Dynamic Networks and Distributed Problem-Solving

Paul R. Smart[1*], Trung Dong Huynh[1], Dave Braines[2] and Nigel R. Shadbolt[1]

[1] *School of Electronics and Computing Science, University of Southampton, Southampton, SO17 1BJ, UK*
[2] *Emerging Technology Services, IBM United Kingdom Ltd, Hursley Park, Winchester, Hampshire, SO21 2JN, UK*

[*] *Corresponding author: ps02v@ecs.soton.ac.uk*

Abstract—*The effect of dynamic networks on distributed problem-solving was examined using a multi-agent simulation environment. Synthetic agents were tasked with the problem of finding optimal solutions to a specific design problem, and they were allowed to communicate the results of their search efforts to other agents via a dynamically-evolving communication network structure. The growth of the network was determined by two parameters. One parameter determined the rate at which the network structure emerged, while the second determined the point at which the first network link was formed. Together, these parameters produced a reliable effect on collective problem-solving performance. Firstly, performance was negatively affected by the rate of network growth, with faster growth rates producing poorer performance. Secondly, performance was improved by introducing longer initial delay periods into the network formation process, a manipulation which also served to attenuate the decline in performance seen with increasing network growth rates. Of particular interest, the study found that networks with dynamic, constructive topologies delivered a better profile of performance relative to networks with fixed, static topologies. The results are discussed in relation to our understanding of how military coalition communication networks may affect performance outcomes in distributed problem-solving environments.*

1. Introduction

The success of military coalition operations often depends on an ability to adaptively orchestrate the activity of multiple distributed elements in support of some common problem-solving goal or mission objective. Thus, when we think about the execution of military missions, it is clear that the coordinated use of distributed military assets (e.g. weapon systems, military units and sensor platforms) is often a vital element in the successful realization of mission objectives. Similarly, when we think about instances of coalition problem-solving (e.g. planning), it is clear that the coordinated interaction of multiple, distributed agents is a key element of problem-solving success.

Part of what makes the coordinated use of resources difficult in military coalition environments is the distributed nature of the resources themselves. Thus, in the case of coalition problem-solving, individual problem-solving agents may be situated in different locations, and this may limit the extent and/or quality of communicative exchanges. Distributed problem-solving is not, of course, something that we only encounter in military coalition contexts (many organizations feature some form of distributed problem-solving); nevertheless, some features of the military coalition environment may pose particular challenges for distributed problem-solving. For example, the increasing reliance by military coalitions on mobile ad hoc networks (MANETS) and wireless communication technologies may introduce a number of features that affect the dynamics of inter-agent communication (examples include intermittent connectivity due to improved nodal mobility, variable quality connections, network interoperability issues, and dynamic network topologies). Although there have been a number of attempts to understand how specific network structures (e.g. small-world networks) influence distributed problem-solving (see Section 2 for more details), we have, as yet, very little understanding of the precise way in which communication networks with the kind of features seen in military coalition contexts will affect the problem-solving performance of multiple, distributed agents[1].

This paper is an attempt to improve our understanding of how at least some features of the coalition communication environment may affect distributed problem-solving. We report the results of an initial study in which the collective problem-solving performance of synthetic agents was assessed using dynamic, incrementally-constructive networks. Since such networks have highly dynamic topologies, they more closely resemble the type of communication networks seen in military coalition environments. As a result, they may provide some insight into how coalition networks affect the problem-solving performance of distributed agent communities[2].

In addition to reporting the results of studies investigating the effect of dynamic networks on distributed problem-solving, we also discuss a number of directions for future research in this area (see Section 6).

[1] The use of the term 'agents' in this paper is meant to refer to both human and synthetic agents. It is important to bear in mind that although we most commonly think of problem-solving as something that is undertaken by human agents, teams of distributed synthetic agents may also be tasked with specific problem-solving responsibilities. This is particularly true in coalition environments where agent-based capabilities are the focus of considerable research attention.

[2] The use of dynamic networks in the current study is of particular importance because previous studies have tended to restrict their attention to static networks with fixed topologies [1, 2]. As a result, our understanding of the relationship between dynamic network structures and collective problem-solving performance is currently limited.

2. NETWORK STRUCTURE AND DISTRIBUTED PROBLEM-SOLVING

In order to appreciate the effect of physical distribution on coalition problem-solving, it helps to understand something about the way in which communication networks affect the problem-solving performance of both human and synthetic agents. One study which sheds light on this issue is a study by Mason et al [1]. Mason et al [1] examined the ability of groups of networked human subjects (organized into different structural topologies) to collectively explore a problem space and find optimal solutions within that space. What Mason et al [1] found was that the topological organization of the communication network exerted a significant effect on the collective problem-solving ability of the human subjects. Thus, when subjects were confronted with a simple search problem, network structures that supported the most rapid dissemination of information (e.g. totally-connected networks) delivered the most effective profile of performance; however, when subjects were confronted with more difficult search problems, networks that impeded the rate of information dissemination turned out to be the most effective.

Results similar to these were reported by Lazer and Friedman [2], this time using synthetic agents. As with Mason et al [1], Lazer and Friedman [2] examined the effect of network structure on collective problem-solving performance using a collective search task. However, the search task used by Lazer and Friedman [2] was based on the search for optimal design solutions using a variant of the NK model paradigm, as developed by the evolutionary theorist Stuart Kauffman [3, 4]. An NK model is essentially a means of generating evolutionary fitness landscapes with different degrees of 'ruggedness'. By using a particular type of NK model, Lazer and Friedman [2] were able to set up a problem-solving task in which synthetic agents had to explore a moderately rugged fitness landscape (comprising about 100 local optima) and discover optimal solutions within that landscape [see 5 for more details]. As part of the task, agents were each assigned a 20-bit solution string (consisting of 20 binary values of either '0' or '1'), and their goal was to find more optimal (or fitter) design solutions by progressively modifying the individual elements of the solution string. The fitness landscape generated by the NK model determined the relative fitness of different 20-bit solutions, so the process of modifying the binary elements of the 20-bit solution string was essentially akin to the exploration of a fitness landscape. Lazer and Friedman [2] allowed the agents in their study to communicate the results of their search to other agents by organizing them into communication networks with different topologies.

