of meetings, the expected $BAPP$ can be approximated as $\frac{\tau}{T} \times 10 + (1 - \frac{\tau}{T}) \times 2CPP$. In particular, a complete graph has the same $BAPP$ with or without meetings and graphs like a star with low CPP will see significant increases in $BAPP$ as meetings become more frequent.

Figures 16 and 17 near here.

Can meetings be efficient if we restrict costs? In the $TUF(1) - PTUF(1)$ plane (figure 16), the complete graph with no meetings is uniquely efficient over all tested graph types and meeting frequencies. Figure 17 shows the efficient frontier in the $TUF - BAPP$ plane for different group meeting frequencies. Here we see that meetings are never unambiguosly efficient, because the right choice of network structure renders meetings less valuable. So, while ideation time can be reduced by meetings if the out-of-meeting structure is poor, the right structure renders meetings unecessary. Also, under any given frequency of meetings, the efficient frontier consists only of the complete wheel-stars. This confirms that the “right structure” can always be found in the complete wheel-star family.

9.2 Results with group meetings with a complex preference landscape

In this section we investigate the effect of group meetings with underlying networks on a complex preference landscape with multiple local maxima. We used the Nk model previously described and our 14 robustness test networks, and conducted a 3×3 full factorial design experiment by varying k among $k = 0, 1$ and 3 with N fixed at 9, and testing meeting frequencies of 20/10, 50/10 and no meetings. As before, in complex landscapes the process may never reach the best idea, so we measure performance by % progress toward the optimum, and we report the results at time equal to 50.

Figure 18 shows that complexity hurts, in that for any human resource investment level the % progress up to a given time will be lower as the complexity of the landscape increases. This is to be expected, given that complexity can lead to false starts, course reversals, and dead ends as the ideation process unfolds.

Figure 18 near here.

Figure 19 shows the impact of group meetings on performance with $k = 3$, which is representative. The results confirm that meetings help poorly performing networks, but do not help well-performing networks, as before. In fact, here the results are even more stark, because meetings can strictly hurt well-performing networks. For example, a complete graph declines from $P(50) = 0.8721$ with no meetings to $P(50) = 0.8498$ with 20/10 meetings; a statistically significant difference with a one-tailed p-value of 0.0160. The efficient frontiers with group meetings and complex preference landscapes are shown in figures 18 (different levels of complexity) and 20 (different meeting frequencies); again, we find that the efficient frontier is comprised of the wheel-stars. An interesting finding in figure 20 is that when the entire space of network types and meeting frequencies is considered, different networks with different meeting frequencies may be efficient. For example, a star is very slow but also very low cost. It is the only graph with $PTU \leq 200$ in

50 time periods, and so is on the efficient frontier. However, a star is helped by team meetings, so some combination of the star and team meetings can be efficient at some levels of allowable cost. For example, Star 20/10 is on the frontier for $PTUF \leq 325$. With more time for conversation other graphs, which benefit less from meetings, become possible. In particular, if one wanted to maximize performance with no constraint on $PTUF$, one would choose a complete graph with no meetings.

Figures 19 and 20 near here.

10 Conclusions

What form of communication pattern is best for the ideation process by which an organization generates great ideas to feed into the formal project pipeline? We investigate this question with a model of ideation as the mixing and matching of different bits of information communicated through a social network, and assembled in people's minds into new syntheses. A "great idea" is generated when the pieces have been put together in a particularly compelling way by somebody in the organization. What to recombine with what cannot be obvious ex ante, for otherwise the new synthesis would be neither distinctive nor innovative. Search or consideration beyond the obvious suggests very weak pre-screens about what is worth talking about, a feature we model using randomly selected topics of conversation. The boundaries of this work include our assumption that everybody in the organization shares a common objective (although may differ on what ideas are good in pursuit of that objective) and is willing to converse freely with others. We abstract away from the cognitive and relational aspects of communications to focus on the structure of the social network. We do not explicitly include some realistic influences (political, motivational and otherwise) that complicate actions and interactions in actual organizations. We use a parsimonious model of interaction that is broadly consistent with existing models of organizational learning yet simple enough to focus on the effect of the organizational structure without being confounded by a host of other considerations.

Our results suggest that efficiently innovative organizations look nothing like what one intuitively associates with standard formal organizations with strict and unchanging lines of communication, nor do they conform with what one might expect from static social network representations of communication patterns. Instead, ideation is accelerated by a “complete graph” type network, in which people in the organization dynamically churn through a large (ideally the entire population) set of conversational partners over time, a feature that naturally begets short path lengths and diameters and eliminates information bottlenecks. In organizations with these features group meetings do not help and can hurt the process, because many parallel conversations can achieve the same or better results as one-to-many communications. However, in inefficient organizations with information bottlenecks and long path lengths, group meetings can be an effective antidote to those ills.

We explicitly consider the time invested in conversations, suggesting that “conversations per period” (CPP) join “number of arcs” as a network characteristic associated with cost. With sufficient amounts of churning higher CPP accelerates ideation, but at a cost, so we identified efficient network structures for organizations with little slack time for informal, hallway-style conversations (disallowing high CPP). The complete wheel-star family, which ranges from the star to the complete graph as CPP increases from its minimum to maximum values, figured prominantly on the efficient time-cost frontier. These results unambiguously identify the complete wheel-stars as an important family of networks for ideation, and the complete graph an important member of the family. These networks feature maximal conversational churning for any given CPP limit, short path lengths and diameters, and no information bottlenecks (after the star, which is the only graph with $CPP = 1$). Other graphs with these features can also be competitive on the efficient frontier. Again, meetings do not help an already efficient network design, but can help an inefficient one.

To translate these findings into real organizations, we need to focus on the interactions related to the far upstream ideation activities in the firm, and on what patterns to look for in those conversations. Consider two types of conversations in an organization. The first are those directly related to some work-related activity for which random conversational topics are not natural. These may be specific on-going projects or routines that require that a person interact with certain other parties on prescribed topics to move the project forward or to get their job done. It is not that these prescribed conversations cannot support innovative thinking, but more work needs to be done beyond our model, in which conversational topics are random rather than predetermined. The second type (henceforth type II) of conversation is casual hallway-style conversations with weak pre-screens regarding appropriate topics. Our research suggests that the patterns in type II conversations can affect the process by which the organization generates ideas. Specifically, for any upper bound on the time available for type-II conversations, an effective ideating network will feature high conversational churning begetting low path lengths and diameters and reducing information bottlenecks. Given these features, higher levels of hallway conversations promote faster ideation. The family of complete wheel-stars achieves these ends. These results are robust to changes in initial information endowments and the complexity of the preference landscape, and to modest changes in network size.

Some of the features desirable for ideation are not well measured using current sociometric techniques. There is a need to develop new metrics that focus on the dynamic nature of conversational pattterns (churning), to augment the more common features of short path lengths and the absence of information bottlenecks. The next step for this research should focus on developing these metrics for the ideation process and then testing them in simulation, in the lab, and/or in the field.

Acknowledgements The authors would like to acknowledge Professor Christoph Loch for stimulating interactions in the formative stages of this research effort, for the constructive suggestions of two early anonymous referees on an early version of this manuscript, and to Professor James Moody and two later anonymous referees for their conscientious efforts and helpful suggestions in developing this work.

References

[1] Ahuja, G. Collaboration networks, structural holes, and innovation: A longitudinal study. Administrative Science Quarterly 45, 2000, 425–455.

[2] Albert, R. and A.-L. Barabási. Statistical mechanics of complex networks. Reviews of Modern Physics 74, 2002, 47–97.

[3] Aldrich, H. Organizations Evolving, Sage Publications, London, 2000.

[4] Astrom, K. Optimal control of markov processes with incomplete state information. J. Math. Anal. App. 10, 1965, 174–205.

[5] Baker, W. Achieving Success Through Social Capital: Tapping Hidden Resources in Your Personal and Business Networks, Jossey-Bass, San Francisco, 2000.

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

[7] Bavelas, A. Communication patterns in task-oriented groups. Journal of Acoustical Society of America 22, 1950, 271–282.

[8] Bertsekas, D.P. Dynamic Programming and Stochastic Control. Academic Press, Orlando, Florida, 1976.

[9] Bertsimas, D. and J. Nino-Mora. Restless bandits, linear programming relaxations, and a primal-dual index heuristic. Oper. Res. 48, 2000, 80–90.

[10] Bollobás, B. Random Graphs. Cambridge 2001.

[11] Borgatti, S.P. and M.G. Everett. Models of core/periphery structures. Social Networks 21, 1999, 375–395.

[12] Brown, M.B. and A.B. Forsythe. Robust tests for equality of variances. J. American Statistical Association 69, 1974, 364–367.

[13] Burt, R. The contingent value of social capital. Administrative Science Quarterly, 1997, 339–365.

[14] Burt, R. The network structure of social capital. Research in Org. Behavior 22, 2000, 345–423.

[15] Carley, K. A Theory of Group Stability. American Sociological Review 56(3), 1991, 331-354.

[16] Cohen, J. Multiobjective Programming and Planning, Academic Press, NY, 1978.

[17] Coleman, J. Foundations of Social Theory, Belknap Press, Cambridge, MA, 1990.

[18] Erdös, P. and A. Renyi. The evolution of random graphs. Publ. Math. Inst. Hung. Acad. Sci. 5, 1960, 17–61.

[19] Evans, S.N. and D. Steinsaltz. Estimating some features of Nk preference landscapes. The Annals of Applied Probability 12(4), 2002, 1299–1321.

[20] Fleming, L. Recombinant uncertainty in technological search. Management Science 47, 2001, 117–132.

[21] Gittins, J. and D. Jones. A dymamic allocation index for the sequential design of experiments. Colloquia Mathematica Societatis Janos Bolyai, European Meeting of Statisticians, Budpest, Hungary 1972.

[22] Gittins, J. Bandit processes and dynamic allocation indices. J. Roy. Statist. Soc. 41, 1979, 148–164.

[23] Goldberg, D. Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading MA 1989. A readable reference.

[24] Grabner, P.J. and H. Prodinger. Maximum statistics of N random variables distributed by the negative binomial distribution. Combinatorics, Probability and Computing 6, 1997, 179–183.

[25] Harrison, J.M. Brownian Motion and Stochastic Flow Systems. John Wiley and Sons, New York, 1985.

[26] Kaufer, D. and K. Carley, K. Communication at a Distance. Lawrence Erlbaum, NJ, 1993.

[27] Kauffman, S. and S. Levin. Towards a general theory of adaptive walks on rugged landscapes. Journal of Theoretical Biology 128, 1987, 11–45.

[28] Kogut, B. and I. Zander. What do firms do? Coordination, identity and learning. Organization Science 7, 1996, 502–518.

[29] Kraatz, M. Learning by association? Interorganizational networks and adaptation to environmental change. Academy of Management Journal 41, 1998, 621–643.

[30] Leavitt, H. Some effects of communication patterns on group performance. Journal of Abnormal and Social Psychology 46, 1951, 38–50.

[31] Levene, H. Robust tests for equality of variances. I. Olkin (ed.), Contributions to Probability and Statistics, Stanford University Press, Palo Alto, CA, 1960.

[32] Liebeskind, J., A. Oliver, L. Zucker and M. Brewer. Social networks, learning and fleixibility: Sourcing scientific knowledge in new biotechnology firms. Organization Science 7, 1996, 428–443.

[33] Lovejoy, W.S. Some monotonicity results for partially observed Markov decision processes. Operations Research 35, 1987, 736–743.

[34] Lovejoy, W.S. A survey of algorithmic methods for partially observed Markov decision processes. Annals of OR 28, 1991, 47–66.

[35] Lovejoy, W. and C. Loch. Minimal and maximal characteristic path lengths in connected sociomatrices. Social Networks 25, 2003, 333–347.

[36] Nahapiet, J. and S. Goshal. Social capital, intellectual capital, and the organizational advantage, Acad. Mgt Review, April 1988.

[37] Nerkar, S. and A. Paruchuri. Need to add reference. Evolution of R&D capabilities: The role of knowledge networks within a firm, Management Science 51, 2005, 771–785.

[38] Newman, M. A. Barabasi and D. Watts. The Structure and Dynamics of Networks. Princeton University Press, Princeton, N.J., 2006.

[39] Powell, W., K. Koput and L. Smith-Doerr. Interorganizational collaboration and the locus of innovation: Networks of learning in biotechnology. Administrative Science Quarterly 41(1), 1996, 116–145.

[40] Puterman, M.L. Markov Decision Processes: Discrete Stochastic Dynamic Programming, John Wiley and Sons, New York, 1994.

[41] Reagans, R. and E. Zuckerman. Networks, diversity and productivity: The social capital of corporate R&D teams. Organization Science 12 (4), 2001, 502–517.

[42] Schumpeter, J. Business Cycles, McGraw-Hill, NY, 1939.

[43] Smallwood, R. and E. Sondik. Optimal control of partially observable processes over the finite horizon. Oper. Res. 21, 1973, 1071–1088.

[44] Sloane, N.J.A. A Handbook of Integer Sequences. Academic Press, New York, 1973.

[45] Sosa, M., M. Gargiulo and C. Rowles. Component connectivity, team network structure and the attention to technical interfaces in complex product development. Insead working paper 2007/68/TOM/OB, 2007.

[46] Uzzi, B. The sources and consequences of embeddedness for the economic performance of organzations: the network effect. American Sociological Review 61, 1996, 674–698.

[47] Valente, T. Network Models of the Diffusion of Innovations. Hampton Press, NJ, 1995.

[48] Wasserman, S. and K. Faust. Social Network Analysis. Cambridge University Press, Cambridge 1999.

[49] Watts, D.J. and S.H. Strogatz. Collective dynamics of ‘Small-Worlds’ networks. Nature 393(6884), 1998, 409–410.

[50] Watts, D. Small Worlds. Princeton University Press, Princeton, NJ, 1999.

[51] Weber, R. and G. Weiss. On an index policy for restless bandits. J. App. Prob 27, 1990, 637–648.

[52] Whittle, P. Optimization over Time, Vol I. John Wiley and Sons, N.Y. 1982.

[53] Whittle, P. Restless bandits: Activity allocation in a changing world. J. App. Prob 25, 1988, 287–298.

Figures

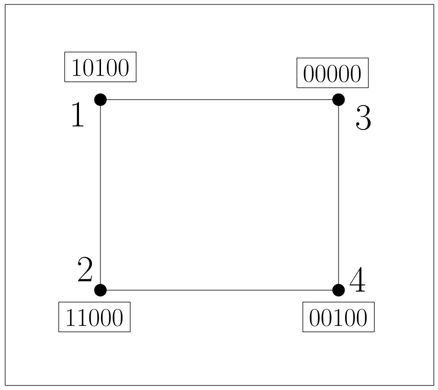


Figure 1: Four-node example network with agent beliefs.

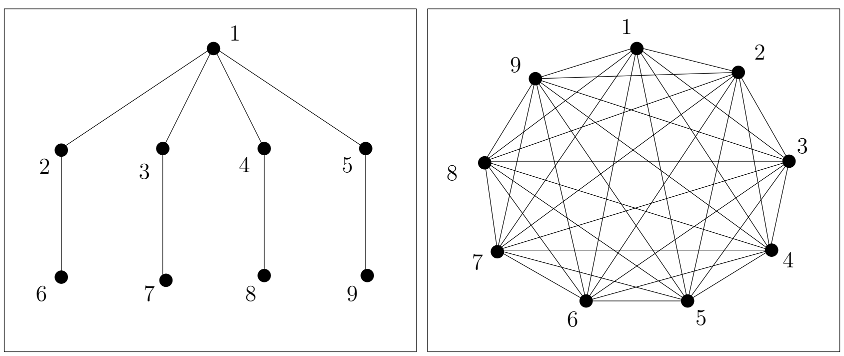


Figure 2: Nine-node networks with equal CPP values: (a) spider, (b) complete

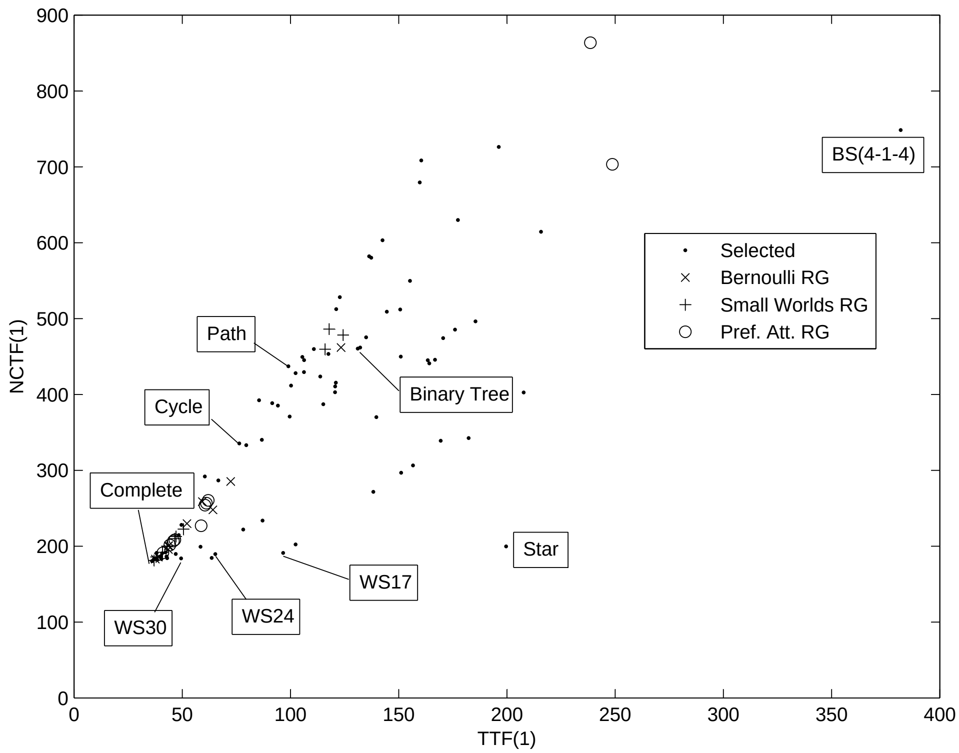


Figure 3: Simulation output: $TTF(1)$ vs. $NCTF(1)$. The complete graph minimizes both metrics, although several other dense graphs are observed to have statistically similar performance.

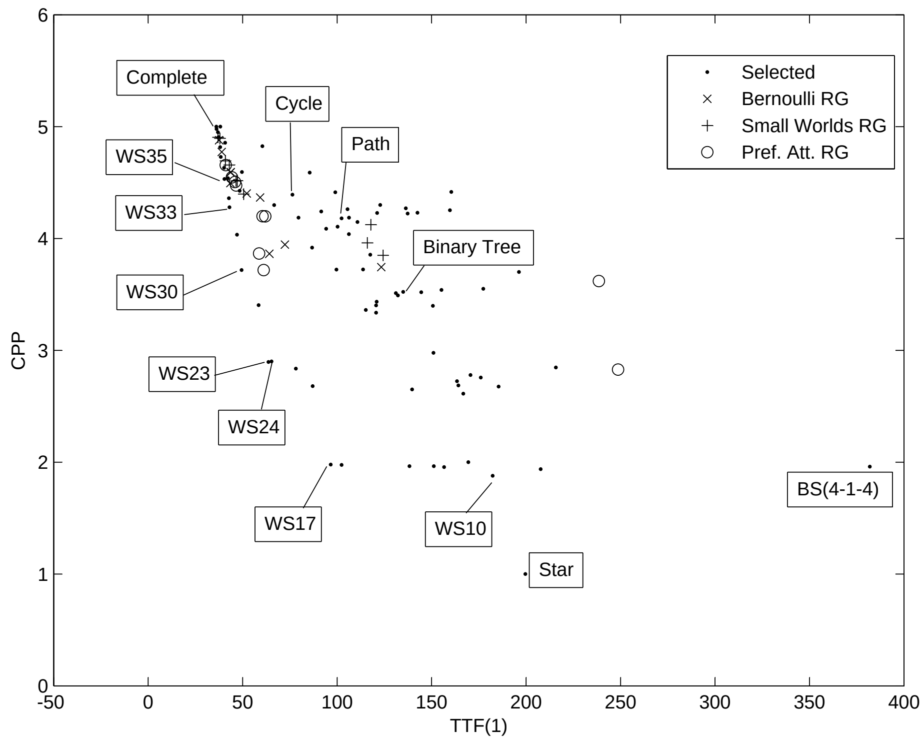


Figure 4: Simulation results: $TTF(1)$ vs. CPP.

