Topology Control for Time-Varying Contested Environments

Ertugrul N. Ciftcioglu†, Kevin S. Chan‡, Ananthram Swami§, Derya H. Cansever† and Prithwish Basu‡
†IBM Research, Yorktown Heights, NY 10598
‡U.S. Army Research Lab, Adelphi, MD 20783
§Raytheon BBN Technologies, Cambridge, MA 02138

Email: enciftci@us.ibm.com, kevin.s.chan.civ@mail.mil, ananthram.swami.civ@mail.mil,
derya.h.cansever.civ@mail.mil, pbasu@bbn.com

Abstract—This work considers a problem of topology design and control when some of the links may become unavailable. When a network operates in a contested environment, the operator may wish to alter the network topology to enhance robustness. This may come at the expense of operational costs. We consider a dynamic programming framework where the goal is to minimize time average costs, where costs are the sum of network properties (e.g. eccentricity), edit costs and maintenance costs. We consider practical low-complexity control algorithms which focus on instantaneous network states. We particularly introduce a modified myopic policy which attempts to reduce average cost by benefiting future costs. We also provide decision rules for different algorithms and characterize expected costs for random hostility models. Our numerical results demonstrate that our modified myopic scheduler outperform other schedulers in terms of average sum cost for various settings.

I. INTRODUCTION

Tactical operations are increasingly relying on networked infrastructures to provide timely, accurate and reliable communications between network operators and information systems [1]. To ensure mission success, network control must maintain topologies that yield satisfactory performance metrics. One aspect of this problem is to most effectively provide communications between nodes of interest while operating within the constraints of the environment. After detecting some change in the environment, the operator has the task of switching between topologies to maximize the effectiveness and efficiency of the network. This paper provides a dynamic programming [2], [3] approach to topology switching in the presence of an active contestant.

The tactical network environment is highly dynamic, with constantly changing mission requirements and random uncontrollable environmental influences. With regard to the tactical network, these networks are limited in resources, often operating in power-saving modes to conserve energy. As a result, communications within these networks are multihop and only a subset of available links may be activated. Further, if the commander decides to switch operating topologies through densification (adding) or by thinning (removing) specific links, there may be significant costs associated with these processes. Further, links may have varying temporal permanence, organizational restrictions, or require a temporary loosening of current restrictions.

Decisions to switch or repair topologies must be made after taking into account the estimated level of interference in the network. We assume that links can be disabled, specifically those that have been part of the densification process. This paper studies the impact of external influences on the control of the network topology. By considering hostility and resource costs due to densification, we are able to develop a preliminary set of strategies to maximize network performance over a specific mission duration.

Network topology design has received significant attention in the graph theory and network science literature. However, the research till date has focused on augmentation of static graphs for improved fault tolerance (maximization of $k$-connectivity) or information flow properties (minimization of eccentricity [4], diameter [5], and average shortest path length [6]). We investigate various network design problems for graphs whose structure evolves over time.

Topology control has been studied in the sensor network community [7], [8], [9]. Majority of these models focus on graph theoretic or linear/integer programming solutions, where the network objective is traditional metrics as transmission power [8], [10]. Moreover, the networks are usually static [9], [10]. To our knowledge, topology control has been seldom considered along with dynamic programming. Nevertheless, these works focus on more traditional network objectives as throughput [11], reliability [12] and consider optical mesh networks [11]. There has also been research on “network controllability” [13] (and more recently multi-layer networks) in the context of linear continuous-time control of node-states describes how to find a minimal driver-node-set which can move the network to any desired point in state-space. However, actually controlling the network structure to achieve a certain
and response time in a network. Centrality, eccentricity, which in effect can improve robustness, new edges for larger topology indices. This densification, which can result in multiple link additions and/or deactivations. In the most general model, we can define $K$ different actions which switches to one of the $K$ allowed topologies.

We consider a few different hostility models: $h \in \{0, 1, ..., H\}$ is the number of links disabled in the current slot due to hostility.

**Hostility attacking Fresh links (HM-F):** In this model, we assume that the attacker/environment probabilistically disables the freshest links, that is the link(s) between the current topology index $I$ and the previous index $I - 1$. When $L = 1$, and $h = 1$, disability of the freshest link means that the topology index directly decrements by one.

**Random hostility (HM-R):** In this model, a hostile attack can randomly disable any of the links. This means that the resultant topology after the hostile attack may be an “unallowed” topology. We call this topology a perturbed topology. If the perturbed topology is indeed an unallowed topology, it is mandatory to either fix the disabled link(s), or deactivate a number of links to transform it to an allowed topology with lower density and index.

An intermediate/hybrid model between the first two hostility models on which we focus in this paper is the following: The core network consisting of the base topology $G_0$ is well-protected and is not affected by the hostility. On the other hand, the hostile environment randomly disables the subset of links added to $G_0$.