What Lazer and Friedman [2] found was an apparent trade-off between what they refer to as 'exploration' and 'exploitation'. Exploration is the tendency of agents to explore the solution space independently of other agents, and exploitation is the tendency of agents to adopt the solutions proposed by other agents. The trade-off between exploration and exploitation is revealed by a profile of topology-dependent performance in which networks with low average path lengths (e.g. totally-connected networks) yield better performance outcomes in the short-term (compared to networks with higher average path lengths (e.g., linear networks)), but worse performance in the longer term (again compared to networks with higher average path lengths).

The results of both Mason et al [1] and Lazer and Friedman [2] point to a common conclusion regarding the effect of network structure on distributed problem-solving performance. It is that different types of network topology can effectively influence the rate at which information is disseminated within a group of problem-solving agents, and this can, in turn, affect the ability of the group to discover globally-optimal solutions.

The studies of Mason et al [1] and Lazer and Friedman [2] both focus on the use of networks with fixed, static structural topologies. However, there is no reason to assume that such networks necessarily exhaust the space of cognitively-interesting network simulations, or even that such networks necessarily represent the kind of networks typically encountered in most cases of real-world distributed cognitive processing. As mentioned in Section 1, the kind of communication networks encountered in military coalition contexts are likely to be ones with highly dynamic structural topologies, and these kinds of networks are clearly unlike those seen in the Mason et al [1] and Lazer and Friedman [2] studies. Furthermore, there are a number of reasons to think that dynamic networks may have a number of specific virtues when it comes to collective cognitive processing. Firstly, such networks support the temporally-specific coupling of various resources into flexibly-configured and dynamically-bounded cognitive systems. Thus, instead of searching for a network structure that is uniformly beneficial for all forms of distributed problem-solving, dynamic networks enable us to think in terms of adaptively configured networks that strategically modify their connectivity at specific junctures in the problem-solving process in order to best meliorate problem-solving performance. Secondly, there is evidence from the neuro-developmental, neural network and developmental psychology literatures that suggests dynamic networks may have specific cognitive virtues when compared to their more statically-configured network counterparts [see 5 for a review]. All of this highlights the need to perform empirical studies that investigate the effect of dynamic networks on the collective problem-solving performance of physically-distributed agents.

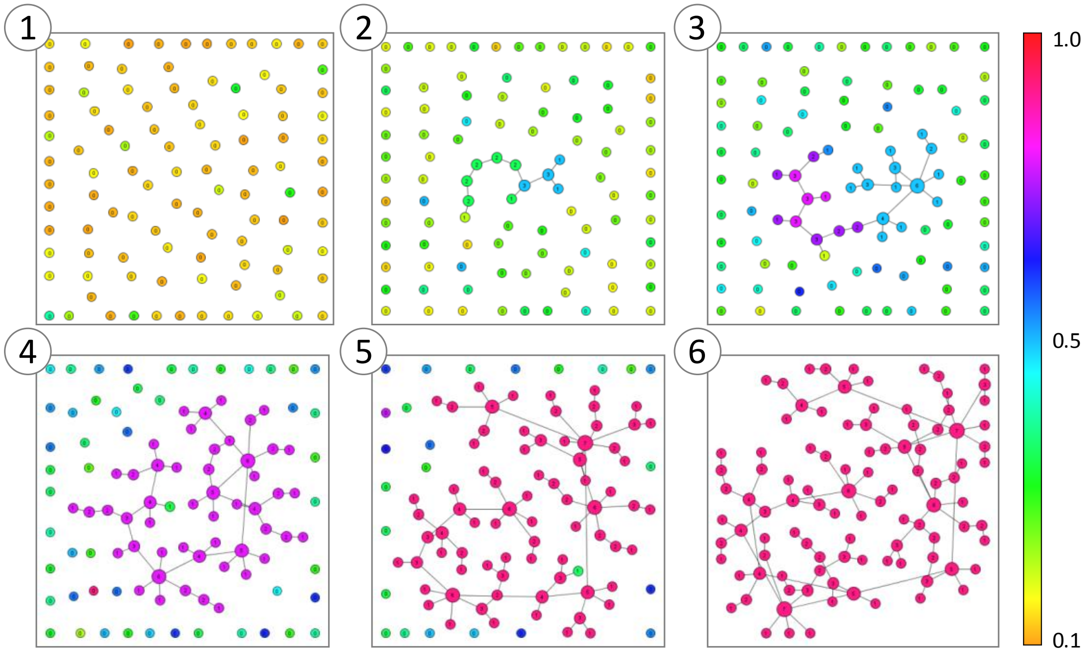

Figure 1. Figure showing the progressive addition of links during the early stages of a simulation. The graphics numbered 1 to 6 show the 100 agents (represented as nodes) at the outset of the simulation (tile 1) when none of the agents are connected, and at the 100th processing cycle (tile 6) when all the agents are connected into a single network component. The colour coding of the nodes in these tiles indicates the value of the solution associated with each agent, with orange/yellow values indicating poor solutions and purple/red values indicating good solutions. As can be seen from the figure, the agents in the simulation gradually discover progressively better solutions throughout the course of the simulation..

3. METHOD

3.1. Experiment 1: Dynamic Networks and Collective Problem-Solving Performance

In order to investigate the effect of dynamic networks on collective problem-solving performance, a series of computer simulations were undertaken using dynamic constructive networks[3]. The simulations relied on the same problem-solving paradigm as that used by Lazer and Friedman [2], and the NK models were of the same complexity; i.e. all simulations used NK models with parameters of N=20 and K=5[4]. Before running the simulations, 1000 different NK models were generated using the same code as that used by Lazer and Friedman [2][5], and the models were then used in simulations where a collection of 100 agents were tasked with the exploration of the NK fitness landscapes and the discovery of optimal solutions (peaks) within those landscapes. As with the Lazer and Friedman [2] study, problem-solving in this experiment consisted of a form of collective search. Thus, agents were initially assigned 20-bit solution strings (consisting of 20 binary values) at the outset of the simulation, and they then had to search for the most optimal solution string (as determined by a specific NK model[6]) by randomly selecting and flipping individual values of the solution string throughout the course of each simulation. Information about the search efforts of other agents in the simulation was provided by the structure of the communication network, so as the network structure emerged across the course of each simulation, agents were progressively able to share more and more information about the results of their search efforts.