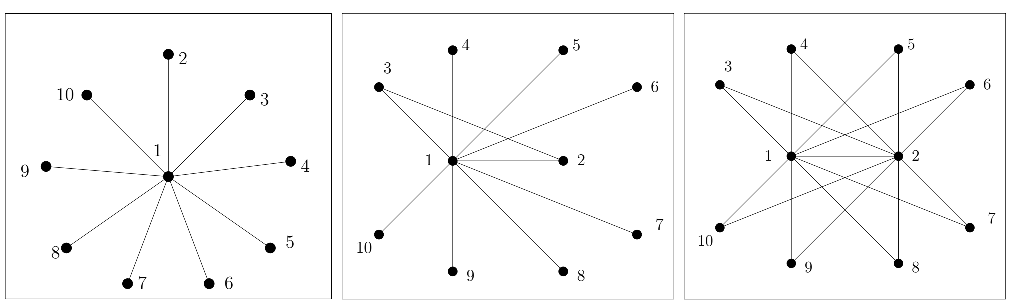


Figure 5: Wheel-star networks: 1. WS9, or the star; 2. WS10; 3. WS17.

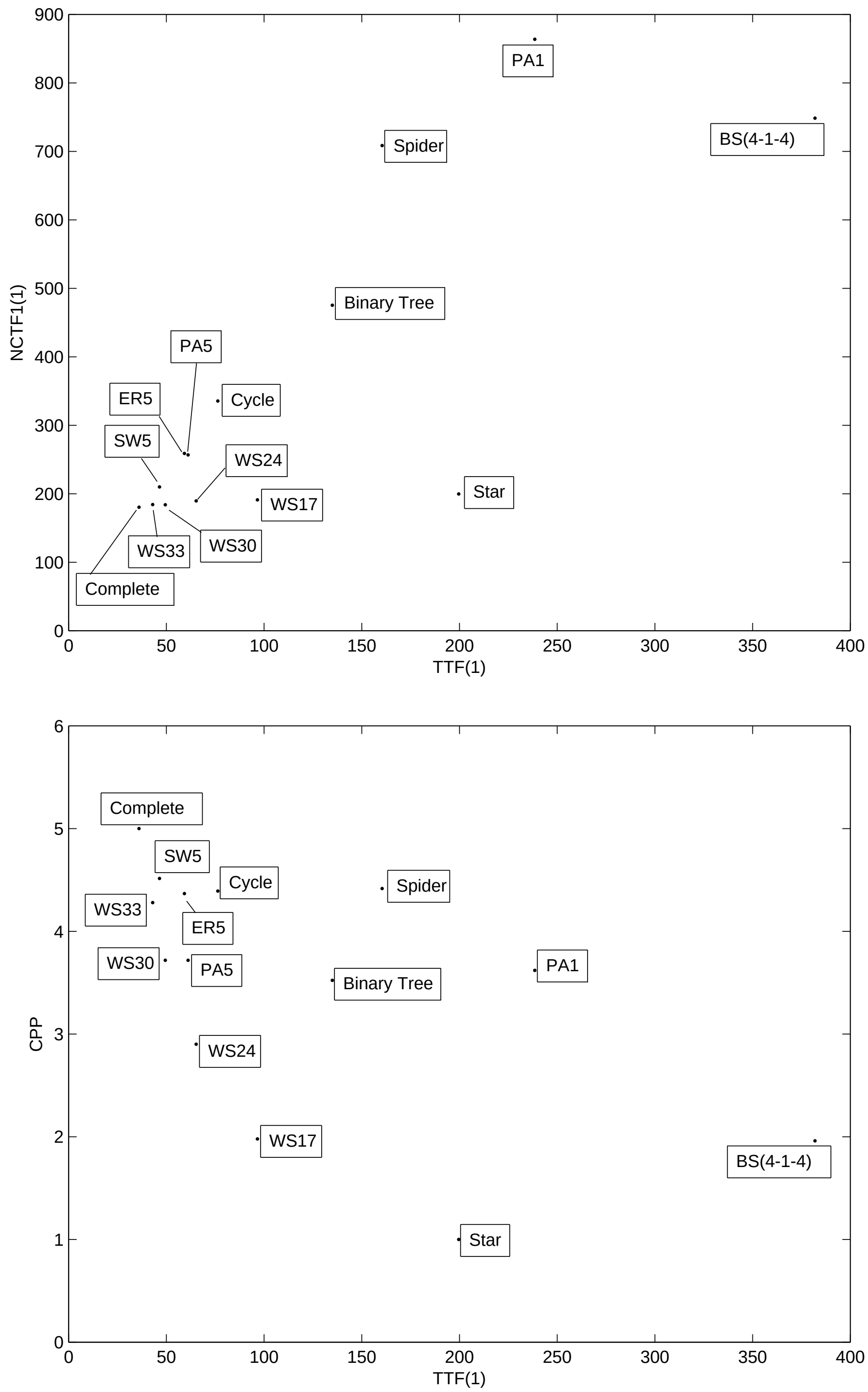


Figure 6: Performance metrics of the 14 networks chosen for robustness analysis.

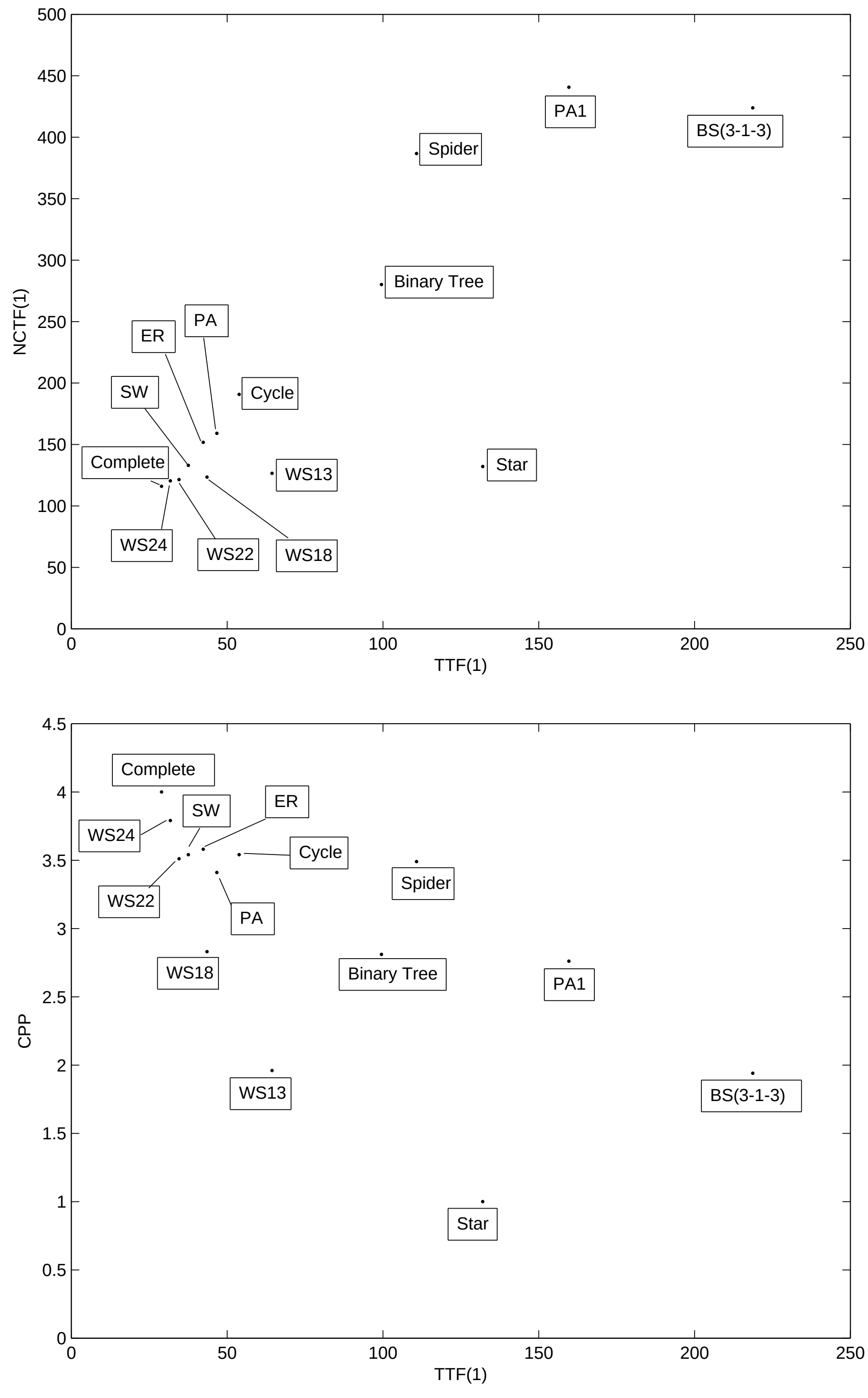


Figure 7: Performance metrics of the 8-node networks.

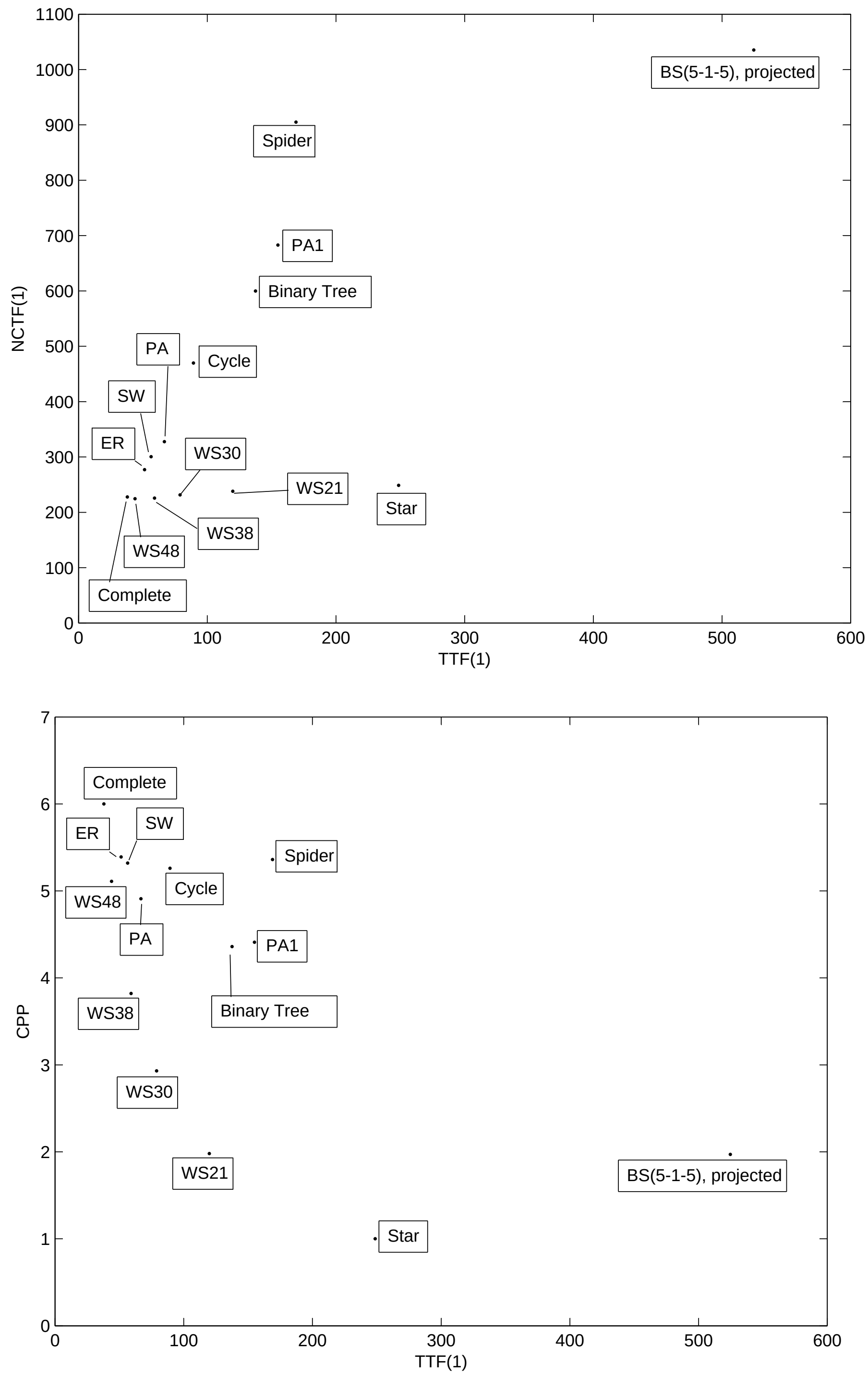


Figure 8: Performance metrics of the 12-node networks

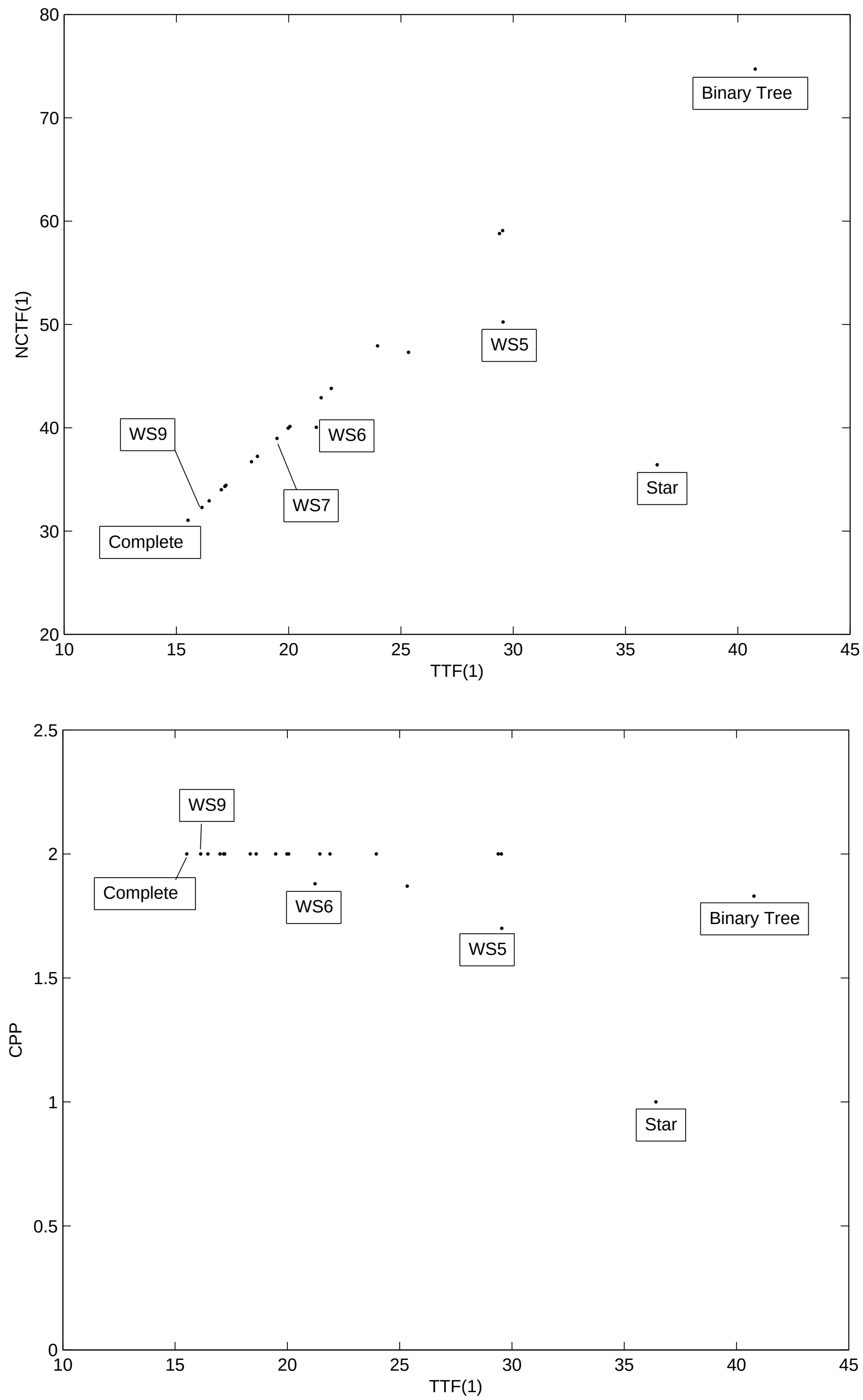


Figure 9: Performance metrics for 5-node networks.

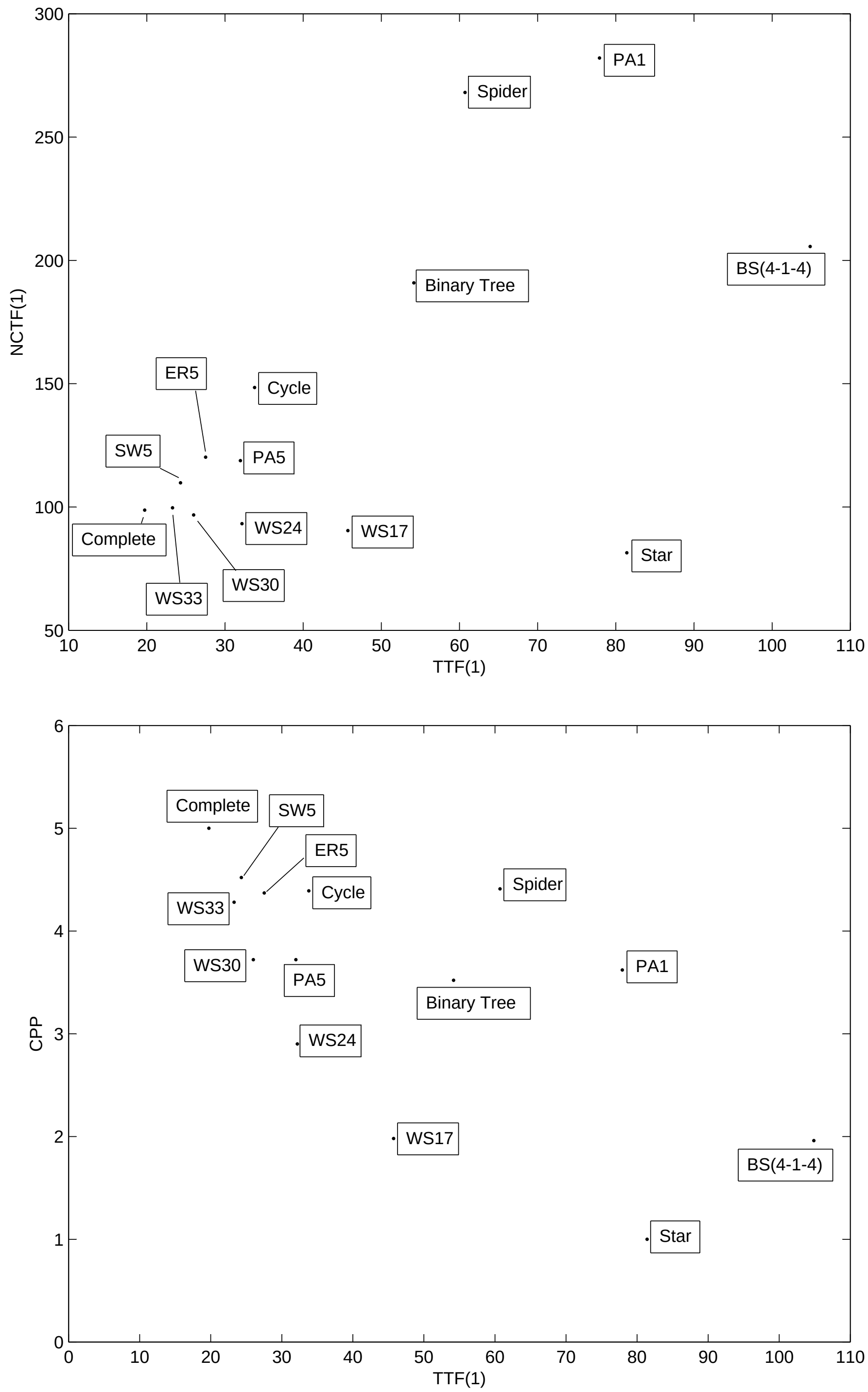


Figure 10: Performance metrics when initial knowledge distribution has each bit 1 with $p = 1/3$.

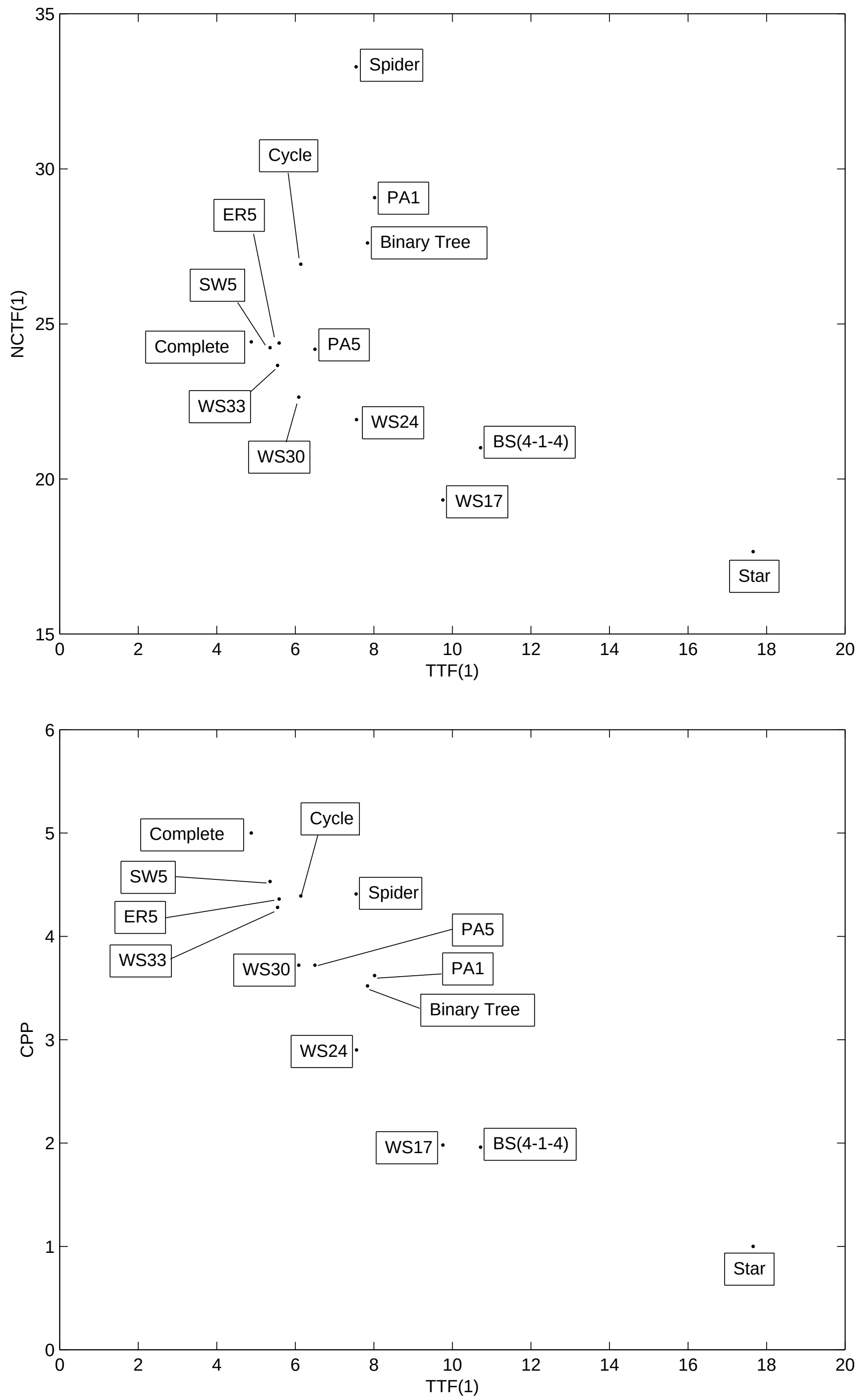


Figure 11: Performance metrics when initial knowledge distribution has each bit 1 with $p = 2/3$.

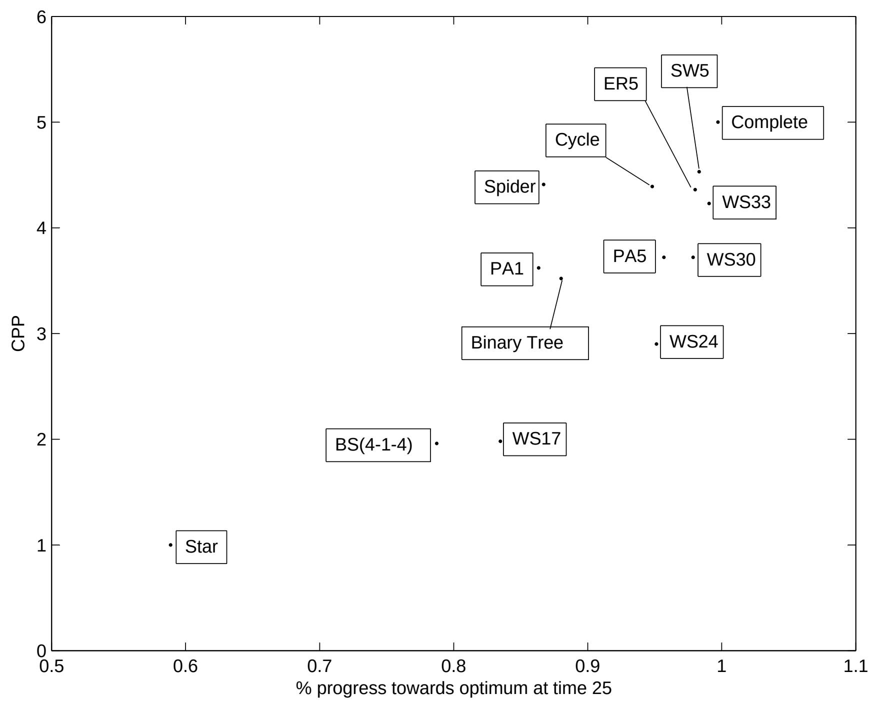
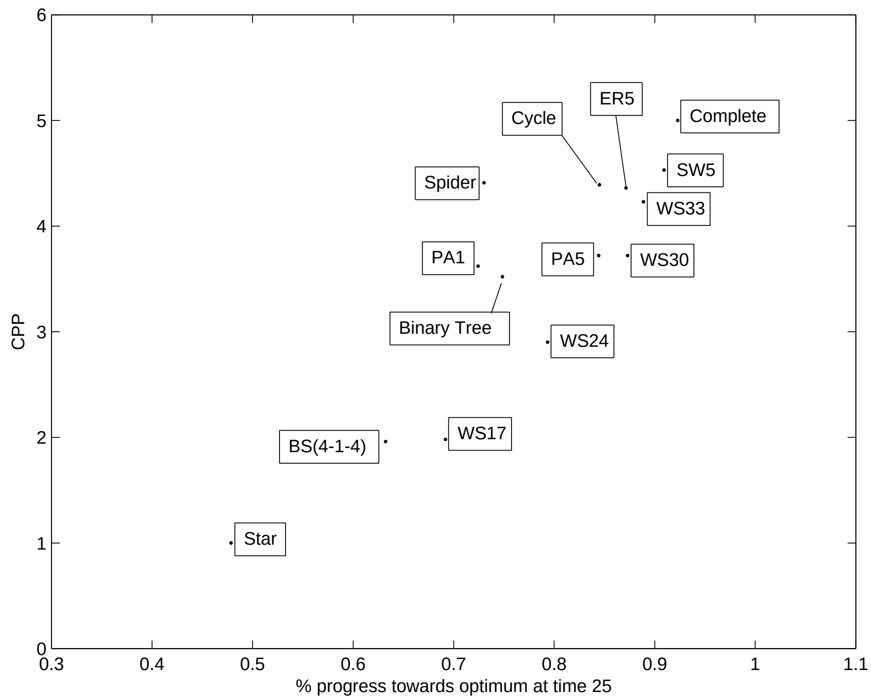


Figure 12: Performance in the Nk model with $k = 0$.



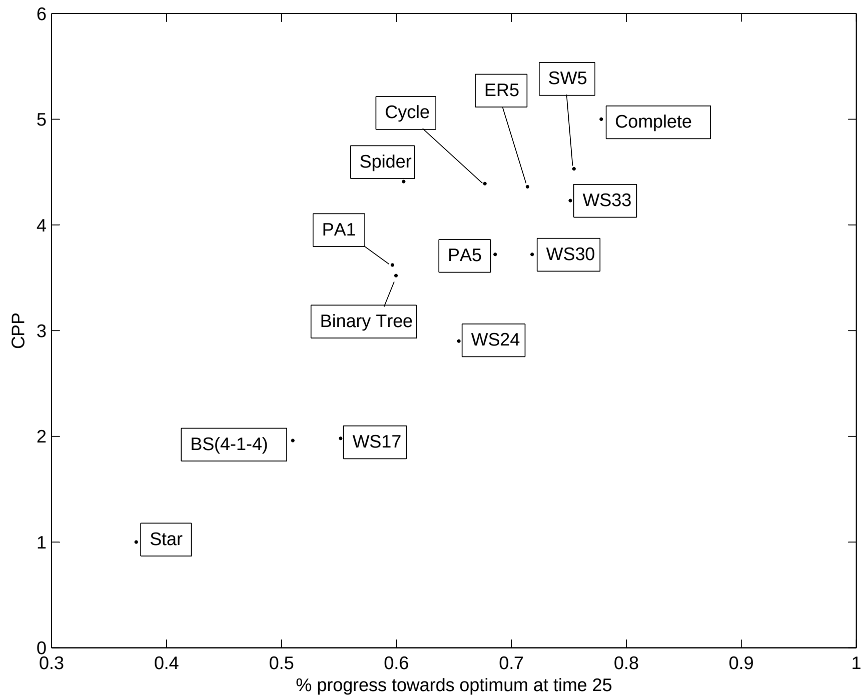


Figure 13: Performance in the Nk model with $k = 1$ and $k = 3$.

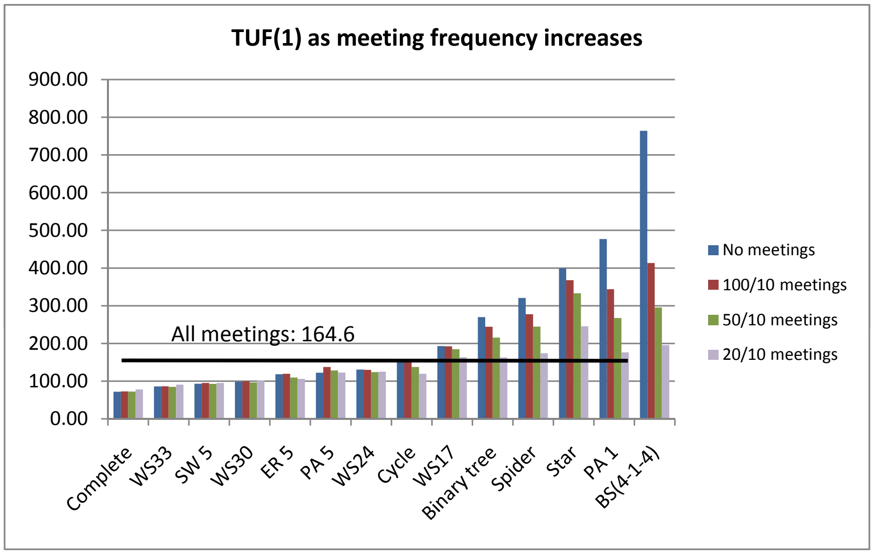


Figure 14: $TUF(1)$ with group meetings

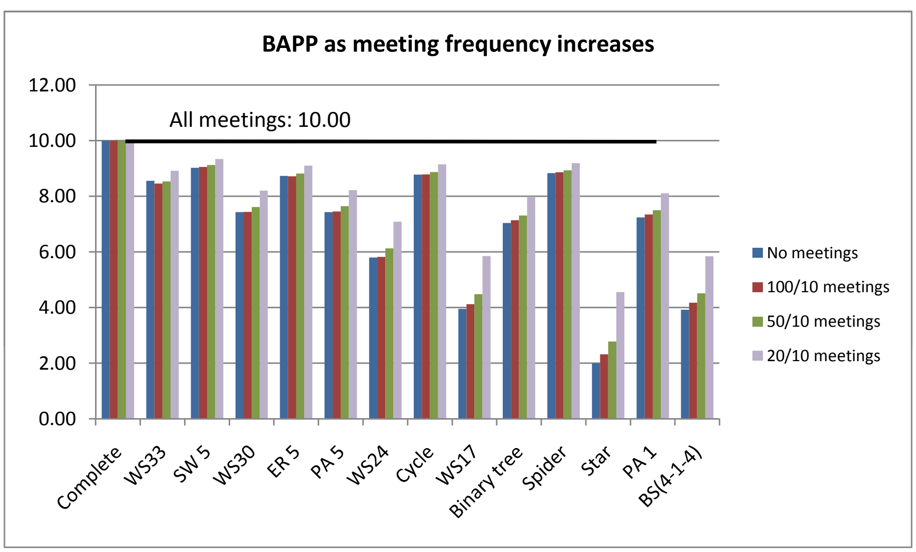


Figure 15: $BAPP$ with group meetings

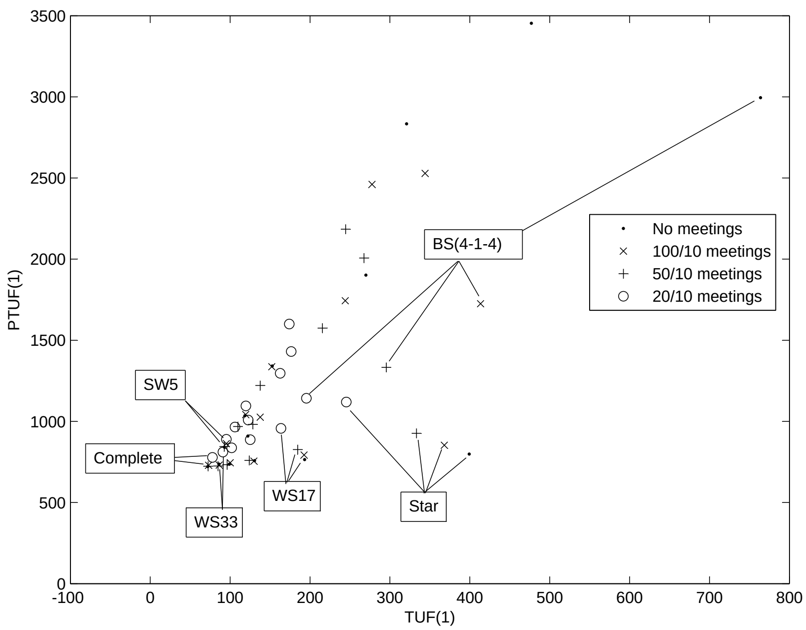


Figure 16: Efficient frontier ($TUF(1)$ vs $PTUF(1)$) with group meetings

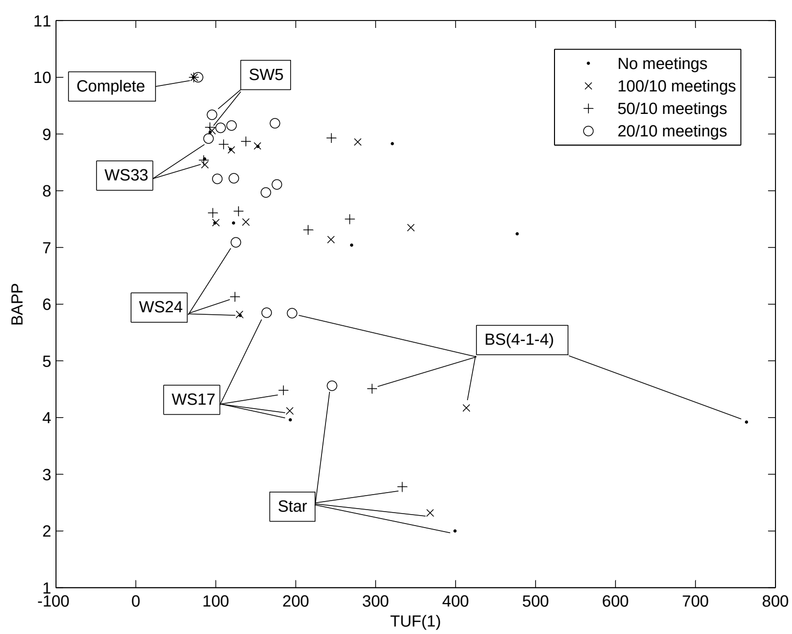


Figure 17: Efficient frontier ($TUF(1)$ vs $BAPP$) with group meetings

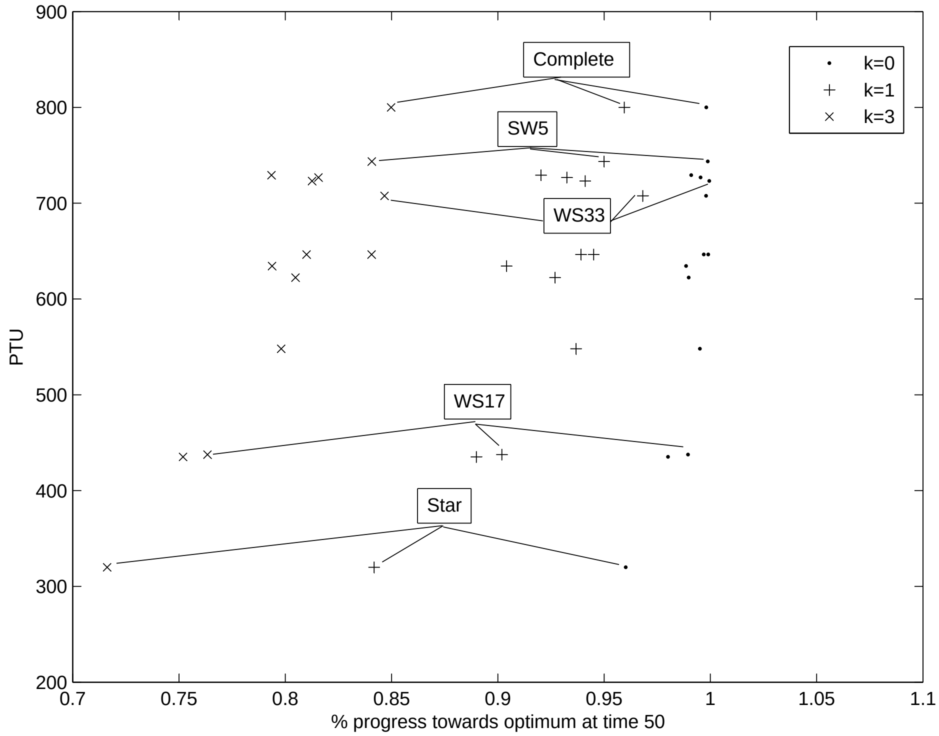


Figure 18: Efficient frontier with 20/10 meetings and changing preference complexity