The experiment involved the manipulation of two independent variables: Network Growth Rate (NGR) (which represents the rate at which links were added to the network) and Network Growth Delay Period (NGDP) (which represents the period that elapsed before the first link was added to the network). In total, there were 7 levels of the NGR variable and 6 levels of the NGDP variable, giving a total of (7*6) 42 experimental conditions (a two-way factorial experimental design was used). Within each condition, the 100 agents were tested against the same set of 1000 NK models (i.e. the same set of 1000 NK models were used for each condition), resulting in a total of 1000 simulations for each condition (i.e. a total of 1000*42 = 42,000 simulations).

At the start of each simulation, each of the 100 agents was assigned to a particular solution within the NK model space

[3] A dynamic constructive network is a network whose structural connectivity emerges across the course of a particular simulation. Thus, at the outset of a simulation, no linkages exist between the nodes; rather, the linkages are added progressively over the course of a simulation until the full structure of the network is realized.

[4] These parameters yield fitness landscapes that are moderately rugged, with a few hundred local optima with high correlations between proximate solutions. As Lazer and Friedman [2] comment, these kind of problem spaces probably best capture "the essence of most interesting problems that individuals and organizations in the real world face—rugged, but not chaotic" (pg. 674).

[5] See http://sites.google.com/site/parallelproblemsolving/.

[6] The value (or 'fitness') of each particular solution is determined by whatever NK model is being explored during each simulation. Thus, within each experimental condition, the same solution could be evaluated very differently in different simulations. This follows from the fact that each simulation was run against a different NK model.

selected for the simulation. Agents were assigned the same value for each NK model, thus the starting point for each agent in each NK model across all experimental conditions was the same. The simulation then consisted of a number of processing cycles during which the following steps were applied to each agent (X):

1. the current solution (S) for X was compared to all the solutions of X's immediate neighbours in the network (if they had any)
2. if the solution of a neighbouring agent was found to be better than S, then X adopted the solution of its neighbour
3. if none of the solutions of X's neighbours were better than S, then X modified S (by randomly selecting and flipping one of the 20 binary values comprising S) to generate a new solution (S')
4. if S' was better than S, then X adopted S' as its current solution; otherwise, X adopted S as its current solution

Each simulation was then run until at least one of the following termination conditions had been reached:

1. 1500 processing cycles had elapsed, or
2. all possible links had been added to the network and the network was thus fully (or totally) connected, or
3. all the agents were connected into a single network component, they had all converged on a common solution, and at least 20 processing cycles had elapsed with no new solutions being generated by any of the agents[7].

Within each simulation, the structure of the network determined the profile of information sharing between the agents. However, at the outset of the simulation, none of the agents were connected (i.e. there were no links between any of the agents). Throughout the course of each simulation links were added in order to connect the agents together into networks of increasing density (see Figure 1), and the rate at which these networks emerged was determined by the specific values of the NGR and NGDP variables. As is indicated by Figure 1, the networks in this study were generated in a particular way. Links were added to the agent community by first selecting an agent at random from the total population of 100 agents. A second agent was then selected (again at random) from the subset of agents that were already connected to at least one other agent (i.e. the agent possessed at least one link). The two selected agents were then connected via the addition of a new link[8]. Once all unconnected agents had been integrated into a single network component, a link was added at random to the emerging network structure. The network growth law in this study is thus a form of (initial) preferential attachment (of unconnected nodes to connected ones), but there was no bias in favour of connecting to nodes that had a particular number of links (e.g. the most links)[9].

The rate at which links were added to the emerging network depended on the value of the NGR variable for the specific experimental condition, and the cycle at which the first link (or set of links) was added depended on the value of the NGDP variable. Thus, if the NGR variable was set to 1.0 and the NGDP variable was set to 50, then the first link would be added on the 50th cycle of the simulation, and 1 link would be added to the 'network' every cycle thereafter. The levels of the NGR variable used in this study were 0.1 (1 link added every 10 cycles), 0.2 (1 link added every 5 cycles), 0.5 (1 link added every 2 cycles), 1 (1 link added every cycle), 2 (2 links added every cycle), 5 (5 links added every cycle) and 10 (10 links added every cycle). The levels of the NGDP variable were 0 (no delay period), 10 (a delay period of 10 cycles), 20 (a delay period of 20 cycles), 30 (a delay period of 30 cycles), 40 (a delay period of 40 cycles) and 50 (a delay period of 50 cycles).

3.2. Experiment 2: Dynamic vs. Static Networks

In order to compare the performance of dynamic and static networks, an additional series of simulations was undertaken using static networks. A single network was generated by adding links to a population of 100 agents until a single network component had formed. The procedure used to generate the static network was identical to that used to create the dynamic networks in Experiment 1; i.e. unconnected agents were selected at random and then randomly connected to agents with existing links. This meant a single network component emerged after the addition of only 100 links.

The performance of agents within this static network was assessed by running 1000 simulations using the same NK models as those used for Experiment 1. The procedure for generating new solutions and copying superior solutions from network neighbours was exactly the same as that used for the dynamic network simulations, and the termination conditions for each simulation were also the same. Once the simulations were completed, the collective problem-solving performance of agents in the static network condition were compared with the performance of those seen in the dynamic network conditions.