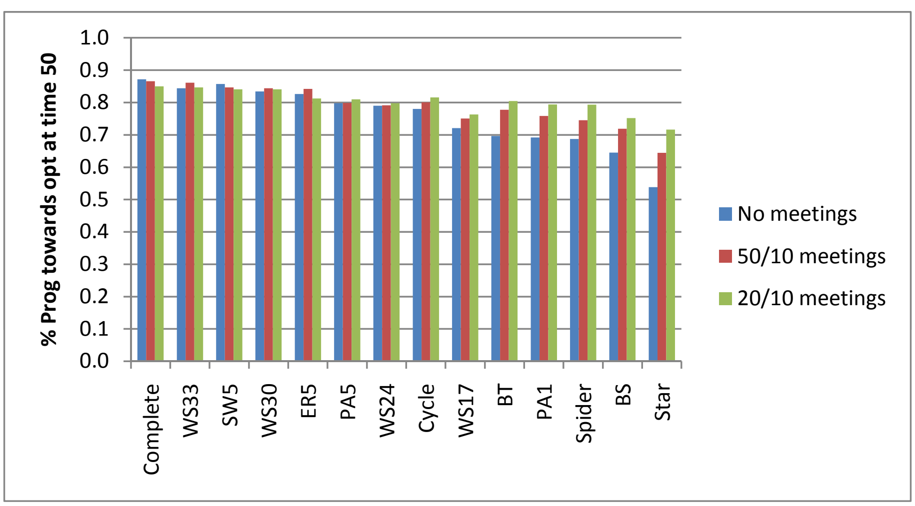


Figure 19: Impact of group meetings on performance with $k = 3$

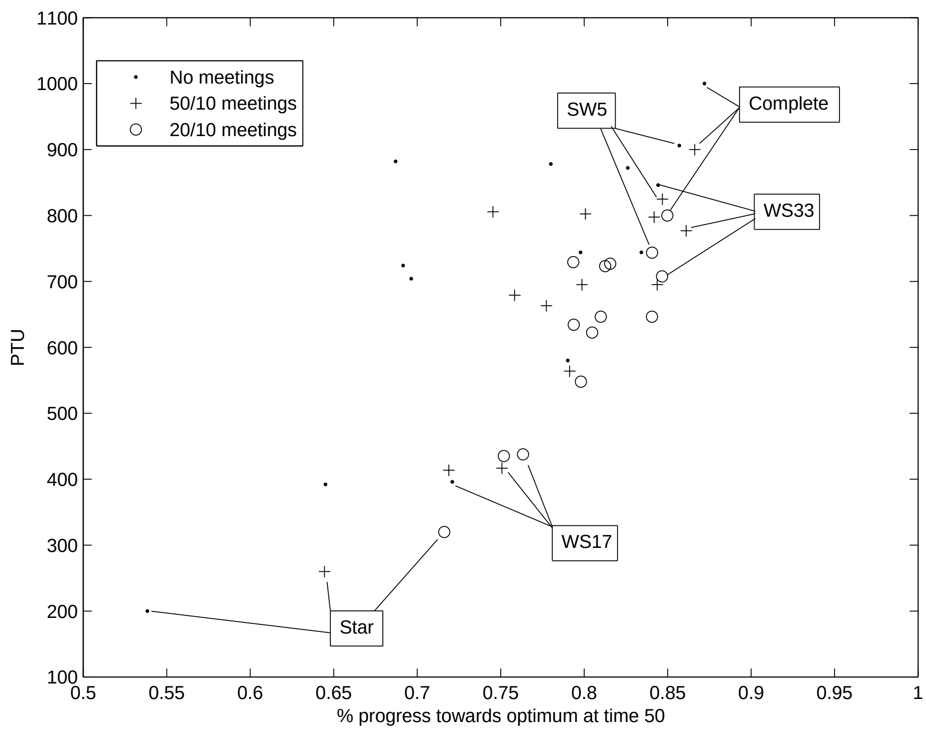


Figure 20: Efficient frontier with group meetings and $k = 3$

Appendices: Efficient Structures for Innovative Social Networks

W.S. Lovejoy A. Sinha

November 12, 2009

A The landscape of graphs and vital statistics

Our study includes 108 networks in total, spanning a wide variety of network types and structures. They range from the well-connected to the sparse, trees to bipartite graphs, random to very structured, and cover a continuum of values along all network metrics. We first briefly describe the networks in our collection, and then examine the range of values of the various network metrics covered by our networks. Because all our experiments were with 10-node networks, we only describe 10-node networks here. The extension to general n-node networks is straightforward.

While most of our networks fall into various broad categories, some are more difficult to categorize. Most of these are derivatives of the broad categories of networks. These networks are described below by first stating the broader family of networks we start with, and then listing additional edges as unordered pairs (x, y) of their end-points x and y. Since the broader family of networks follow unique labeling systems (shown in the figures), such a description for our derivative networks also allows us to reconstruct them unambiguously.

A.1 Trees

A tree is a connected network where there are no cycles, so there is a unique path between any two nodes. Classical organizational hierarchies resemble trees, with a senior manager at the top connected to a few junior managers under his/her supervision, several other workers subordinate to the junior manager, and so on. The trees in our study are described below.

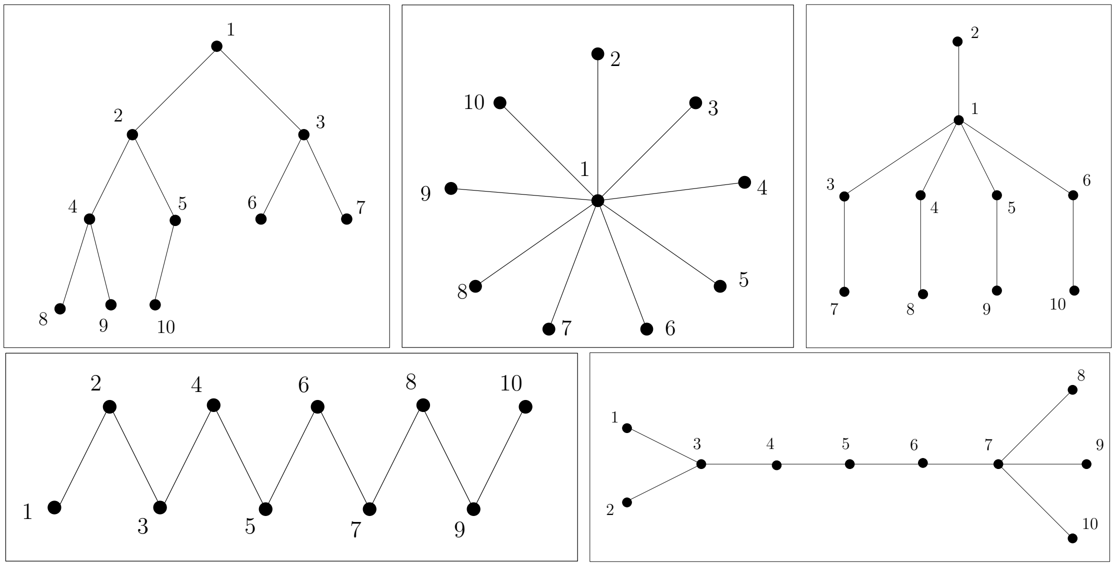


Figure 1: Tree networks: 1. Binary tree, 2. Star, 3. Spider, 4. Path, 5. Broom-star

1. **Binary Tree**: A binary tree is a network where each agent has two agents directly under him/her. The binary tree restricted to 10 nodes is shown in figure 1.1.

2. **Star**: A star has one agent at the center and all other agents connected directly to the central agent. The star is shown in figure 1.2.

3. **Spider**: In a spider, the single central agent is connected to several agents at the second level. Each of these second-level agents has exactly one other agent connected to them. The 10-node spider is shown in figure 1.3.

   We include several other networks that are derivatives of spiders and stars. These are described here. (i) Spider + 2 joined feet: The edge $(7, 8)$ is added to the spider. (ii) Spider + 2 joined knees: The edge $(3, 4)$ is added to the spider. (iii) Spider + one knee-foot edge: The edge $(3, 8)$ is added to the spider. (iv) Star + 1 spider leg: We start with a 9-node star (i.e. 8 leaves), and add the tenth node to one of the leaves. (v) Star + 2 spider legs: This is a 8-node star (i.e. 7 leaves), with the remaining two agents each connected only to one distinct star leaf. (vi) Star + 2 spider legs joined at feet: We start with the previous network, and join the two legs of the spider by a new edge. (vii) Star + 3 spider legs: We start with a

7-node star, and connect the three remaining nodes each to one distinct star leaf. (viii) Star + 3 spider legs + 1 pair joined knees: This is the previous network, with a new edge added between two of the star leaves that have spider legs connected to them. (ix) Star + 3 spider legs + 2 pairs joined at knees: We start with a 7-node star, and add new nodes to three of the leaves resulting in 3 spider legs. Then two of the spider legs are joined by single edge between the nodes of degree 2. (x) Star + 3 spider legs + 3 pairs joined knees: As (ix) above, except that all 3 pairs of spider knees are joined with edges.

4. **Path**: A path is a network where all the agents are laid out on a line, and each agent is connected only to the agents immediately adjacent to him. Among all networks with a fixed number of nodes, the path maximizes characteristic path length and diameter. The path is shown in figure 1.4.

   We also considered three networks that are derivatives of paths: (i) 5Posts, defined as a path between agents 1 and 5, with each agent i on the path also connected to agent $5+i$; (ii) Posts(6-4), defined as a path between agents 1 and 6, along with the following additional edges: (1,7), (2,10), (5,9) and (6,8); (iii) Posts(6-4)_II, defined as a path between agents 1 and 6, along with the following additional edges: (2,7), (3,8), (4,9), (5,10). Observe that all three networks are also trees.

5. **Broom-Star**: Broom-star networks are parameterized by three numbers, and are denoted $BS(x-y-z)$, where $x+y+z=n-1$. The center of the network consists of a path with y edges (and therefore, $y+1$ nodes). At one end of the path, the end-node is connected to x other agents, none of whom are connected to each other. These nodes are labeled 1 through x. At the other end, the end-node is connected to z other agents, none of whom are connected to each other either; these nodes are labeled $11-z$ through 10. The network $BS(2-4-3)$ is shown in figure 1.5 Special cases of broom-stars include the regular star ($BS(9-0-0)$) and the path ($BS(0-9-0)$). Observe that a broom-star described as $BS(x-y-z)$ has a unique labeling of the nodes. We also include several derivative networks of broom-stars, althought these are not trees since they include edges added to the broom-stars which form

cycles. Given the unique labeling of the underlying broom-stars, one can easily construct these derivative networks uniquely. They described as the underlying broom-star, along with a list of additional edges. The broom-star derivatives we considered in our experiments are: (i) $BS(2-5-2)$+(1,4); (ii) $BS(2-5-2)$+(1,4)+(7,10); (iii) $BS(3-3-3)$+(4,6); (iv) $BS(3-3-3)$+(1,5)+(6,10); (v) $BS(4-1-4)$+(1,6); (vi) $BS(4-1-4)$+(1,6)+(2,3); (vii) $BS(4-1-4)$+(1,6)+(2,3)+(9,10).

A.2 Other connected graphs

In addition to the trees, our experiments include several other networks, with varying levels of connectivity. The important networks/network families among these are described below.

1. **Complete**: A complete graph is a network where every agent is connected to every other agent. With 10 nodes, such a network has 45 links, and each agent is guaranteed to have a conversation partner in every time period (i.e., $CPP = 5$). The complete graph is the only network that maximizes density and clustering coefficient (both at 1), and minimizes diameter and CPL (both also at 1).

2. **Complete Bipartite**: A complete bipartite graph is characterized by two numbers x and y, and denoted K_{xy}. It consists of x nodes in one set and $y = 10 - x$ in the other. No agent is connected to any agent in their own set, but each agent is connected to every agent in the other set. K_{55} is shown in figure 2.1, and is another network with $CPP = 5$. Our experiments also contain some networks that are extensions of complete bipartite graphs; for example, the network $K_{64} + c6$ consists of K_{64} with a cycle connecting the agents in the set of 6 nodes. Another such network (labeled $K_{64} + m6$) is K_{64} with the partition of 6 nodes further partitioned into 3 pairs of nodes, and an edge added between the nodes within each pair.

3. **Clique-Paths**: A clique-path is parameterized by two numbers x and $y = 10-x$, and denoted $CP(x, y)$. It consists of a clique (fully connected subset) of x nodes, with one of those nodes connected to one end-node of a path of the remaining y nodes. Special cases include $CP(10, 0)$

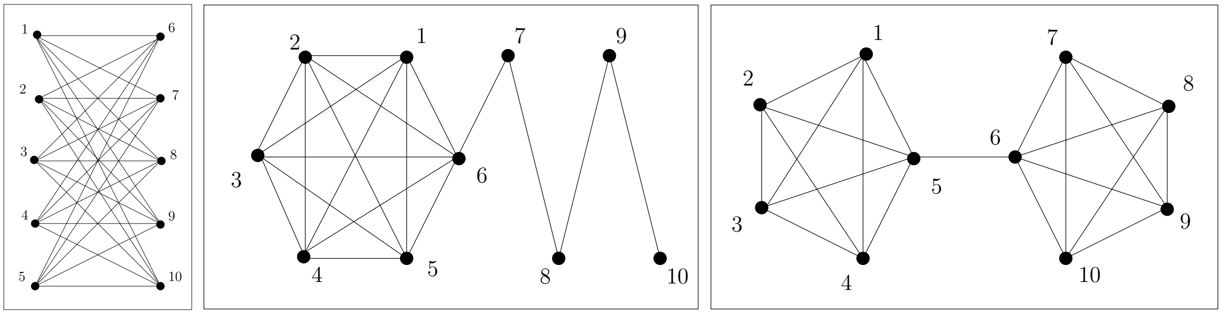


Figure 2: Clique-type networks: 1. K_{55}, a complete bipartite graph; 2. $CP(6,4)$, a clique-path; 3. $K5pK5$, two cliques connected to each other.

which is the complete graph, and $CP(0,10)$ which is the path. Clique-paths are interesting because for a given number of nodes and edges, these networks maximize CPL (Loch and Lovejoy 2003). The network $CP(6,4)$ is shown in figure 2.2.

4. **Two connected cliques**: This network consists of two cliques, one with x nodes and the other with $y = 10 - x$, which are connected to each other by means of a single edge. Such a graph is denoted $KxpKy$. They model organizations consisting of two well-connected teams that have a single line of communication between the teams. The network $K5pK5$ is shown in figure 2.3.

5. **Cycle**: A cycle is just a path where the two end-nodes are also connected to each other. The 10-node cycle is shown in figure 3.1.

6. **Cycle-Star**: A cycle-star is characterized by two numbers, x and $y = 10 - x$, and is denoted CxS. It consists of a cycle on x nodes, with the remaining y nodes each connected to exactly one node from the cycle. These y nodes are distributed as evenly as possibly among the cycle nodes. For example, the network $C4S$ is shown in figure 3.2.

   Three other networks in our collection are derivatives of Cycle-Stars. The first is $C3S + 1$: We start with the $C3S$ network, and connect with an edge two of the three leaves connected to the unique degree-5 node. The second is $C5S + (2, 5)$: an extra edge between two non-adjacent nodes in the cycle is added to $C5S$. The third is $C5xC5$: We start with $C5S$, and

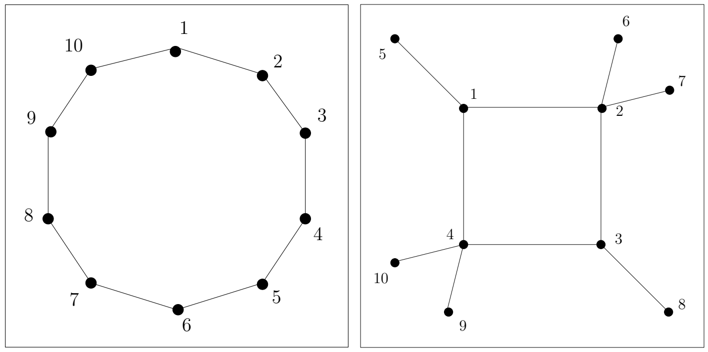


Figure 3: Cycle-type networks: 1. Cycle on 10 nodes; 2. $C4S$, a cycle-star.

connect the five leaves in a cycle in the same order as their parents are connected in the cycle of $C5S$.

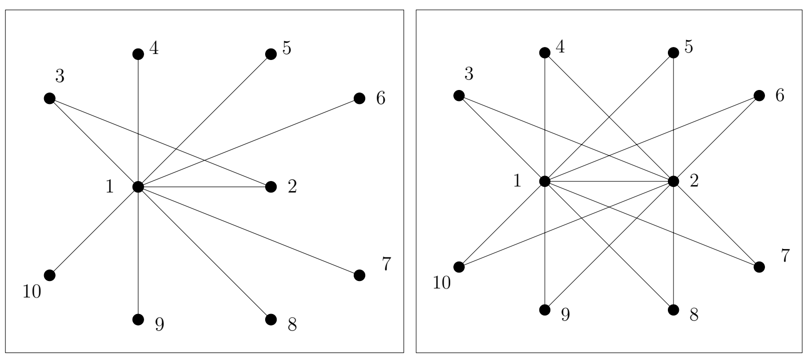


Figure 4: Wheel-star networks: 1. WS10; 2. WS17.

7. **Wheel-Star**: Wheel-star networks are characterized by a single number, x, denoting the number of arcs (since we have fixed the number of agents equal to 10). We denote the wheel-star with x arcs by WSx. Wheel-stars are a family of graphs generated by starting with a star ($WS9$ when $n = 10$) and then successively adding an arc from any non-fully-connected agent to the available node with the highest nodal degree (breaking ties randomly). Algorithmically, for an n-node network, let $\mathcal{L}_n$ be the lexicographically ordered list of all unordered pairs of distinct nodes. For instance, if $n = 4$, then $\mathcal{L}_4$ is the set $\{(1,2),\ (1,3),\ (1,4),\ (2,3),\ (2,4),\ (3,4)\}$. Then, for any $x \in [n-1, \binom{n}{2}]$, WSx is the network whose edges are the node-pairs corre-

sponding to the first x elements of $\mathcal{L}_n$. Thus for any n, the network $WS(n-1)$ is a star, while $WS\binom{n}{2}$ is a complete graph. For our 10-node networks, WS10 and WS17 are shown in figure 4. Wheel-star networks display a high degree of core/periphery structure, and our complete wheel-stars are the "idealized core/periphery" networks of Borgatti and Everett (1999).

We also considered three networks that are derivatives of Wheel-Stars: (i) WS10+(4,5) consists of WS10 with an additional edge (4,5); (ii) WS10+(4-5)(6-7) consists of WS10 with two additional edges, (4,5) and (6,7); (iii) WS10+3 consists of WS10 with three additional edges, (4,5), (6,7) and (8,9).

A.3 Random graphs

In addition to the 78 networks designed and chosen by us, our experiments also include 30 random graphs, 10 each from three distinct random graph models. This allows us to expand our landscape of networks to include those that are not influenced by any biases of our experiments, as well as validate our findings amongst a larger set of networks.

Our first set of random graphs are from the standard Bernoulli model, introduced by Erdös and Renyi (1960). With $n = 10$ nodes fixed, a Bernoulli random graph is obtained after fixing an edge probability p. Every pair of nodes has an edge between them with probability p, independent of all other node-pairs. For our experiments, we constructed 10 such random graphs (labeled ER1 through ER10), with different values of p in the range 0.2 to 0.95. We restricted our attention to connected Bernoulli random graphs, due to the nature of our model. One such graph, labeled ER5 and generated with $p = 0.4$, is shown in figure 5.

It has been observed that Bernoulli random graphs are not an appopriate model for the types of networks observed in practice, such as social networks or the internet. In recent years a literature has developed featuring other random graph models which mirror some structural properties in real observed networks. A survey of this literature appears in Albert and Barabási (2002). In this work, we focus on two of the more popular models, each aiming to replicate a distinct feature of observed networks.

The Small Worlds model of random graphs was proposed by Watts and Strogatz (1998), in an attempt to reconcile the relatively short path lengths of real networks compared to those found in Bernoulli random graphs. Graphs in this model are parametrized by n (total number of nodes), k (mean degree, even), and β (re-wiring probability). We start with a cycle on the n nodes, and add enough more edges so that each node is connected to all $k/2$ nodes to its left as well as all $k/2$ nodes to its right. Then, we consider nodes in the order $1, 2, \ldots, n$. When node i is considered, each of its edges (i, j) where $j > i$ is selected in turn. Each such selected edge is then re-wired with probability β: that is, with probability β, the edge (i, j) is replaced by edge (i, k) where k is selected uniformly at random from all nodes other than i and its neighbors. The random re-wiring results in the small-worlds phenomenon of short path lengths. Our experiments include 10 such Small Worlds random graphs, labeled SW1 through SW10, with k varying from 2 to 8 and β varying from 0.25 to 0.75. A Small Worlds random graph generated using $k = 4$ and $\beta = 0.25$, labeled SW5, is shown in figure 5.

A third model of random graphs is the Preferential Attachment model of Barabási and Albert (1999), which replicates the scale-free distribution of nodal degree found in observed real networks. Graphs in this model are parametrized by n (the total number of nodes), m_0 (number of nodes in initial set), and m (number of edges per new node). We start with a clique of m_0 nodes. The remaining $n - m_0$ nodes are added sequentially. When node i is added, it is connected to exactly m of the preceding nodes, and the probability of connecting node i to node j is proportional to the nodal degree of j at the moment just before i was considered. This process generates a "rich get richer" phenomenon – once a node has high degree, it is more likely to attract new edges from the new nodes, increasing its degree even more. Our experiments include 10 such Preferential Attachment random graphs, labeled PA1 through PA10, with m_0 varying from 2 to 6 and m varying from 1 to 4. The PA5 graph, generated using $m_0 = 4$ and $m = 2$, is shown in figure 5.

In figure 5, the ER5 graph shows no discernible structure. The SW5 network, on the other hand, displays how the 4 edges in the interior of the circle aid in reducing path lengths among the nodes on the circle. The PA5 network displays the scale-freeness of the nodal degree, with node 4 having a very high degree and other nodes relatively small.

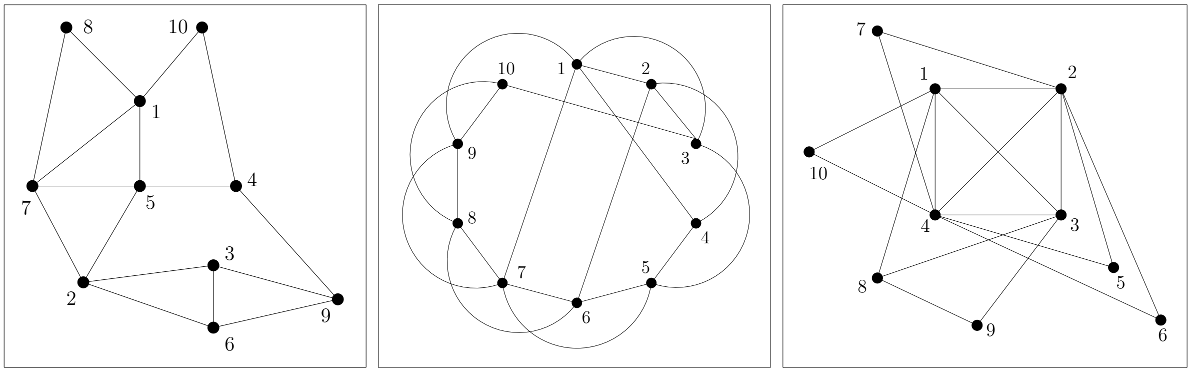


Figure 5: Random networks: 1. ER5, generated with $p = 0.4$ and containing 16 edges; 2. SW5, generated with $k = 4$ and $\beta = 0.25$, containing 20 edges; 3. PA5, generated with $m_0 = 4$ and $m = 2$, containing 18 edges.

We briefly explore the probabilities of generating some of our important structured graphs via these random processes. Recall that our graphs are *labeled*; that is, each node is identified by a unique label. Thus for any chosen graph shape, there may be more than one labeled graphs isomorphic to it. For example, there are 10 possible labeled stars, each obtained by letting a different node be the star center. For a graph shape GS, let $N(GS)$ denote the number of labeled versions of the graph shape (so $N(\text{star}) = 10$). Let $P_l(GS, p)$ denote the probability that the Bernoulli random graph process with $n = 10$ nodes and edge probability p generates one specific labeled graph of shape GS. So, for example, $P_l(\text{star}, 0.2) = 0.2^9 \times 0.8^{36}$, because for one specific labeled star to be generated, we need nine specific edges to be generated and the remaining 36 node-pairs to not have an edge between them. In general, if the graph shape GS has $m(GS)$ edges, then it is easy to see that $P_l(GS, p) = p^m(1-p)^{45-m}$. Therefore, the probability $P(GS, p)$ that the Bernoulli random graph process with edge probability p generates the graph shape GS is now given by:

$$P(GS, p) = N(GS) \times P_l(GS, p)$$

A simple calculation of the first-order conditions reveals the following observation.

Observation 1 *$P(GS, p)$ is maximized when $p = m(GS)/\binom{n}{2}$.*

Graph shape	$N(GS)$	p^*_{GS}	$P(GS)$
Star	10	0.2000	1.662×10^{-9}
Path	10!/2	0.2000	6.029×10^{-4}
WS10	360	0.2222	1.600×10^{-8}
WS17	45	0.3778	4.974×10^{-12}
Complete	1	1.0000	1.000

Table 1: Probabilities of Bernoulli random graph process generating specific graph shapes

It is now straightforward to compute the probabilities $P(GS)$ that the Bernoulli random graph process with the optimally chosen edge probability generates a random graph of shape GS. These probabilities for some networks are shown in table 1, where the column $p^*_{GS} = m(GS)/\binom{n}{2}$ denotes the optimal edge probabilities.

We observe that the probabilities for specific graphs being generated are extremely small. The efficiency of the wheel-star family of graphs is one of the major findings of this paper; we observe that the probability of WS17 being randomly generated is approximately 5 in a trillion.

A.4 Network metrics

A brief overview of the network metric statistics in our collection of 108 networks is shown in Table 2. For many network metrics such as CPL, diameter, CPP, density, clustering coefficient, etc., our collection of networks includes graphs with the extreme (maximum and minimum possible values) of the metric, and a near-continuum of values in between. In figure 2, "conversational CPL" and "conversational diameter" are the critical path length and diameter of the graph if we value each arc at the inverse of its probability of being active in conversation each period, so the "distance" increases as the probability of the arc being active decreases. We felt initially that these new graph metrics should be predictive of performance, because the usefulness of an arc seemed intuitively to be inversesly related to its probability of being used. However, this was not the case, in general. A graph with many arcs can have very low conversational probability on each one, but perform well because there are many paths to choose from. Each arc is chosen with low probability,

Statistic	Min	Max	Mean	Stdev
CPL	1.0000	3.6667	1.9938	0.6484
Diameter	1.0000	9.0000	3.5421	1.7336
CPP	1.0000	5.0000	3.8037	0.8940
Density	0.2000	1.0000	0.4037	0.2310
Var(deg)	0.0000	8.7111	2.5902	2.2904
Conv. CPL	6.2015	20.5556	10.6159	2.8203
Deg. centrality	0.0000	1.0000	0.3525	0.2526
Clust. coeff.	0.0000	1.0000	0.3857	0.3517

Table 2: Network statistics

but knowledge flows nicely. We include these new metrics as another indicator of the breadth of our test set of graphs.

Figure 6 shows the spread of our graphs along the nine metrics listed in table 2, where all metrics have been normalized to $[0,1]$. We see that for most metrics, the spectrum of graphs included does span the possible values of the metric. The gaps in CPP are not really indicative of poor choice of networks; instead, there are no networks possible with those values of CPP as will be shown in the next section of this appendix. We also observe that for some metrics such as Var(deg) and Deg. centrality (among others), the space covered by random graphs is a fraction of the space of possible values–once again pointing to the merits of carefully choosing a consciously designed set of graphs on which to conduct this study.

The relatively high standard deviation is also indicative of the large variability along these metrics in our collection of networks.

A.5 Cartesian products of metrics

A central question in our methodology is whether our selected 78 networks cover the landscape of all possible networks among 10 nodes. To guard against a bias, we included 30 random networks as noted. However, random graphs tend to cluster in certain regions of the space of possibilities,

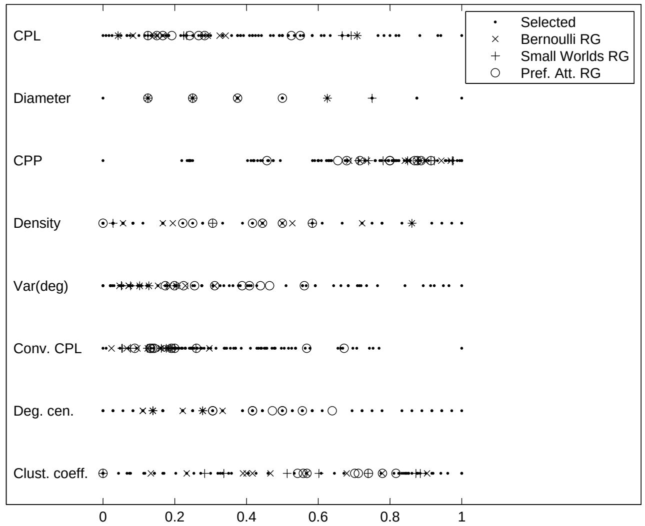


Figure 6: Spread of graphs on various network metrics. All values have been normalized to $[0, 1]$ for better visual representation.

and so are unlikely to span that space. For instance, for most values of the edge probability p, most Bernoulli random graphs have similar diameters (see Bollobás 2001 for this and other results about random graphs). Our selection of 78 consciously generated graphs significantly expands the range of characteristics of our test set.

To support this assertion, we study the network metrics of the two classes of graphs. For instance, consider figure 7, where we place each of the 108 graphs on an $X - Y$ scatter diagram with diameter on the X-axis and CPP on the Y-axis. The 78 selected graphs and 30 random graphs are displayed with different markers. As can be seen, the random graphs occupy only a small portion of the chart, with diameter varying from 2 to 7 and CPP from about 3.0 to 5.0. In contrast, the 78 selected graphs significantly expand the space of possible (diameter,CPL) values, with diameter values in the range $[1, 9]$ and CPP values as low as 1. In fact, the 30 random graphs are fully contained inside the convex hull formed by the 78 selected graphs. Furthermore, there exist no graphs in areas far outside our landscape. We show below that the space of possible (diameter, CPP) values is bounded by the polygon shown in figure 7. The scatter plot and boundary show that our consciously selected graphs span close to the entire possible space of (diameter, CPP) values.

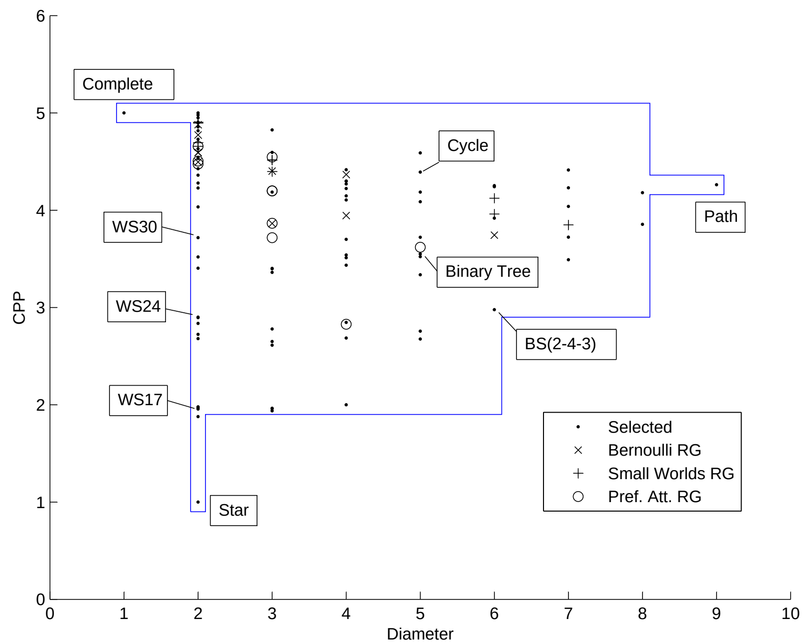


Figure 7: Diameter vs. CPP of the two classes of networks

Boundary of possible graphs in (diameter, CPP) landscape The north boundary of the landscape is fairly straightforward: a graph with 10 nodes cannot have CPP greater than 5. Likewise, a graph with 10 nodes cannot have diameter greater than 9, and the only diameter-9 graph is the path: this gives us the east boundary. The only graph with diameter 1 is the complete graph; any other graph has at least two nodes with no edge between them, and therefore a distance of at least 2 between them. This yields the west boundary. The south boundary is the most interesting one, and is derived from the propositions below (with proofs in appendix D).

Proposition 2 *Except for the star, all n-node networks with at least 4 edges have CPP at least $2 - \frac{n+1}{n(n-1)}$ (or 1.878 in our 10-node networks).*

The tight case for the proof above is a star with two of its leaves connected by an edge. For the 10-node case, this network can be verified to result in a CPP of 1.878.

Proposition 3 *A network with diameter m has CPP at least $\lceil m/3 \rceil$.*

In fact, the proposition above is of the worst-case type: in any diameter-m network, in any sample path for the matching algorithm, the CPP is at least $\lceil m/3 \rceil$. We care about the average CPP rather than the worst-case CPP. So while the proposition above provides a lower bound for the average CPP, it's not a tight lower bound.

The two propositions above combine to yield the south boundary of figure 7.

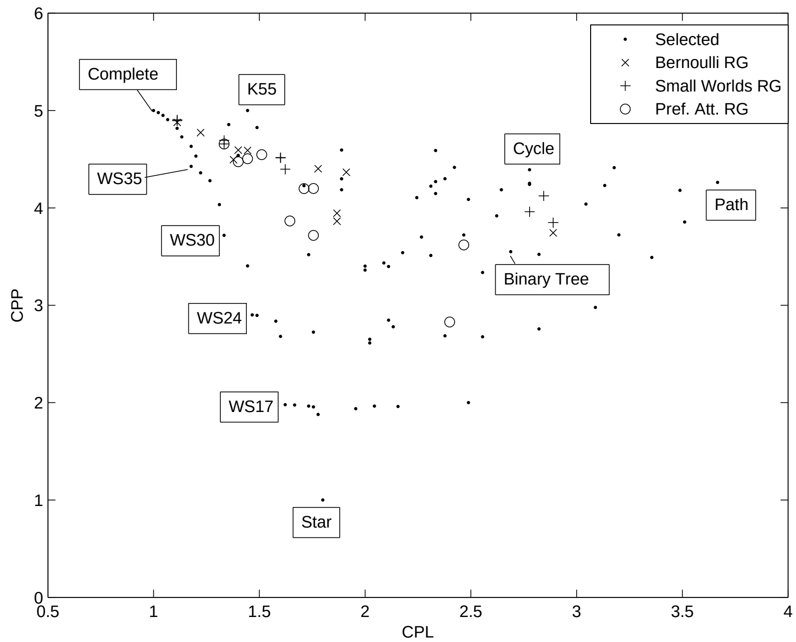


Figure 8: CPP vs. CPL of the two classes of networks.

These phenomena (spread of selected graphs, narrow range of random graphs) are observed when any two network metrics are plotted on scatter diagrams. Two more such scatter diagrams are displayed in figures 8 and 9. Figure 8 shows a plot of CPP vs. CPL, while figure 9 shows clustering coefficient vs. degree centrality. As can be seen, all 30 random graphs are fully contained in the convex hull formed by the 78 selected graphs for both the plots. Furthermore, the random graphs typically occupy a smaller region of the scatter diagrams, and may exhibit strong patterns of relationship (e.g. in the CPP vs. CPL plot). These results and plots lend credence to our assertion

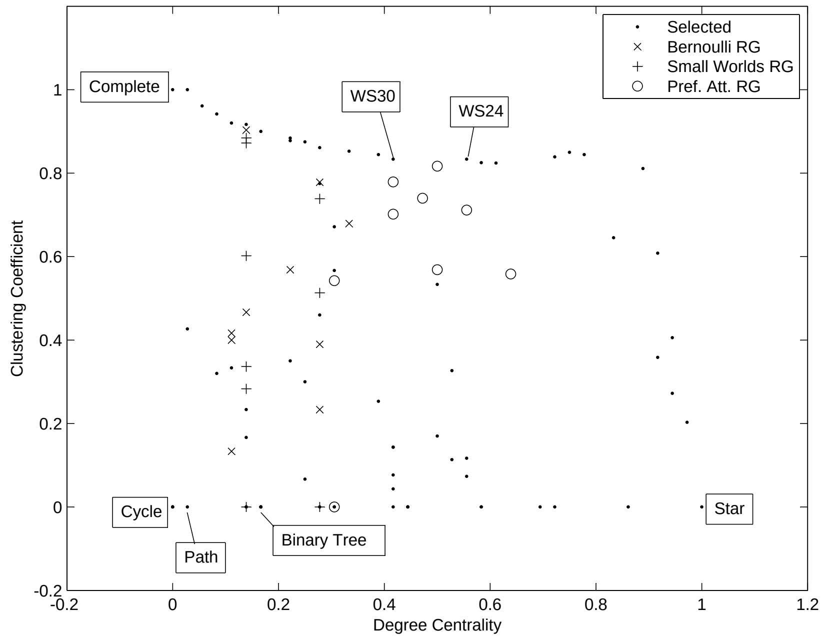


Figure 9: Clustering coefficient vs. Degree centrality of the two classes of networks

that our 78 selected networks cover a large diversity of possible network types and characteristics.