[7] On each processing cycle, an agent randomly selects one element of the 20-bit solution string with which it is associated. This means that over the course of a 20-cycle period an individual agent may fail to completely explore all neighbouring solutions (i.e. they may fail to select some elements of the 20-bit solution string). However, because termination condition 3 is based on the fact that we are considering the solutions generated by *all* the agents (and all the agents have converged on a common solution), then the chance that the agents will *collectively* fail to explore all neighbouring solutions is very small (i.e. 2.8 E-45).

[8] This is why, in Figure 1, we see the emergence of a single network component after the addition of only 100 links.

[9] Clearly, there a number of ways in which the agent network could be generated. These alternative growth laws are the subject of ongoing empirical investigations (see Section 6.3).

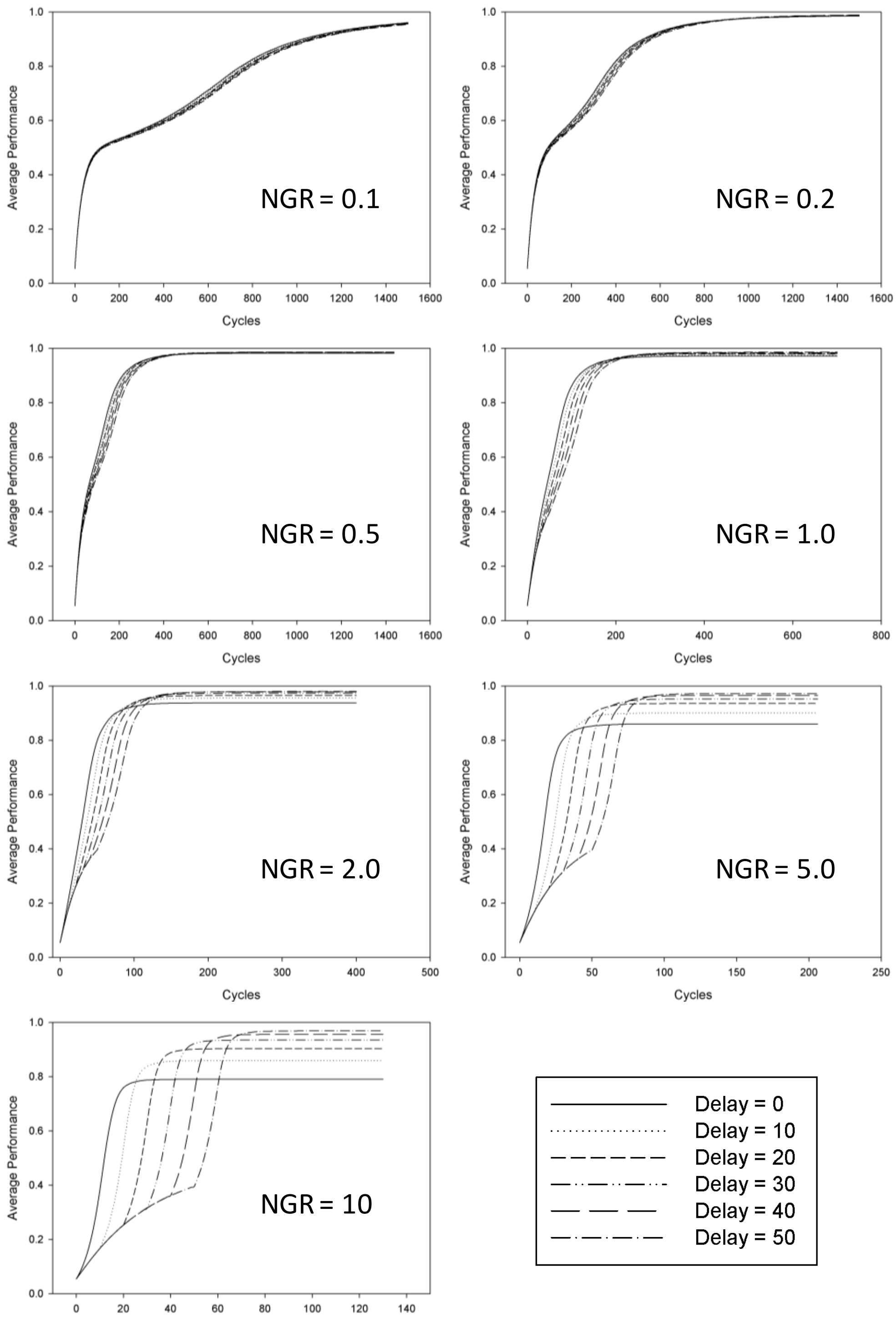

Figure 2. Figure showing the average performance over 1000 simulations in networks with increasing growth rates. The lines on each chart show the effect of the initial delay period, with longer delays causing a rightward shift in the performance curves.

4. Results

4.1. Experiment 1: Dynamic Networks and Collective Problem-Solving Performance

Figure 2 summarizes the results obtained from Experiment 1. The figure shows the average performance of agents over the course of successive processing cycles for different levels of the NGR and NGDP variables. The average performance for each cycle is calculated as the average fitness score associated with the solutions adopted by all agents across all simulations (the fitness score is thus averaged across all the agents within a particular cycle (100 data points) and across all the simulations within a particular treatment condition (1000 data points)).

What Figure 2 appears to show is that as the rate of network growth increases (i.e. as the rate at which links are added to the network increases), the length of the simulation decreases (i.e. the number of cycles that must elapse before one of the termination conditions is encountered is reduced). However, as the rate at which links are added is increased, the final performance of the agents (the quality of the final solution) also becomes more variable, with performance apparently negatively affected in conditions involving both high growth rates and shorter delay intervals. The effect of the initial delay interval, therefore, seems to be that it offsets a growth rate-related decline in collective problem-solving performance. Furthermore, at each level of the NGR variable, the effect of increasing the initial delay period seems to be a shift in the performance curve to the right, particularly during the middle part of the simulation. Thus, for cycles in the middle of a simulation, the effect of the delay interval is to reduce the quality of the solutions found by agents; however, by the end of the simulation this performance deficit is eliminated. In fact, for the higher growth rate conditions, the slight performance deficit seen during the middle of the simulation is reversed at the conclusion of the simulation, with higher levels of performance being seen in the conditions involving longer initial delay periods.

These observations are backed up by the statistical analysis of the experimental data. Firstly, a two-way (7*6) between subjects factorial ANOVA on the performance scores associated with solutions at the end of the simulations revealed a significant interaction between the NGR and NGDP variables ($F_{(30,\ 41958)} = 148.629$, $P < 0.001$) and significant main effects for both the NGR ($F_{(6,\ 41958)} = 1218.754$, $P < 0.001$) and NGDP variables ($F_{(5,\ 41958)} = 538.756$, $P < 0.001$). Figure 3 illustrates the average final performance score obtained in the various experimental conditions. As can be seen from the figure, an increase in NGR results in a progressive deterioration of performance, and this deterioration seems to be most pronounced for conditions involving shorter initial delay periods. An analysis of simple main effects at the levels of the NGR variable revealed that differences between the various delay period conditions begins to emerge at growth rates of 0.5 links per cycle and above (i.e. NGR = 0.5). Thus the performance differences between NGDP conditions at the 0.5 level of the NGR variable were statistically significant ($F_{(5,\ 41958)} = 3.091$, $P < 0.01$), but below this level (i.e. growth rates of less than 0.5 links per cycle) there were no significant differences between the NGDP conditions.

Besides the ability of higher initial delay periods to attenuate a growth rate-related decline in final performance scores, Figure 3 also suggests that performance was negatively affected by the slowest rate of network growth (i.e. NGR = 0.1). A one-way ANOVA comparing the performance data for various levels of the NGR variable revealed a significant difference between the means of the NGR group ($F_{(7,\ 41993)} = 962213.491$, $P < 0.001$), and post hoc comparisons using Tukey's Honestly Significant Difference (HSD) test revealed that performance was worse at the 0.1 level of the NGR variable when compared to intermediate levels of the NGR variable (i.e. growth rates

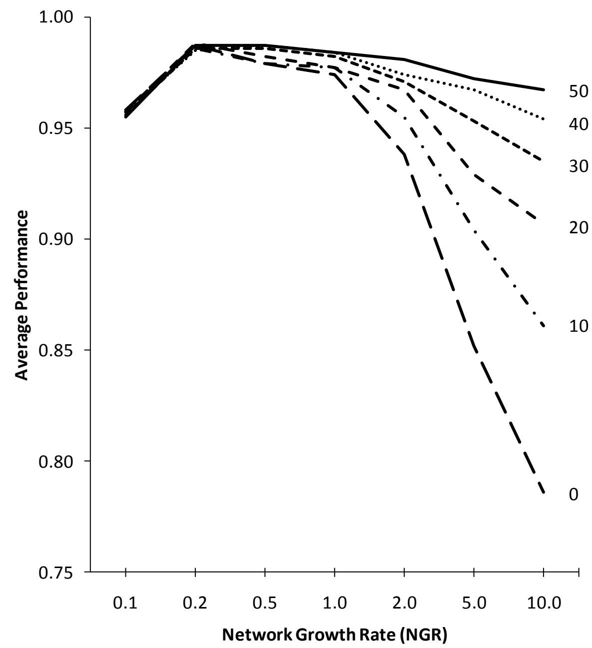

Figure 3. Figure showing the performance scores for all experimental conditions at the end of the simulations. The numbers next to the lines on the chart represent the levels of the NGDP variable.

between 0.2 and 2.0). The reason for this performance deficit at very slow rates of network growth is probably attributable to the termination conditions established for the current study. One of the termination conditions, recall, concerned the maximum number of processing cycles that could be run in a single simulation. At very slow rates of growth it appears that the agent community did not have sufficient time to reach the same levels of performance as that seen at intermediate rates of growth. This interpretation is supported by an inspection of the performance data for the NGR = 0.1 condition (see Figure 2). From Figure 2 we can see that by the 1500^{th} cycle the performance of agents had not quite reached the same levels as those observed in the NGR = 0.2 and NGR = 0.5 conditions. The relatively poor performance of agents in the slowest growth rate condition therefore appears to be an artefact of the specific termination conditions established for the current study.

4.2. Experiment 2: Dynamic vs. Static Networks

Figure 4 shows the average performance of agents across successive processing cycles in the static network condition. Two things are immediately apparent from this performance profile. The first is that the agents settle on a stable solution very quickly – after about only 15 cycles. This contrasts with the results seen in dynamic networks where it typically took much longer for the agents in dynamic network simulations to converge on a common, stable solution. The second, and perhaps more notable result, however, concerns the final performance scores. In the static network condition, the final performance of the agents reached a value considerably below that seen with dynamic networks, even when compared with the highest growth rate condition

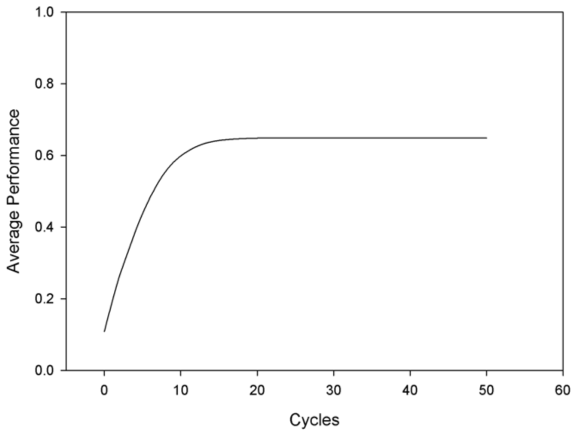

Figure 4. Average performance over 1000 simulations in a network with a fixed topology.