A.6 Networks with 5, 8 or 12 nodes

In Section 8.1, we describe running our experiments with networks of 5, 8 and 12 nodes to assess the robustness of our findings with respect to network size. For some network structures (cycle, complete graph, star, spider, binary tree), the corresponding networks with 8 or 12 nodes are defined unambiguously. For others, we had to carefully define the corresponding networks. For example, the complete wheel-stars have a different number of arcs in an 8 node graph than a 10 or 12 node graph. We used the complete wheel-stars in each, because our claim is these are efficient graphs when minimizing TTF subject to a bound on CPP (or vice versa). In addition to the star and complete graph, our test set of networks (with 10 nodes) included 3 complete Wheel-Stars (WS17, WS24, and WS30) which had 2, 3 and 4 agents in the core, respectively. Therefore, to obtain the corresponding 8 and 12 node networks, we again created the complete wheel-stars with

2, 3, and 4 agents in the core. This resulted in WS13, WS18, and WS22 for the 8-node networks, and WS21, WS30, and WS38 for 12-node networks.

The structure of BS(4-1-4) consists of one edge in the middle (the broom), with the remaining 8 nodes equally distributed in the form of stars at each of the two ends of the broom. Thus, the corresponding broom-stars with 8 and 12 nodes, respectively, are BS(3-1-3) and BS(5-1-5).

Finally, to generate the corresponding random graphs, we used the same parameters that were used to generate ER5, SW5, PA1, and PA5 for the 10-node networks. In particular:

- The ER5 graph was generated with $p = 0.4$, so the ER graphs for both 8 nodes and 12 nodes were generated using $p = 0.5$.
- The SW5 graph was generated with $(K, \beta) = (4, 0.5)$, which were the same values used to generate the SW graphs for 8 and 12 nodes.
- The PA1 graph with 10 nodes was generated with $(m_0, m) = (2, 1)$, and the same values were used to generate the PA1 graphs for 8 and 12 nodes.
- The PA5 graph was generated with $(m_0, m) = (4, 2)$, which were the same values used to generate the PA graphs for 8 and 12 nodes.

All networks with 5 nodes Even with just 10 nodes, the number of different networks that are possible is more than 11.7 million (Sloane, 1973). Although we believe that our set of 108 networks comprehensively covers the different types of network structures and characteristics that exist on 10-node networks, we do not exhaustively enumerate over all 10-node networks. However, if the number of nodes is reduced, it is possible to exhaustively enumerate all networks to check our results. We did this for 5 node networks, for which there are 21 distinct, connected graphs. Some of these are shown in Figure 10. The complete Wheel-Stars on 5 nodes are WS7 and WS9. As documented in Section 8.1 of the manuscript, our simulation results over the entire set of 5-node networks confirm our main findings.

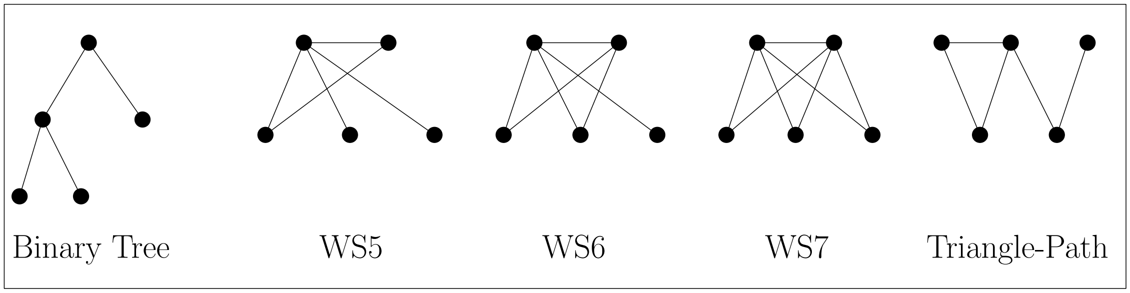


Figure 10: Selected 5-node networks.

B Analytical expressions for the expected number of conversations per period

With random matching the conversations per period (CPP) is an intrinsic feature of a sociomatrix, like its density or diameter. A star has $CPP = 1$ for any number of agents n. The CPP for a complete graph with n agents is the greatest integer less than or equal to $n/2$. Below we derive expressions for the CPP's for a path, cycle, and binary tree.

1. Path

Let $P(n)$ be the expected number of conversations per period for a path of length n, where nodes are chosen at random and then their partners chosen at random. We initialize $P(0) = P(1) = 0$ and $P(2) = 1$.

There is a probability of $2/n$ of choosing a leaf, in which case there is only one partner to choose from, resulting in $1 + P(n-2)$ conversations. We choose each interior node ($2 \leq i \leq n-1$) with probability $1/n$, and then go "right" or "left" with equal probability, each time splitting up the path into two paths. With probability $(1/n)(1/2)$ we choose agent i and go "left" resulting in $1+P(n-i)+P(i-2)$ conversations. With probability $(1/n)(1/2)$ we choose agent i and go "right" resulting in $1 + P(n-i-1) + P(i-1)$ conversations. So

$$P(n) = \frac{2}{n}(1 + P(n-2)) + \frac{1}{2n}\sum_{i=2}^{n-1}[2 + P(n-i) + P(i-2) + P(n-i-1) + P(i-1)]$$

This reduces to

$$P(n) = 1 + \frac{2}{n}P(n-2) + \frac{1}{n}\sum_{i=1}^{n-2}(P(i) + P(i-1))$$

and we can derive a simpler recursion by noting that

$$\frac{1}{n}\sum_{i=1}^{n-2}(P(i) + P(i-1)) = P(n) - 1 - \frac{2}{n}P(n-2)$$

and plugging this into the expression for $P(n+1)$ generates

$$P(n+1) = \frac{1}{n+1} + \frac{n}{n+1}P(n) + \frac{3}{n+1}P(n-1) - \frac{1}{n+1}P(n-2)$$

or

$$P(n) = \frac{1}{n}[1 + (n-1)P(n-1) + 3P(n-2) - P(n-3)].$$

Figure 11 shows the average number of conversations per period for different values of n.

n	$P(n)$
3	1
4	1.75
5	2
6	2.54
7	2.93
8	3.4
9	3.82
10	4.26

Figure 11: Avg. conversations per period for paths of different lengths.

2. Cycle

A cycle is now easy, because if we choose any arc at random we are left with a path of length $n-2$, so the number of conversations per period for a cycle equals $1 + P(n-2)$.

3. Binary tree

The computations for the expected number of conversations for a binary tree are complicated by the probabilities of the various agents being in conversation in different ways. But, an approximation with intuitive appeal is to look at top-down and bottom computations as approximations. Consider a regular binary tree of depth L. That is, a single node is $L = 0$, three nodes and two arcs are $L = 1$, etc.). By "regular" we mean that level k has 2^k nodes in it, so each level is "filled." In a top-down approximation the top node always talks to a neighbor, and removal of that neighbor and the top node's alternative arc from consideration, leaves us with one tree of depth $L-1$ and two trees of depth $L-2$ each. So $C(L)$ (the expected number of conversations in a regular binary tree of depth L) satisfies

$$C(L) = 1 + C(L-1) + 2C(L-2).$$

If we go bottom up, we assume that half of the leaves (there are 2^L of these) talk to their immediate superior, and after removal of all leaves and the superiors we are left with a tree of depth $L-2$. So,

$$C(L) = (1/2)2^L + C(L-2) = 2^{L-1} + C(L-2).$$

It can be verified that these two approximations yield exactly the same expressions for $C(L)$ and also equal

$$C(L) = \sum_{j=0}^{\frac{L-2}{2}} 2^{2j+1} \text{ if } L \text{ is even}$$

and

$$C(L) = \sum_{j=0}^{\frac{L-1}{2}} 2^{2j} \text{ if } L \text{ is odd}$$

All of these expressions are equivalent. Our 10-agent tree is not regular, so these approximations do not apply directly, but using this top-down and bottom-up heuristic anyway suggests that the number of conversations per period is between 3 and 4. Brute force enumeration of all possible cases in our 10-agent tree reveals that the theoretical number of conversations per period is 3.344.

C POMDPs and deliberate conversational strategies

In our model the bit strings held by each agent are abstract proxies for knowledge. While this is a reasonable modeling abstraction to look at macro performance of the network, an agent-specific sequential decision process using bit strings as the state space may be too precise a representation of how real people perceive knowledge in themselves and others, and the learning potential for talking with one neighbor versus another. Rather, people may have a rather imprecise, qualitative notion of the potential learning benefits available from alternative partners. We can plausibly represent this as a subjective probability that something will be learned in conversation, which real people can relate to. We seek an analytical model of partner choice that is based on those current subjective beliefs, which themselves must be a function of a history of realistic and practical observations available to the individuals involved: Who did I talk to and did I learn anything? Finally, the beliefs should be updated in a manner consistent with rationality. These sorts of problems can be cast as "Partially Observed Markov Decision Problems" (POMDP's, references below), a realistic but very difficult class of sequential decision problems. We can, in this case, make some approximations that yield relatively clean insights. We first build up the POMDP model from our underlying bit-string representation of knowledge to the actual observable facts to individuals, and then introduce the approximations and results.

Let agent 0 be the decision maker and let Δ_i denote the number of bits for which agent i has a 1 and agent 0 has a zero. Then, the probability agent 0 will learn something in conversation with agent i is Δ_i/m where recall m is the length of the string. Suppose agent 0 is not sure what agent i knows, but has a subjective probability distribution on Δ_i. Specifically, let π_{ik} be the probability that $\Delta_i = k$, for $0 \leq k \leq m$. Now the probability of learning something in conversation with agent i is

$$R_i(\pi) = \sum_{k=0}^{m} \pi_{ik} \frac{k}{m}.$$

In practical contexts, people may have an intuitive sense for this probability of learning something ($R_i(\pi)$), but the precise manner in which that probability arises may be less easy to articulate. What we seek are optimal conversational policies in the decision problem with the subjective probabilities on Δ_i as state variables, but that also make sense intuitively and are implementable in practical reality. The approach is to derive optimal, or approximately optimal, policies assuming the more detailed representation of knowledge, and note that if these are implementable with much less specific knowledge then they will be optimal in the latter case as well (since one cannot do better with less knowledge). Finally, the policies should be intuitively credible.

So, first we construct the detailed decision problem. Let agent 0, the decision maker, have d (for nodal *degree*) neighbors. Each time period she can choose one of the d neighbors to talk to. For each neighbor i, she has a subjective probability distribution π_{ik} (for $0 \leq k \leq m$) on Δ_i. If agent 0 engages agent i in conversation and does not learn anything, the Bayesian posterior distribution on Δ_i is

$$\pi'_{ik} = \frac{((m-k)/m)\pi_{ik}}{\sum_{j=0}^{m}((m-j)/m)\pi_{ij}}.$$

If agent 0 learns something, however, there are two competing influences. First, it is a signal that Δ might be higher that agent 0 believed. But second, when agent 0 gains a bit from agent i, then of course the actual Δ declines. A learning conversation with agent i can build confidence that agent i has a lot to offer, but this is in tension with the knowledge that the more knowledge you exchange the less remains to learn from agent i. Technically, if agent 0 learns from agent i then the posterior beliefs about what Δ was *before* the conversation is

$$\tilde{\pi}_{ik} = \frac{(k/m)\pi_{ik}}{\sum_{j=0}^{m}(j/m)\pi_{ij}}.$$

To adjust this to the posterior after the conversation, note that if we learned something in conversation then the updated probability that $\Delta_i = m$ is zero. We cannot have m learning opportunities left if we have just learned something. Further, the posterior probability that $\Delta_i = k$ is just the updated, but pre-conversation, probability that $\Delta_i = k+1$ for k from 0 to $m-1$. That is, $\pi'_{im} = 0$ and $\pi'_{ik} = \tilde{\pi}_{i,k+1}$ for $0 \leq k \leq (m-1)$.

Finally, any agent (agent j, say) not engaged in conversation with agent 0 may still have conversations with others in the network, and potentially gather a 1 where both they and agent 0 used to have a zero. This would increase Δ_j. On the other hand, agent 0 may learn something from agent i that agent j also knew, decreasing Δ_j. So, Δ_j for $j \neq i$ can also change from one time period to the next, in a manner that depends on the network and knowledge structure and matching of conversational partners outside the dyad of agents 0 and i.

Agent 0 wishes to choose a conversational strategy that maximizes her learning, and her rate of learning. We can cast this as a sequential decision problem, discounted over time to encourage accelerated learning strategies. The dynamic programming recursion (c.f. Bertsekas 1976) for this problem would be

$$V_{t+1}(\pi) = \text{Max}_i\{R_i(\pi_i) + \beta E_{\pi'|\{\pi,i,outcome\}} V_t(\pi')\} \tag{1}$$

where $\pi \in R^{md}$ is the concatenation of d (one for each neighbor) m-vectors π_i. The expectation is taken with respect to the posterior beliefs π', after conditioning on the prior π and the observed result (who she conversed with, and whether or not she learned anything). $\beta < 1$ is a discount factor that reflects the fact that learning sooner is better.

This is a "partially observed Markov decision process" or POMDP, a well-known but very difficult class of problems (c.f. Smallwood and Sondik 1973, Lovejoy 1991) that is largely intractable except for problems of special structure or small size. The name derives from fact that the core states of the Markov decision problem, Δ_i, are not known to the decision maker, but are "partially observed" via correlated signals from actually observed events, and the decision maker uses these signals to update her beliefs about the core states. In addition to its realism, the importance of this class of problems is its representation of the tension between exploration and exploitation, that is between accepting a delay to gather more information, or disallowing additional delay and acting on the information already on hand. We will see this tension in the examples below.

POMDP's are theoretically straightforward. Indeed, since R_i is bounded between 0 and 1, for any discount factor $\beta < 1$ the above recursion is guaranteed to converge to a unique fixed point V^* of the functional equation (1) and the maximization on the right hand side (using V^*) reveals the optimal policy at any belief vector π (c.f. Bertsekas 1976 or Puterman 1994 and references there). The problems with POMDP's are numerical and structural. Although Δ_i can take on only a finite number of integer values the state space for π_i is uncountably infinite, making numerical computation difficult or impossible. Also, structural results (even simple things like monotonicity of the value function or policy) are impeded in most cases by Bayesian updates that defy easy structural conclusions (c.f. Lovejoy 1987 references there). However, there are some approximations to agent 0's decision problem that can inform an optimal conversational policy, are intuitively compelling, and can be implemented realistically relying only on practically observable events without reference to modeling abstractions (e.g. bit strings or Δ).

Suppose that agents j for $j \neq i$ do not change their core states Δ_j when not engaged in conversation with agent i (we will discuss the relaxation of this assumption below). Then, only component $\pi_i \in R^m$ of π (i indexing the conversation partner) is updated each time period. This belongs to a class of problems called "multi-armed bandit" problems, an allusion to a slot machine with multiple arms with unknown payoffs. The decision maker must choose which arm to pull, in what sequence, all the time updating her beliefs about the payoffs for each based on the outcomes. By pulling repeatedly on one arm, the decision maker will learn more about that arm's payoffs, but learn nothing about other arms which may, in fact, have higher payoffs. This class of problems is the simplest to feature the tension between exploration and exploitation. Gittins and Jones (1974) and Gittins (1979) showed that this class of problems admits an optimal "index policy," which is a decomposition result in which smaller decision problems can be solved for each arm (agent) independently to compute an index for that arm (agent), and it is optimal for the decision maker to choose the arm (converse with the agent) with the greatest index. This history and an accessible derivation of the index policy, along with the now common terminology "Gittins index," appear in Whittle (1982).

Specifically, a modified dynamic programming recursion is used for each agent independently, but with the added feature that the decision maker can quit at any time and garner a reward of M. For agent i this modified recursion is

$$V_{i,t+1}(\pi_i, M) = \text{ Max } [M; R(\pi_i) + \beta E_{\pi_i'|\pi_i} V_{i,t}(\pi_i', M)].$$

Again, the sequence of functions $V_{i,t}$ generated by this recursion will converge to a unique fixed point V_i^* satisfying

$$V_i^*(\pi_i, M) = \text{ Max } [M; R(\pi_i) + \beta E_{\pi_i'|\pi_i} V_i^*(\pi_i', M)]$$

and the optimal action for agent i at any π_i can be computed from the right hand side maximization.

The Gittins index for agent i at belief π_i is

$$M_i(\pi_i) = \text{ Inf}\{M|M = V_i^*(\pi_i, M)\}.$$

The relationship between these decomposed problems and the original aggregate problem is that at any time period it is optimal in the larger problem for agent 0 to converse with the agent with the highest index $M_i(\pi_i)$. Intuitively, we solve a sequential decision problem for each agent i assuming that there is a constant outstanding outside offer as an alternative to continuing with agent i. The Gittins index for agent i is the level of offer that makes us just indifferent between continuing with agent i or abandoning agent i to accept the outside offer. Clearly this is related to the value of continuing with agent i, and mathematically the larger decision problem of choosing among multiple agents is solved by always choosing the agent with the largest index.

An index policy offers key computational advantages, since we need only solve a dynamic program for each agent i independently to solve the joint problem of dynamically choosing among the d neighboring agents. Still, however, the agent-specific problems are POMDP's, albeit smaller ones, and therefore troublesome. We can say something, however, about special cases. These are situations where there are only two possible core states Δ_i, because then $\pi_i \in R^2$ is completely determined by just one of its components, the other predetermined by the constraint that the two components sum to unity.

C.1 One bit left to learn, and the LET policy

Suppose agent 0 has just one bit left to learn, and let π_i now represent the probability that agent i has a 1 in the required location. Of course, practically agent 0 will not know she has just one bit left to learn. However, if we can find a policy that is optimal for agent 0 given ample information when she should know much better exactly what to do, but that is implementable by that agent with realistically minimal information, clearly the policy is optimal in the realistic situation.

If agent 0 knows she has one bit left to learn, converses with agent i and learns, the process stops because she has the great idea. However, if agent 0 converses with agent i and does not learn, she updates her priors to

$$\pi_i' = \frac{\pi_i \frac{m-1}{m}}{\pi_i \frac{m-1}{m} + (1-\pi_i)} \leq \pi_i.$$

Hence the expected return for speaking with agent i, $R_i(\pi_i) = \pi_i \frac{1}{m}$, is non-increasing with probability one as long as the process continues. This is the "deteriorating case" in Whittle (1982), for which the Gittins index is $R_i(\pi_i)$ or equivalently we can use π_i. If agent 0 begins with symmetrical priors on all of her neighbors ($\pi_i = \pi_j$ for all i and j), then she can choose her first partner randomly. If she chooses agent i and learns, the process stops. If she does not learn, then π_i deteriorates to π_i' and it is optimal to speak with anybody *but* agent i. If she learns from the next agent the process stops, and otherwise both of the first two conversational partners are left with deteriorated beliefs, and she chooses anybody but the first two. This continues until either agent 0 completes her learning, or all agents are left with the deteriorated π' at which point agent 0 can again choose randomly among them. It is easy to see that the central agent can practically implement this policy by always speaking to the agent she has not spoken to for the longest amount of time, breaking ties randomly. We call this the "longest elapsed time" or LET policy. Note that this policy is implementable with minimal actual information. The agent need only believe that she is very close to getting a great idea, the odds of which are better for individuals who have invested considerable time in the search already.