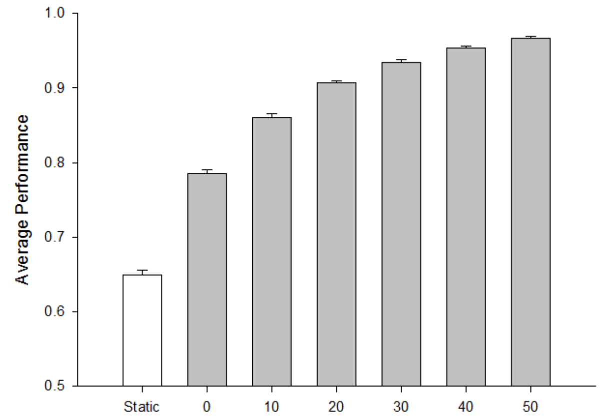

Figure 5. Graph showing the final performance scores associated with static and dynamic networks. The shaded bars represent conditions featuring dynamic network conditions at various levels of the NGDP variable (i.e. delay periods of 0, 10, 20, 30, 40 and 50 cycles). All dynamic networks had a growth rate of 10 (i.e. 10 links were added every processing cycle). Post hoc comparisons revealed that all dynamic network conditions outperformed the static network condition ($P < 0.001$).

(i.e. 10 links added every cycle)[10]. The results of a one-way between subjects ANOVA on the performance data for the static network condition and the performance data obtained in the highest NGR condition (i.e. NGR = 10) revealed a significant difference between the various conditions ($F_{(7, 6993)} = 48537.010$, $P < 0.001$). Post hoc comparisons between the conditions using Tukey's HSD test revealed that all dynamic networks outperformed the static network (see Figure 5). We therefore arrive at the conclusion that, at least in this task context, problem-solving performance is enhanced in networks with dynamic, incremental topologies, relative to networks with fixed, static topologies.

5. Discussion

Accounting for the specific pattern results obtained in the current study requires us to think about the nature of the task confronting agents in the simulation. Perhaps the best way to think of this task is to see the agents as like explorers parachuted into a mountainous landscape at night, equipped only with radios, flashlights and altimeters. Imagine that the goal of each explorer is to find the highest peak in the landscape, and thus all explorers begin to move uphill as soon as they touchdown. As they move ever upward, the explorers can communicate with each other using their radios, reporting the results of their altimeter readings. And if agents hear that one of their colleagues has found a higher point on the landscape, then they all converge on that location and begin searching from that particular point.

We can now begin to see how a particular pattern of time-variant changes in communication network structure will affect the opportunities that agents have for independent exploration of the solution space (i.e. the fitness landscape). Given what we know about the way in which agents are influenced by the superior solutions of connected neighbours (agents simply adopt the superior solutions of their neighbours), it becomes apparent that the more time agents have to independently explore the solution space, the more likely the community is, as a whole, to discover the global optimum. When agents are connected together in a densely connected network, the rate of information dissemination is very high, and thus agents will be inclined to prematurely settle on the highest value solution found at the very outset of the exploration process. The situation is a bit like enabling full two-way radio communication between all the aforementioned explorers as soon they touchdown: all explorers will rapidly converge on the same location as the explorer who just happens to land at the highest point in the landscape. However, given the rugged nature of the terrain, the highest *initial* location is not necessarily the one that leads directly to the highest peak. If one explorer lands on the top of a small hill, and all the other explorers land at the base of a separate, much taller hill, then all the explorers will immediately move to the top of the smaller hill and remain there; they will fail to discover the most optimal solution (the peak of the tall hill) because the explorer atop the smaller hill will broadcast a higher altimeter reading to all other explorers and all explorers will then immediately converge on that location. When initial communication between the explorers is more limited, each explorer is given more time to independently explore the local terrain. This means that initially weak solutions that lie on the path to more optimal solutions are not prematurely rejected in favour of initially fitter, but globally sub-optimal solutions.

We can now see why, in the case of dynamic networks, agents are more likely to prematurely settle on a globally sub-optimal solution in the high growth rate conditions: the presence of more and more links across successive processing cycles progressively increases the rate at which information is transmitted between the agents. It also explains why, in general, we see a better profile of performance in dynamic networks compared to static networks (at least the static networks that were created using the method described here). In dynamic networks, the extent of inter-agent influence is initially very limited; each agent can only influence those agents to which they are directly or

[10] The average performance of the dynamic networks used in this study is also greater than the performance of the single component random networks seen in Lazer and Friedman [2].

indirectly connected, and, at least initially, all agents begin with minimal influence (i.e. no two agents are connected). This poverty of influence, relative to the situation with more densely connected networks, means that each agent has time to undertake a local exploration of the solution space before reporting the results of this local search to all other agents (when a single network component eventually emerges). Incrementally constructive networks therefore strike a productive balance between autonomy and influence; they give each agent the freedom to search for locally-optimal solutions without sacrificing the (eventual) benefits of collective search.

Two additional phenomena from the aforementioned study now require explanation. One is the tendency for increasing initial delay periods to negatively affect performance at the mid-stage of the simulation (recall that the performance curve is shifted to the right in most of the high delay period conditions). The other is the tendency for increases in the initial delay period to counteract the deleterious effect of high growth rates on the quality of final solutions. The former phenomenon probably stems from the fact that with high initial delay periods agents have greater initial autonomy in exploring the solution space. This means they can exhaustively search for locally-optimal solutions without being influenced by the search results of other agents. Unfortunately, however, this freedom comes at a price because the *average* performance of a set of disconnected agents will always be worse than the *average* performance of a set of interconnected agents. When agents are connected, they can share information about the best solution currently on offer, which means that all agents can converge on the best solution. When agents are disconnected, each can only progress as far as the nearest local optimum. So even if one agent is lucky enough to find the global optimum, the *average* performance of the community will still be relatively low.

In respect of the second phenomenon (the fact that longer initial delay periods attenuate a growth-related decline in performance), the longer initial delay period supports the greater initial autonomy of agents and thus enables them to explore more of the solution space before they converge on a common solution. When no delay period exists, agents quickly become interconnected (particularly when the growth rate is very high) with the result that they are inclined to prematurely settle on a solution that, in all likelihood, is sub-optimal relative to the kind of solution that they could have found if they had been allowed to exhaustively search their local part of the solution space.