We now reconsider our assumption that agents not talked to do not change state. To include this feature formally would would yield a "restless bandit" problem. Restless bandit problems have no known analytical solution, although conjectures and approximations have been proposed that get us close to solid intuition and well-performing heuristics (c.f. Whittle 1988, Weber and Weiss 1990, Bertsimas and Nino-Mora 2000). In particular, Whittle (1988) provides an intuitive framing of the problem based on Lagrangian logic and a conjecture that an index-type policy will exist for restless bandits.

In our case, however, when the central agent has just one bit of knowledge to complete her learning, Δ_j for $j \neq i$ can only increase so the restless portion of the bandit problem can only enhance the attractiveness of agents not spoken to. This can only accelerate the abandonment of one's current partner, which reinforces the LET policy.

The LET policy has intuitive appeal. Talking to a neighbor but not learning anything, while other neighbors can engage in knowledge-enhancing side conversations, naturally suggests a churning of conversational partners as long as agent 0 is failing to benefit from her conversations.

C.2 A lot left to learn, and the SWW policy

Suppose that there are two possible core states (that is, two possible levels for Δ_i = the number of learning opportunities for agent 0 when conversing with agent i), k_1 and k_2 with $k_1 < k_2$. Agent 0 can choose to talk to any one of her d neighbors each time period. We first look at the problem where the states of each actor Δ_i do not change over time, and it is agent 0's problem to discover the most rewarding neighbor to talk to by experimentation. By assuming that Δ_i does not change for agents not in conversation with agent 0, we are assuming a standard (not restless) bandit structure as described above. By assuming that Δ_i does not change even for the agent in conversation with agent 0, we are assuming that even when learning occurs we still have, at least approximately, $\Delta_i/m \simeq (\Delta_i - 1)/m$, which will be true if $0 << k_1 < k_2 < m$. That is, agent 0 has so much to learn and her neighbors have so much to offer that the probability of learning from each one does not change appreciably even if learning occurs. We will later comment on the relaxation of this assumption.

Let π_i now be the probability that the core state for agent i is k_2, so that the probability that $\Delta_i = k_1$ is $1 - \pi_i$. That is, we now use as our state variable for the sequential decision problem the real number π_i that denotes the probability that agent i features Δ_i equal to the higher of the two possible core states.

If agent 0 converses with agent i and observes the outcome (she either learns something, or does not), then agent 0 will update her beliefs about agent i based on that experience in the typical Bayesian fashion. We will use the subscripts L and X to denote learning, or not. We will also use the function $L_i(\pi_i)$ to denote $mR_i(\pi_i)$, that is $L_i(\pi_i) = \pi_i k_2 + (1 - \pi_i)k_1$. These two uses of the letter L will be clear in context. If agent 0 talks to agent i and learns something, she updates π_i to

$$\pi'_{i|\{\pi,L\}} = \frac{\pi_i \frac{k_2}{m}}{\pi_i \frac{k_2}{m} + (1 - \pi_i)\frac{k_1}{m}} = \frac{\pi_i k_2}{L_i(\pi_i)}.$$

If agent 0 converses with agent i and does not learn anything, she updates π_i to

$$\pi'_{i|\{\pi,X\}} = \frac{\pi_i \frac{m-k_2}{m}}{\pi_i \frac{m-k_2}{m} + (1-\pi_i)\frac{m-k_1}{m}} = \frac{\pi_i(m-k_2)}{m - L_i(\pi_i)}.$$

The following proposition is proved in appendix D. The proposition statement here contains only those results required for our exposition. We define a "stick with a winner" (SWW) strategy as one in which the decision maker continues conversing with an agent as long as she learns something.

Proposition 4 *a)* $\pi'_{i|\{\pi_i,X\}} \leq \pi \leq \pi'_{i|\{\pi_i,L\}}$*, that is, learning enhances confidence and not learning decreases it. b) The Gittins index* $M_i(\pi_i)$ *is nondecreasing in* π_i*. c) An SWW strategy is optimal.*

Intuitively, since we have eliminated by assumption the diminishing returns feature of learning (the more agent 0 learns from a conversational partner, the less remains to learn from that partner), observing positive evidence that our partner is a wealth of needed knowledge argues for continued conversations. This sustains the relationship at least until the first conversation when learning does not occur. Mathematically an index policy is optimal for this problem and it is natural that the index M_i increase with higher probability on the higher state of knowledge k_2. This, and the increase in that probability after a learning experience (part a) results in the SWW policy.

An SWW policy tells us that once we engage a neighbor in conversation we continue to engage that neighbor until our first non-learning exchange. But, it does not tell us what to do after that. If our assumption of an unchanging core state Δ_i were exact, we might continue to engage that same neighbor. For example, after ten successes in a row we might not abandon that neighbor after just one failure. But, if we add back diminishing returns to learning then we accelerate the abandonment of our current partner. Further, if we don't learn from an interaction with neighbor i, then Δ_j for j for $j \neq i$ can only be enhanced by side conversations. This, too, accelerates the abandonment of neighbor i. These influences combine to suggest that agent 0 should not be too tolerant of failure.

As before, we seek a conversational policy that is informed by the analysis just presented, but is practically implementable in that it relies on easily observed facts. We propose that the decision maker begin by choosing a neighbor at random, and then sticking with that partner as long as learning occurs. Upon the first occasion of non-learning, she should choose an alternative partner using the LET logic, and then stick with that partner until failure, etc. We will call this the SWW-LET policy. Note that naturally this policy will reduce to LET when there is one bit left to learn, because the process only continues if a conversation fails (SWW is no longer relevant). So, SWW-LET invokes SWW early in the search process (with an LET choice of alternate partner upon failure) and reduces naturally to LET when the agent has one bit left to learn.

In the body of the paper we test the LET and SWW-LET strategies for comparison with the base case, which is a random choice of conversational partners among one's neighbors.

D Proofs of Propositions

Proposition 1 $E[NCTF] = CPP \times E[TTF]$

Proof: Let ω_t denote a sample path of outcomes (matches and knowledge exchanges) up to time t. Technically, ω_t is an outcome of a filtered stochastic process as in Harrison (1985). We will use ω without subscript to refer to the sample paths for the stochastic process, understanding that at time t we know exactly the outcomes up to time t and condition on that for the remainder when taking expectations. Define $CPP(t, \omega)$ to be the number of conversations in period t, and for any two time periods $s \leq t$ and sample path ω define $NCTF(s, t, \omega)$ to be the total number of conversations between those two time periods, so that

$$NCTF(s, t, \omega) = \sum_{j=s+1}^{t} CPP(j, \omega)$$

and

$$E_\omega NCTF(s, t, \omega) = (t - s)CPP$$

or $E_\omega NCTF(s, t, \omega) - (t - s)CPP = 0$.

Now, define the stochastic process $X_T(\omega) = NCTF(0, T, \omega) - T \times CPP$ for which we have

$$E[X_T | X_s] = E[X_s + X_T - X_s | X_s]$$

$$= E[NCTF(0, s, \omega) - s \times CPP + NCTF(s, T, \omega) - (T - s)CPP | X_s]$$

$$= E[NCTF(0, s, \omega) - s \times CPP + 0|X_s] = X_s.$$

That is, X is a Martingale, and from the Martingale stopping time theorem (c.f. Harrison 1985) we have that for any stopping time $\tau(\omega)$ the process X_τ is a Martingale. Specifically, letting τ be the first time that the requisite number of agents are completely learned, we have that $E[X_{\tau(\omega)}|X_0] = X_0$, or specifically since $X_0 = 0$

$$E[X_{\tau(\omega))}|X_0] = E[NCTF(0, \tau, \omega) - \tau(\omega) \times CPP|X_0] = X_0 = 0$$

. That is, starting at $X_0 = 0$ we have

$$E_\omega NCTF(0, \tau(\omega), \omega) = CPP \times E_\omega[(\tau(\omega)]$$

or

$$E[NCTF] = CPP \times E[TTF].$$

□

Proposition 2 *Except for the star, all n-node networks with at least 4 edges have CPP at least $2 - \frac{n+1}{n(n-1)}$ (or 1.878 in our 10-node networks).*

Proof: Consider a network with $CPP < 2$. For this to happen, there must exist an edge such that if it is active, then no other edges can be active. That is, there exists an edge e with end-points u and v such that all other edges are adjacent to it; we call this the blocking edge. Let d_u and d_v denote the degree of u and v respectively. Without loss of generality, we assume $d_u \leq d_v$. Let p_i denote the probability that a random matching has cardinality i, so that $CPP = \sum_{i=1}^{\lfloor n/2 \rfloor} i.p_i$. We bound CPP from below by bounding p_1 from above, using the fact that $CPP = \sum_{i=1}^{\lfloor n/2 \rfloor} i.p_i = p_1 + \sum_{i=2}^{\lfloor n/2 \rfloor} i.p_i \geq p_1 + \sum_{i=2}^{\lfloor n/2 \rfloor} 2.p_i = p_1 + 2(1 - p_1) = 2 - p_1$. We compute an upper bound on p_1 by considering the following exhaustive set of cases:

Case (i) $d_u = 1$: Since we know the edge (u, v) exists, $d_u = 1$ means all other nodes are connected to v. By virtue of (u, v) being a blocking edge, no edges exist connecting two neighbors of v. Therefore, the only edges in this network are those connecting v to all other nodes, so the network by definition is a star, with $CPP = 1$.

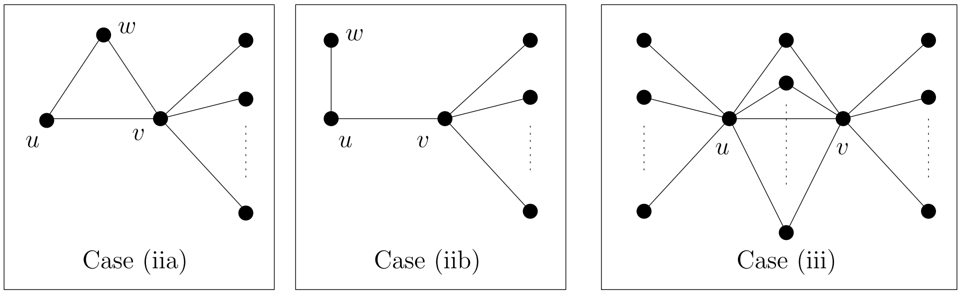


Figure 12: Illustrations of cases in proof of proposition 2

Case (ii) $d_u = 2$: We already know that one of u's neighbors is v; let the other neighbor be w. We distinguish between two sub-cases:

Case (iia) w is connected to v: An example of such a network is shown in figure 12. Note that all other nodes are only connected to v, by virtue of (u, v) being a blocking edge. Furthermore, since the network has at least 4 edges, it also has at least 4 nodes, which means that there is at least one node that is connected only to v. In this case, the matching has cardinality 1 if either (u, v) is chosen in the matching, or (v, w). The edge (u, v) is chosen in the matching if either u arrives first and chooses to talk to v (with probability $\frac{1}{n} \times \frac{1}{2}$), or v arrives first and chooses to talk to u (with probability $\frac{1}{n} \times \frac{1}{n-1}$). Thus (u, v) is chosen with probability $\frac{n+1}{2n(n-1)}$. By an identical calculation, (v, w) is chosen with the same probability, resulting in $p_1 = \frac{n+1}{n(n-1)}$.

Case (iib) w is not connected to v: An example of such a network is also shown in figure 12. In this case, for the random matching to have cardinality 1, the edge (u, v) must be chosen in the matching. This could happen if either u is the first agent to arrive and chooses to talk to v, or vice-versa. Resultantly, $p_1 = \frac{1}{n} \times \frac{1}{2} + \frac{1}{n} \times \frac{1}{n-2} = \frac{2}{n(n-2)}$.

Case (iii) $d_u \geq 3$: An example of such a network is shown in figure 12. Observe that some of u's neighbors may also be connected to v. Node u cannot be unmatched, because at most one of node u's neighbors can be matched to v, leaving at least one other node free to match with u. However, if node u is matched to any node other than v, then node v can also be matched to some other node (recall our assumption that $d_v \geq d_u$), resulting in a matching of cardinality at least 2. Therefore, for the CPP to be exactly 1, we must have once again that nodes u and v are matched

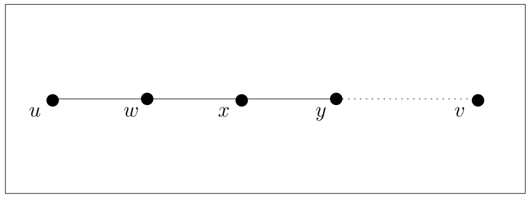


Figure 13: Illustration for proof of proposition 3

to each other. Again, for this to happen, either node u must be the first to arrive and select node v for conversation, or vice-versa. Therefore, $p_1 = \frac{1}{n} \times \frac{1}{d_u} + \frac{1}{n} \times \frac{1}{d_v}$. This probability is maximized when $d_u = 3$ and $d_v = n - 2$. Therefore, $p_1 \leq \frac{1}{n} \times \frac{1}{3} + \frac{1}{n} \times \frac{1}{n-2} = \frac{n+1}{3n(n-2)}$.

The above three cases are exhaustive, so we find that $p_1 \leq \max\{\frac{n+1}{n(n-1)}, \frac{2}{n(n-2)}, \frac{n+1}{3n(n-2)}\}$. For the network to have at least 4 edges, we must have $n \geq 4$; in that case, the first of the three bounds above is the maximizer, so $p_1 \leq \frac{n+1}{n(n-1)}$. Since $CPP \geq 2 - p_1$, we obtain $CPP \geq 2 - \frac{n+1}{n(n-1)}$, resulting in the bound claimed in the proposition statement. Setting $n = 10$ in the preceding bound yields $CPP \geq 1.878$. □

Proposition 3 *A network with diameter m has CPP at least $\lceil m/3 \rceil$.*

Proof: The proof is by induction. The base case is for $m \leq 3$, and is trivially proved. Suppose the proposition is true for $n = 1, 2, \ldots, m-1$, where $m - 1 \geq 3$. That is, suppose that for any network with diameter $n \leq m - 1$, the CPP is at least $\lceil n/3 \rceil$.

Now, consider a network where with diameter $m \geq 4$. Then, there exist two nodes u and v such that the shortest path between them has m edges. Let w denote the node adjacent to u in this path, and let x denote the node adjacent to w, and y denote the node adjacent to x, as shown in figure 13. Now consider any maximal matching in the entire graph, where a maximal matching is defined as a matching that leaves no two nodes that have an edge between them unmatched. Either node u or node w must be matched, because if neither was, then (u, w) could be added to the matching. Remove from the graph the matched edge as well as all edges adjacent to it, because none of the adjacent edges can be in any matching. Remove any resulting zero-degree nodes as well. Now consider what remains of the path between u and v. The segment $\{y, \ldots, v\}$ must remain, possibly in addition to some other edges in the path. The worst case is when the matched edge is (w, x), in which case the segment $\{y, \ldots, v\}$ is all that remains.

Therefore, removal of these edges and nodes still leaves a graph where there exist a pair of nodes with shortest path between them of at least $m-3$. That is, the diameter of the graph that remains is at least $m-3$. By the induction hypothesis, the CPP of such a network is at least $\lceil \frac{m-3}{3} \rceil = \lceil m/3 \rceil - 1$. Adding the one matching edge that was just removed, the total CPP of our initial network is therefore at least $\lceil m/3 \rceil$. □

Proposition 4 *a)* $\pi'_{i|\{\pi_i,X\}} \leq \pi \leq \pi'_{i|\{\pi_i,L\}}$. *b)* $M_i(\pi_i)$ *is nondecreasing in* π_i. *c) An SWW strategy is optimal.*

Proof: As noted in the text, we only stated the material needed for the exposition. The proof of these claims makes use of other facts regarding this bandit process, shown in the following Lemma. The first part of the lemma is also part (a) of the proposition.

Lemma 1 *La)* $\pi'_{i|\{\pi_i,X\}} \leq \pi \leq \pi'_{i|\{\pi_i,L\}}$. *Lb)* $L_i(\pi_i)$ *is nondecreasing in* π_i. *Lc)* $\pi'_{i|\{\pi,L\}}$ *and* $\pi'_{i|\{\pi,X\}}$ *are both nondecreasing in* π_i. *Ld)* $V_i^*(\pi_i, M)$ *is nondecreasing, convex and continuous in* π_i *and* M.

Proof: Since bandit processes are solved for each agent i independently, we will drop the subscript i for notational convenience. Hence $\pi'_{i|\{\pi_i,L\}}$ will be represented by $\pi'_{\pi,L}$, $L_i(\pi_i)$ by $L(\pi)$, etc.

La) $\pi'_{\pi,L} = \frac{\pi k_2}{L(\pi)} = \frac{k_2}{\pi k_2 + (1-\pi)k_1}\pi$ which is greater than or equal to π because $\pi k_2 + (1-\pi)k_1 \leq k_2$. $\pi'_{\pi,X} = \frac{\pi(m-k_2)}{m-L(\pi)} = \frac{m-k_2}{m-L(\pi)}\pi \leq \pi$ because $L(\pi) \leq k_2$ so $\frac{m-k_2}{m-L(\pi)} \leq 1$. This proves (La).

Lb) $L(\pi) = \pi k_2 + (1-\pi)k_1 = k_1 + \pi(k_2 - k_1)$ which is increasing in π since $k_2 > k_1$. This proves (Lb).

Lc) $\pi'_{\pi,L} = \frac{\pi k_2}{L(\pi)}$. So

$$\frac{\partial \pi'}{\partial \pi} = \frac{k_2}{L(\pi)}\{1 - \frac{\pi L'(\pi)}{L(\pi)}\}$$

and

$$\frac{\pi L'(\pi)}{L(\pi)} = \frac{\pi(k_2 - k_1)}{\pi k_2 + (1-\pi)k_1} = \frac{\pi(k_2 - k_1)}{k_1 + \pi(k_2 - k_1)} \leq 1$$

So $\frac{\partial \pi'}{\partial \pi} \geq 0$ and $\pi'_{\pi,L}$ is nondecreasing in π. Also

$$\pi'_{\pi,X} = \frac{\pi(m - k_2)}{m - L(\pi)}$$

where the numerator is increasing in π and the denominator decreasing in π (from part Lb), so $\pi'_{\pi,X}$ is increasing in π. This completes the proof of (Lc).

These first three parts of Lemma 1 formalize some intuitive results. If agent 0 learns in conversation she upgrades her priors and if she does not learn she downgrades her priors (part La). $L(\pi) = mR(\pi)$ and $R(\pi)$ is the probability of learning something, so we would expect L to increase as our priors increase that the agent has $\Delta = k_2$ rather than k_1 (part Lb). Finally, we would intuitively expect that regardless of what we observe, our posterior beliefs cannot get worse as our priors get better (part Lc). We note that these seemingly natural relationships do not necessarily hold if there are more that two core states, just one of the many troublesome aspects of POMDPs.