6. Future Work

The results of the current study suggest that dynamic networks may deliver performance benefits, relative to their static network counterparts, in at least some task contexts. The results of the current study are, however, limited in a number of ways, particularly when it comes to a consideration of the way in which coalition communication environments may affect distributed problem-solving in multi-national coalition contexts. In this section, we present a number of directions for future research which may help to address these shortcomings and expand the overall generalizability of the research results[11].

6.1. Going Beyond Collective Search

One limitation of the current study concerns its focus on a specific type of distributed problem-solving, namely collective search. Many types of real world collaborative problem-solving involve significant amounts of specialization in which agents work on particular parts of a problem and then attempt to coordinate their activities with respect to some larger, overarching problem-solving goal. It seems unlikely that the results presented here can generalize to these more differentiated and hierarchically-structured tasks because such tasks feature complex inter-dependencies between the behaviours of particular agents. In the case of the search task used here, the success of one particular agent has no effect on the success of any other agent, and thus the solutions proposed by one agent are independent of all the others. However, some of the most interesting cases of real-world collaborative problem-solving, such as military planning, involve situations where the suitability of candidate solutions as proposed by one agent are heavily dependent on the solutions proposed by other agents.

Clearly, what is required here in terms of future research is the development of simulation capabilities that are based on other types of problem-solving activity, particularly those involving some degree of task specialization and agent interdependence. The ultimate aim of this strand of work is to understand how the features of specific tasks relate to the effect of network-level variables on collective problem-solving.

6.2. The Brighter Side of Distrust

The current research is concerned with the way in which network structure affects the opportunities that agents have to exchange information and influence one another. However, in human social networks it should be clear that a variety of psychosocial factors may conspire to influence the actual nature of information flow and influence. Trust, for example, is likely to influence the *effective* rate at which information propagates through a social network, with low trust levels negatively affecting the rate of information propagation. Inasmuch as inter-agent trust is something that effectively regulates the extent of information flow and

[11] Relative to the goal of understanding how the features of the coalition communication environment may affect collective problem-solving, the current research may be criticized on the grounds that 1) it does not adequately model the agents involved in coalition-based problem-solving, 2) the tasks in which such agents engage, or 3) the networks that support inter-agent communication. Actually, however, the current research does not seek to shed light solely on human-based problem-solving; it recognizes that many problem-solving processes in coalition environments may be undertaken by synthetic as well as human agents. In this case, the use of synthetic agents in the current study is perfectly legitimate, although we do recognize that the nature of the problem-solving processes in many coalition contexts is likely to be more knowledge-intensive than the kind of task used in the current study.

influence in a problem-solving community, it is possible that both trust and distrust may, at different times, play adaptive roles in enabling a community of problem-solving agents to reach optimal solution outcomes. Smart et al [5] thus advance the idea that there may be a brighter side to distrust when it comes to a consideration of socially-distributed cognition:

> *"Initial levels of distrust, reflecting perhaps the initial caution people bring to new social situations, are not necessarily to be regarded as maladaptive when it comes to collective problem-solving. Sometimes distrust may play an adaptive role in configuring the functional connectivity of a network in a way that best meliorates collective cognitive processing."*

Clearly, given the considerable research interest in trust relationships, particularly in the context of coalition operations, there is a strong basis for undertaking further work that specifically addresses the role of trust in enabling (and perhaps disabling) distributed problem-solving. The general technical approach for this strand of work would, in all likelihood, consist in the development of computational models of inter-agent trust (preferably ones in which agents dynamically update their trust estimates in light of previous interactions and other information), followed by an analysis of how (dynamic) trust relationships contribute to the collective performance of agent teams in different problem-solving situations.

6.3. Constructive Algorithms

The communication networks in the current study were generated using a particular type of constructive algorithm, namely one that relied on the random addition of links to the emerging network. Obviously, this is not the only kind of constructive algorithm that could be used. Thus, networks could be created by relying on a variety of preferential attachment laws (e.g. preferential attachment to nodes with the most links), or the networks could be configured based on some desired topological outcome (e.g. links could be added to yield networks with linear or small-world topologies). The impact of these alternative constructive algorithms on distributed problem-solving performance is the subject of ongoing investigations in our laboratory.

6.4. Adaptive Coupling

Constructive algorithms are, of course, not the only way to produce dynamic networks. A dynamic network is essentially any network whose functional (or effective) connectivity changes throughout the course of some period of distributed information processing, and there are clearly a number of ways in the functional connectivity of a network can change. A network may, for example, begin as a totally-connected network and then progressively lose linkages in order to assume some other form of topological organization. Alternatively, some linkages may become periodically active or inactive, thereby contributing to complex time-variant patterns of network-mediated information flow and influence. Not only are these kinds of networks closer approximations to the kind of networks we encounter in military coalition (and other real-world) settings, they are also the kind of networks that Smart et al [5] have in mind when they present the thesis of adaptive coupling:

> *Adaptive Coupling Thesis: In situations where cognitive outcomes depend on the coordinated activity of multiple resources, cognitive performance will benefit from the ability to dynamically and flexibly couple those resources into transient networks of information flow and influence. Dynamic networks support the realization of multiple time-variant patterns of functional connectivity, and these enable the component resources to adaptively coordinate their activity at critical junctures in a collective problem-solving process.*

The notion of adaptive coupling is thus based on the idea that networks should support the time-dependent coupling of distributed resources into highly configurable nexuses of information flow and influence. This process could, at least in some situations, be supported by the kind of incrementally-constructive networks described in this chapter, but it is perhaps more likely that most instances of collective cognitive processing will rely on networks whose functional connectivity is free to vary in any number of ways. In order to begin the empirical evaluation of the adaptive coupling thesis, it is important that future studies should consider networks that are capable of undergoing all manner of structural changes throughout the course of distributed problem-solving.

6.5. MANETs

If we are to generalize the results of empirical studies such as those reported here to military coalition environments, it will be important to focus not just on the features of the agents involved in the problem-solving process (e.g. their tendency to form variable trust relationships – see Section 6.2), the features of the problem-solving processes in which they engage (e.g. the type of tasks they perform – see Section 6.1), or the way in which the communication network topology changes throughout the problem-solving process (see Section 6.4), it will also be important to consider the features of the communication environment itself. In particular, given that military coalitions are likely to be making increasing use of wireless MANET technology in the near future [e.g. 6], it is important to understand how the specific features of MANETS will affect the collective cognitive capabilities of coalition teams. Research in this area is best supported by developing network models that incorporate the features of MANET-based communication environments.

6.6. Network Structure and Shared Interpretation

The current paper focuses on a particular form of collective cognition, namely distributed problem-solving. However, collective cognition can assume a variety of forms, including the attempt by multiple agents to reach a consensus regarding the interpretation of ambiguous information [see 5]. In one study, for example, the cognitive anthropologist, Edwin Hutchins, investigated the effect of inter-agent communication on the ability of agents to arrive at an accurate shared interpretation of ambiguous environmental information. What Hutchins [7] discovered was that early forms of interaction led to a situation of confirmation bias in which agents failed to give due weight to information that conflicted with their initial interpretations of some external state-of-affairs. Such work has potential implications for our understanding of how a network affects a group's ability to deal with poor quality information, for example, information that is ambiguous, inaccurate, inconsistent or incomplete. In thinking about possible extensions to the current body of work, it will no doubt be important to reflect on the relevance of Hutchins' [7] work to our notions of both shared situation awareness [8] and shared understanding [9, 10]. It will also be important to assess whether factors like time-variant trust evaluations and dynamic network topologies deliver the same profile of performance benefits in the case of shared interpretations as they do in other cases of collective problem-solving.

6.7. Hybrid Networks

When we think about distributed problem-solving, it is common to think about networks comprised of intelligent problem-solving agents (i.e. human or synthetic agents). In this situation, each node in the network is an agent who is capable of receiving, processing and communicating information. In the real world, however, intelligent agents seldom act in isolation from their environment. Their individual information processing loops may extend across a rich array of material props, aids and artefacts, resulting in a variety of forms of extended cognitive system [11]. The point for present purposes is simply that most real-world cases of distributed cognition involve networks that consist of a variety of disparate resources, (e.g. human agents, synthetic agents, sensors, multiple types of information resources, software applications, and so on), and we have, as yet, very little understanding of how these kinds of networks (which we refer to as hybrid networks) can best be organized so as to support cognitive processing at both the individual and collective levels. Since dynamic hybrid networks would seem to resemble the kind of networks that we typically encounter in military coalition environments, such networks are an obvious and interesting target for future empirical studies.

7. Conclusion

Problem-solving in military coalition contexts often relies on the interaction of multiple, distributed agents who communicate with one another via one or more coalition communication networks. In order to begin to understand the effect of such networks on distributed problem-solving performance, we investigated the effect of dynamic, incrementally-constructive networks on a collective search task using populations of synthetic agents. Our results suggest that, at least in some task contexts, dynamic networks may yield performance benefits relative to their statically-configured network counterparts. Although the generalizability of these results is somewhat limited at the present time, the results do raise a number of interesting issues regarding the best way to configure the coalition communication environment so as to best support distributed problem-solving processes.

Acknowledgment

This research was sponsored by the US Army Research laboratory and the UK Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the U.S. Government, the UK Ministry of Defence, or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

[1] W. A. Mason, A. Jones, and R. L. Goldstone, "Propagation of innovations in networked groups," in *27th Annual Conference of the Cognitive Science Society*, Stresa, Italy, 2005.

[2] D. Lazer and A. Friedman, "The Network Structure of Exploration and Exploitation," *Administrative Science Quarterly*, vol. 52, pp. 667-694, 2007.

[3] S. Kauffman, *At Home in the Universe: The Search for Laws of Self-Organization and Complexity*. London, England: Penguin Books Ltd, 1995.

[4] S. Kauffman, *The Origins of Order: Self-Organization and Selection in Evolution*. New York, USA: Oxford University Press, 1993.

[5] P. R. Smart, T. D. Huynh, D. Braines, K. Sycara, and N. R. Shadbolt, "Collective Cognition: Exploring the Dynamics of Belief Propagation and Collective Problem Solving in Multi-Agent Systems," in *1st ITA Workshop on Network-Enabled Cognition: The Contribution of Social and Technological Networks to Human Cognition*, Maryland, USA, 2010.

[6] C.-K. Chau, J. Crowcroft, K.-W. Lee, and S. H. Y. Wong, "Inter-Domain Routing in Mobile Ad Hoc Networks: Challenges and New Directions," in *Network Science for Military Coalition Operations: Information Extraction and Interaction*, D. Verma, Ed. Hershey, Pennsylvania, USA.: IGI Global, 2010.

[7] E. Hutchins, *Cognition in the Wild*. Cambridge, Massachusetts: MIT Press, 1995.

[8] A. A. Nofi, "Defining and Measuring Shared Situational Awareness," Center for Naval Analyses, Alexandria, Virginia, USA CRM D0002895.A1/Final, 2000.

[9] P. R. Smart, T. D. Huynh, D. Mott, K. Sycara, D. Braines, M. Strub, W. R. Sieck, and N. R. Shadbolt, "Towards an Understanding of Shared Understanding in Military Coalition Contexts," in *3rd Annual Conference of the International Technology Alliance (ACITA'09)*, Maryland, USA, 2009.

[10] P. R. Smart, D. Mott, K. Sycara, D. Braines, M. Strub, and N. R. Shadbolt, "Shared Understanding within Military Coalitions: A Definition and Review of Research Challenges," in *Knowledge Systems for Coalition Operations (KSCO'09)*, Southampton, UK, 2009.

[11] R. A. Wilson and A. Clark, "Situated Cognition: Letting Nature Take its Course," in *Cambridge Handbook of Situated Cognition*, M. Ayded and P. Robbins, Eds. Cambridge, UK: Cambridge University Press, 2009.