We next prove part (Ld), that the optimal value function $V^*(\pi, M)$ is nondecreasing, convex and continuous in π and M. That $V^*(\pi, M)$ is nondecreasing and convex in M is shown by Whittle (1982, Theorem 2.1 page 212). Continuity follows from convexity. That $V^*(\pi, M)$ is convex (and therefore continuous) in π is a well-known feature of POMDPs (c.f. Astrom 1965 or Smallwood and Sondik 1973; the latter proving this for finite horizons from which a limiting argument completes the proof). To prove that V^* is nondecreasing in π, assume inductively that $V_t(\pi, M)$ is nondecreasing in π. Let $\pi \geq \tilde{\pi}$, then

$$\begin{aligned}
E_{\pi'|\pi} V_t(\pi', M)] &= \frac{L(\pi)}{m} V_t(\pi'_{\pi,L}, M) + (1 - \frac{L(\pi)}{m}) V_t(\pi'_{\pi,X}, M). \\
&\geq \frac{L(\pi)}{m} V_t(\pi'_{\tilde{\pi},L}, M) + (1 - \frac{L(\pi)}{m}) V_t(\pi'_{\tilde{\pi},X}, M) \\
&\geq \frac{L(\tilde{\pi})}{m} V_t(\pi'_{\tilde{\pi},L}, M) + (1 - \frac{L(\tilde{\pi})}{m}) V_t(\pi'_{\tilde{\pi},X}, M)
\end{aligned}$$

the first inequality by the inductive hypothesis and the second because $V_t(\pi'_{\tilde{\pi},L}, M) \geq V_t(\pi'_{\tilde{\pi},X}, M)$ by (La) and $L(\pi)$ puts more weight on the greater of the two than $L(\tilde{\pi})$ does (Lb). Hence, $E_{\pi'|\pi} Vt(\pi', M)$ is nondecreasing in π. So is $R(\pi)$ and therefore $R(\pi) + \beta E_{\pi'|\pi} V_t(\pi', M)$, and hence

$$V_{t+1}(\pi_i, M) = \text{ Max } [M; R(\pi) + \beta E_{\pi'|\pi} V_t(\pi', M)]$$

is nondecreasing in π, completing the induction. Since $V_t \to V^*$ we have that V^* is nondecreasing. This completes the proof of the lemma. $\square$

To complete the proof of the proposition, we need to use these facts to show that $M(\pi)$ is nondecreasing in π and that a SWW policy is optimal. By Lemma 1 part (Ld), $V^*(\pi, M)$ is convex and non-decreasing in M, and by the dynamic program used to compute $V^*(\pi, M)$ we know that $V^*(\pi, M) = M$ for M large enough (certainly for M greater than $1/(1-\beta)))$. By convexity V^* is differentiable in M almost everywhere, and we can use the right side derivatives at any points of non-differentiability. Then, for M large enough $\frac{\partial V^*}{\partial M} = 1$ and by convexity $\frac{\partial V^*}{\partial M} \leq 1$ everywhere, implying that $M - V^*(\pi, M)$ is continuous and nondecreasing and equals zero for M large enough. The Gittins index is defined as

$$M(\pi) = \text{ Inf } \{M | M - V^*(\pi, M) = 0\}$$

which because $M - V^*(\pi, M)$ is continuous and nondecreasing we can equivalently define as

$$M(\pi) = \text{ Inf } \{M | M - V^*(\pi, M) \geq 0\}.$$

Now, let $\tilde{\pi} \leq \pi$. Because V^* is nondecreasing in π, we have

$$\{M | M - V^*(\pi, M) \geq 0\} \subseteq \{M | M - V^*(\tilde{\pi}, M) \geq 0\}$$

so that

$$M(\pi) = \text{ Inf } \{M | M - V^*(\pi, M) \geq 0\} \geq \text{ Inf } \{M | M - V^*(\tilde{\pi}, M) \geq 0\} = M(\tilde{\pi})$$

so $M(\pi)$ is nondecreasing in π. It remains to show that an SWW policy is optimal. If agent 0 chooses agent i it means that $M_i(\pi_i) \geq M_j(\pi_j)$ for all j. Then, if agent 0 learns something $\pi_i' \geq \pi_i$ so $M_i(\pi_i') \geq M_i(\pi_i) \geq M_j(\pi_j)$ for all j and it is optimal for agent 0 continue to converse with agent i. $\square$

E The Nk model for rugged preference landscapes

In Section 8.3, we described testing the validity of our conclusions if agents interacted in a rugged preference landscape. As described there, we use the Nk model of Kauffman and Levin (1987) to model the rugged preference landscape. Here, we examine in some detail how the Nk model captures ruggednss of the preference landscape.

The Nk model is specified by two integers, $N \geq 1$ and $k \in \{0, 1, \ldots, N-1\}$. N is the total number of bits in the belief string, and k is the number of other bits impacting the fitness of any single bit. Given an agent's belief string $b = (b_1 b_2 \ldots b_N)$, we assume that the bits impacting the preference value of bit position i are its immediate neighbors. That is, the preference value $f_i(b)$ of bit position i is a function of the vector of bits $b^i = (b_{i-\lceil k/2 \rceil}, \ldots, b_{i-1}, b_i, b_{i+1}, \ldots, b_{i+\lfloor k/2 \rfloor})$, where the subscripts are numbered modulo N (i.e., $b_{N+1} = b_1$, $b_{N+2} = b_2$, etc.). The preference value of the overall bit string $f(b)$ is simply the sum of the preference values of each bit position; $f(b) = \sum_{i=1}^{N} f_i(b)$. Following Evans and Steinsaltz (2002), the preference value at each bit position $f_i(b) = f_i(b^i)$ for every combination of bits b^i is drawn from an i.i.d. exponential distribution with unit mean.

For instance, if $k = 0$, then no other bit impacts the preference value of any bit, reverting to our base case with one global optimum and no other local optima. On the other extreme, if $k = N - 1$, then the preference value of any bit position depends on all other bits. In this case, the number of local optima could be as large as $\frac{2^N}{N+1}$ (Kauffman and Levin 1987). For intermediate values of k, the number of local optima increases in k. Thus, k is a parameter that allows us to tune the complexity of the knowledge structure.

In our tests we set $N = 9$, as in our original experiment. We tested 3 different values of k: 0, 1, and 3 (recall that $k = 0$ is our base case so we do it primarily for benchmarking). Evans and Steinsaltz (2002) provide a method for computing the number of local maxima, albeit one that is extremely complicated for $k \geq 2$. From this we have the expected number of local maxima for $k = 1$ is $1.125^9 = 2.887$, and we do not have an estimate of that number in our $k = 3$ trials, although it will be higher still.

Using $k = 1$ and $k = 3$ allows us to model knowledge structures with significant inter-dependence between bit positions leading to several local maxima, albeit ones that still have enough structure to be amenable to a search based on local improvements. To see how the different values of k impact the ruggedness of the preference value function, consider an example with $N = 4$, where we chose $N = 4$ rather than 9 for visual clarity.

With $N = 4$, there are exactly $2^4 = 16$ possible bit strings. First, consider figure 14. Here, the preference function was randomly generated with $N = 4$ and $k = 0$ as described above: each value of $f_i(b^i)$ was drawn from an i.i.d. exponential distribution with unit mean. In this figure, each circle represents one of the 16 possible bit strings, and the height of the circle represents the preference value of the corresponding bit string. One of the 16 bit strings was randomly chosen to represent the belief string of a chosen agent; this is represented by the circle at $X = 0$ (X representing magnitude along the horizontal axis). The X-axis position of the remaining 15 bit strings represents the number of bit positions in which the bit string differs from the chosen agent's belief string. Therefore, for instance, there are exactly 4 circles at $X = 1$: each representing a bit string formed by changing exactly one bit from the chosen agent's belief string.

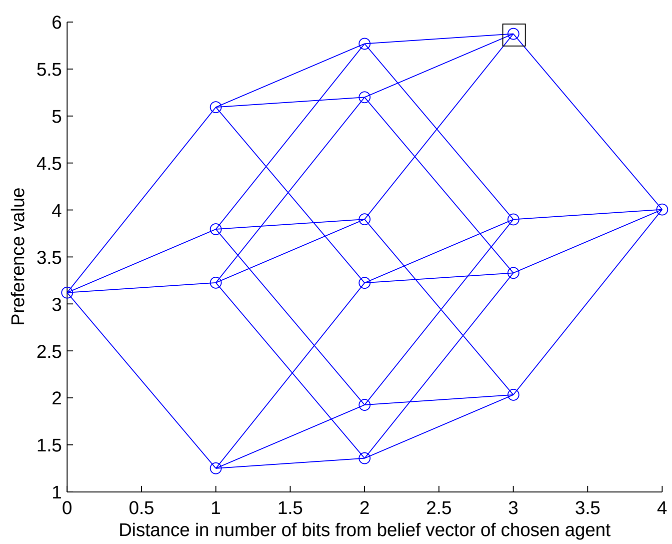


Figure 14: Preference landscape with $N = 4$ and $k = 0$

Two bit strings are connected by a line if they differ in exactly one bit position. This indicates that an agent can move from one circle to another on a line by changing a single bit. The maximum preference value is a little under 6, and this is at distance 3 (i.e. 3 bits need to be changed) from the chosen agent. This maximum is shown by a square around the circle representing the optimal preference belief string.

The salient observations are the following. There is only one maximum element (this is provable when $k = 0$); there are no local optima. Furthermore, any sequence of local improvements from the chosen agent will terminate only upon reaching this global maxima. That is, there is a path from the chosen agent's belief string to the optimal, and there is no possibility of being stuck in a local optimum. This, therefore, captures the case of an "easy" fitness function, where a series of local improvements guarantees convergence to the optimum.

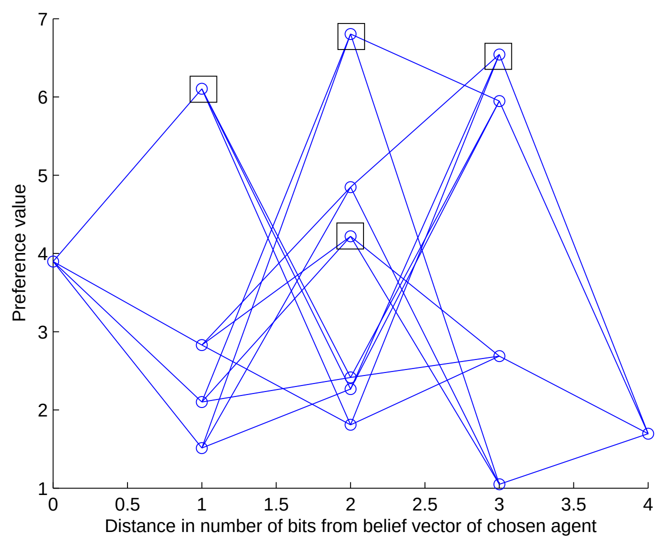


Figure 15: Preference landscape with $N = 4$ and $k = 3$

In contrast, consider figure 15, which represents a similar figure with $N = 4$ and $k = 3$. Observe that now, in addition to the global maximum at distance 2, there are 3 other local maxima. Furthermore, local improvements by the chosen agent are not effective. Local improvements can only lead to a local optimum from which one cannot reach the global optimum via further local improvements. Note, however, that this does not mean that no agent in the network will find the global optimum: if an agent has a belief string that is exactly the opposite of the chosen agent's belief string (and therefore is represented by the circle at $X = 4$ in the graph), then there is a sequence of two bit changes that will enable discovery of the global optimum. The $k = 3$ case with $N = 9$ has even more local optima than in this example, and therefore allows for a large probability of being stuck at local optima.

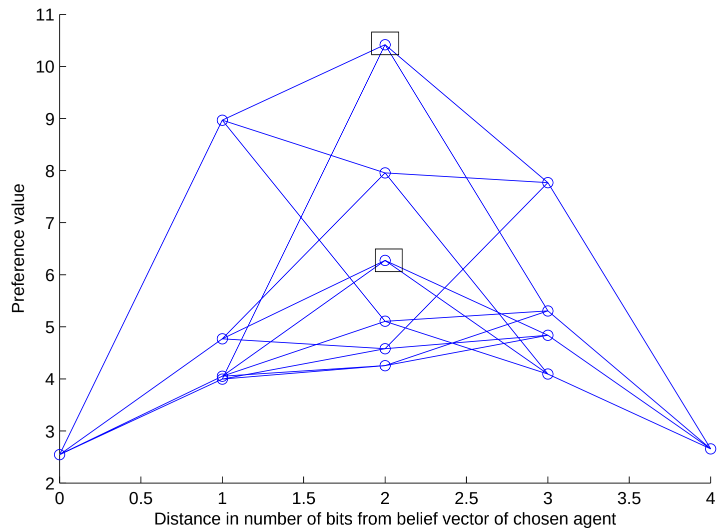


Figure 16: Preference landscape with $N = 4$ and $k = 1$

Finally, consider figure 16, which represents the case of $N = 4$ and $k = 1$. In this case, there is only one local optimum in addition to the global maximum. Additionally, there is a path of local improvements that allow the chosen agent to reach the global maximum (although there is also a path that results in the chosen agent being stuck at the local maximum). This case may be considered to be "in-between" in terms of the preference landscape ruggedness/complexity compared to the two cases above.

Our experiments with these different values of k thus allow us to capture different levels of complexity in the preference function, and explore the impact of such complexity on our findings.

F Analytical investigation of group meetings

In section 10 of the manuscript, we studied the impact of augmenting our simulation with group meetings. We found that with the unimodal preference function, group meetings help poorly performing networks like the star and broom-star, but hurt the well-performing networks like the complete graph and WS30. In this section, we first analytically investigate the impact of group meetings on $TUF(1)$, and verify that our simulation results match it. We then provide a more detailed explanation of exactly why group meetings hurt network performance.

First, let us compute analytically $TUF(1)$ for Complete 10/10, and compare that with the simulation results. Complete 10/10 represents a network of only group meetings. Recall that we start with a belief string where agent i has a 1 in bit position i and 0 everywhere else, and agent 10 has all zeros. Here, $TUF(1)$ is the earliest point of time when exactly 8 successful bit broadcasts have occured. Let agent i be the one who has not yet broadcast bit position i, so that bit position i is 0 for everyone except agent i. Since all other bits have been successfully transmitted, agent i has all 1s and $TUF(1)$ has been reached.

We can compute $E[TUF(1)]$ as follows. Let X_1 be a random variable denoting the time until the first successful bit transmission. Since there are 10 agents, 9 of which have exactly one 1 each, the probability of a successful transmission in any time period is 9/90. So, $E[X_1] = 90/9 = 10$. In general, if X_i denotes the time between the $(i-1)^{th}$ successful transmission and the i^{th} successful transmission, then X_i follows a geometric distribution with success probability $(10-i)/90$, so $E[X_i] = 90/(10-i)$. Since $TUF(1)$ is the time at which 8 successful transmissions have occured, we have $E[TUF(1)] = \sum_{i=1}^{8} \frac{90}{10-i} = 164.6$. The simulation over 1000 trials resulted in a mean of 168 and a standard deviation of 66.4, so a t-test with the null that the two quantities are equal has a two-sided p-value of 0.111. Hence the simulation and analytical computations statistically confirm each other.

Observe that despite $TUF(1)$ for an all-meetings regime being 164.6, the impact of group meetings on the well-performing networks is not felt at the meeting frequencies investigated in the paper. It turns out that group meetings start having a negative impact on performance only when

Meeting regime	Mean $TUF(1)$	% time in meetings
1000/0	98.98	0.00%
100/10	100.14	5.26%
50/10	96.33	11.11%
20/10	102.01	33.33%
18/10	106.57	38.46%
16/10	106.06	45.45%
14/10	113.45	55.56%
12/10	124.71	71.43%
11/10	137.07	83.33%
21/20	149.51	90.91%
10/10	167.96	100.00%

Table 3: $TUF(1)$ for the WS30 network as group meeting frequency increases

they become very frequent. Table 3 shows the impact of increasing meeting frequency on network performance for the WS30 network, and we find that $TUF(1)$ starts degrading significantly only under the 14/10 regime, when more than 50% of the duration of the project is spent in group meetings. So, although we can prove that group meetings hurt performance as they become more frequent, our simulation results in the paper only show that they do not help the performance of well-performing networks.

F.1 Why do group meetings degrade performance?

We now provide a brief explanation of why group meetings result in such a drastic degradation of performance. To do so, we compare the performance of the following two networks: Complete 10/10 (i.e. all group meetings), and Complete 10/0 (i.e., no group meetings). We focus on a single time period, and compare the expected gain in overall network performance in one time period at different times. That is, if $K(t)$ denotes the overall number of 1s in the system, we compute

$E_g[\Delta K(t)] = E_g[K(t) - K(t-1)]$, where the subscript g denotes the group meetings regime. Similarly, we compute $E_b[\Delta K(t)]$ for the bilateral conversations regime. We do these for different values of t.

So, consider the first time period, where $t = 1$ and $K(0) = 9$. In the group meetings case, a successful transmission occurs with probability 9/90, as discussed above. If the transmission is successful, then each of 9 agents gain 1 each, otherwise nobody gains anything. So $E_g[\Delta K(1)] = 9 \times \frac{9}{90} = 0.9$. Next consider the bilateral conversations case. Here, there are 4 pairs of agents with a single 1 in each agent, and one pair consisting of 1 agent with a single 1 and one agent with all zeros. For each of the first four pairs, a successful transmission occurs with probability 2/9, and the gain is 1 if the transmission is successful. For the last pair, a successful transmission occurs with probability 1/9, and the gain is 1 if the transmission is successful. So, $E_b[\Delta K(1)] = 4 \times \frac{2}{9} + 1 \times \frac{1}{9} = 1$, and this is greater than $E_g[\Delta K(1)] = 0.9$. This difference, while small, is not an accidental consequence of having 10 agents but only 9 bit positions. Even with 10 bit positions, the bilateral meetings regime will outperform the group meetings regime. However, this difference is too small to result in the extremely large difference observed in $E[TUF(1)]$. There is something else that causes the extreme performance degradation of the group meetings regime, and that is discussed below.

Consider a time point when $K = 54$: exactly 54 of the 90 bit positions have been set to 1. The number 54 was chosen for computational convenience, as will become apparent below. First, consider the group meetings regime. For K to be 54, the only possible configuration is as follows: 5 bits have been successfully transmitted to all 10 agents, while 4 bits remain to be transmitted. Now consider a single time period. If the chosen agent chooses one of the 5 bit positions that have already been set to 1, this is a useless group meeting because everyone already has a 1 here. The meeting is useful only if one of the 4 untransmitted bit positions are chosen, and the chosen agent actually has a 1 there. So, the probability of a successful transmission is 4/90. If the transmission is successful, there will be 9 agents who gain, so $E[\Delta(K(t)] = 9 \times \frac{4}{90} = 0.4$.

Now consider the bilateral meetings regime, again when $K = 54$. We will assume that the values of two different, randomly selected, agents at any position are independent of each other. This may not be technically accurate, but facilitates a clean intuition that is likely robust to this approximation. If an agent and a bit position is selected at random, the probability of finding a 1 is $54/90 = 0.6$. So, for each bilateral conversation, the probability of a successful transmission is $2 \times 0.6 \times 0.4 = 0.48$. Since there are 5 pairs of agents, we have $E[\Delta K(t)] = 2.4$, which is significantly better than the expected return to a group meeting.

So, at $K = 54$, the bilateral meetings regime is gaining 1s about 6 times as fast as the group meetings regime. The highly correlated belief distribution in the group meetings results in a useless meeting with high probability (either because the transmitted bit is already set to 1 for everyone, or because the transmitted bit is a 0). In general, as K increases, the expected gain under the group meetings regime decreases. The expected gain has a maximum of 0.9 at $t = 0$, and only decreases or stays the same in subsequent time periods. In the bilateral conversations regime, on the other hand, the expected gain is very high for intermediate values of K (a quick calculation shows that under the independence assumption, the expected gain is maximized when half of the bits have been set to 1). It is always at least 1 (when $K = 9$), so it is always more than the expected gain in the group meetings regime. This is why the bilateral meetings regime outperforms group meetings so drastically.