We do not consider the scenario where hostile attacks have spatial and temporal components.

**Costs:** Operating networks in a satisfactory manner for the mission at hand is associated with several different costs. While some of them are more in line with the traditional notion of costs (e.g., energy, computational resources), the first category of costs we address depends on the topology and desired network properties.

**Network Property Costs:** This group of costs are associated with network properties relevant to the problem of interest, such as eccentricity, connectivity and centrality. While these properties distinctly depend on the topology structure, a general tendency is that the network property improves with extra links and deteriorates when links are removed. Moreover, diminishing returns are typical in terms of improvement of the network property with the addition of edges. In this work, we consider eccentricity as a representative property. Eccentricity of a node is the maximum length of the shortest paths from the node to all other nodes. This property can describe the maximum response time to (or from) a given node, which can be the command center/tactical operations center for tactical applications.

Eccentricity of several networks was analyzed in [4], where a greedy algorithm was presented to add links in order to reduce eccentricity. The results in [4] suggest that eccentricity $\epsilon$ can be approximated as a function of the number of newly added links $L_N$ as follows:

$$\epsilon(L_N) \approx \min(C + \frac{\gamma}{L_N}, U),$$

where $U \geq C$, $C$ is the lower bound on the eccentricity, $U$ is
the eccentricity of the base topology, and \( \gamma \) relates the decay in eccentricity with the number of links added. We also note that computing \( \epsilon \) for graph \( G = (E, V) \) has a computational complexity of \( O(|E| + |V| \log |V|) \) from Dijkstra. Assuming the base topology \( G_0 \) is the basic default topology, we can run the greedy algorithm from [4] to determine the ordered sequence of links to minimize the eccentricity. Afterwards, we can define \( G_I \) as the resulting network with \( G_0 \) and \( LI \) additional links. Accordingly, the associated/resulting eccentricity relates the decay and additional links. Accordingly, the associated/resulting eccentricity could be concave increasing, implying that there is a fixed node activation cost and additional link activation costs. For brevity, we assume it could be concave increasing, which could be expressed/approximated as:

\[
\epsilon(I) \approx \min(C + \frac{\gamma}{t_I}, U).
\]  

(2)

While we focus on this particular function, we also demonstrate that the results in this paper also follow under other functions (e.g. exponential, linear) decreasing in number of added links.

**Edit/Switching Costs:** The most basic model is to characterize the cost of any topology modification from \( G_I \) to \( G_J \) with the edit distance \( E_I \ominus E_J \). We consider a couple of enhancements on top of this as follows.

The cost to adding links (activating from scratch) is likely to be more than the cost of repairing/fixing links. While this needs further consideration (recovering disabled links might be actually even more difficult), both of these costs would be greater than deleting/deactivating links. For brevity, we assume that adding a link results in cost \( \alpha_a \) and deactivating a link results in a cost of \( \alpha_d \).

**Maintenance Costs:** This component of cost could reflect energy costs for keeping links active. The specific costs could depend on factors such as distances between the endpoints. We assume that the total cost is linear in the number of links \( |E| \), with \( \alpha_m \) denoting the maintenance cost per link. The total cost could be concave increasing, implying that there is a fixed node activation cost, and additional link activation costs. For instance, it could be linear in the number of links, \( |E| \) (which we assume in the paper with unit cost of \( \alpha_m \) per link), or we could assume it to be concave increasing, which could be implied that there is a fixed node activation cost and additional link activation costs. Without activation costs (\( \alpha_m = 0 \)), there is no incentive for deactivating links, and the only decision would be to add link(s) or not.

**State:** The state \( s(t) \) is the topology index.

**Randomness and State Transitions:** The main source of randomness is the hostility, and the transition probability depends on the hostility distribution. The sequence of events is \( hostility \rightarrow decision \rightarrow hostility \rightarrow decision \) etc.

\[
G(0) \xrightarrow{dyn} G(0)^{design} \xrightarrow{design} G(1) \rightarrow \ldots G(T - 1)^{design} \rightarrow G(T)
\]

Recall that for the HM-R model, the random hostility may cause the topology to not fall in one of the \( K \) permitted states, yet the design step maps them back into an allowed state.

### III. Problem Statement and Objective

The goal is to minimize the time average cost, which includes both costs due to network properties (e.g., eccentricity) \( y \) and operational \( z \) (e.g. edit, activation, maintenance), i.e. \( y + z \),

\[
\min_{a(t)} \frac{1}{T} \sum_{t=1}^{T} z(s(t), a(t)) + y(s(t), a(t))
\]  

(3)

where \( s(t) \) is the state and \( a(t) \) is the action.

The problem can be described by a finite horizon dynamic program [2]. The problem can be approached via the Bellman equation of optimality, that is:

\[
V_t(s(t)) = \max_{a(t)} \{z(s(t), a(t)) + y(s(t), a(t)) + \mathbb{E}\{V_{t+1}(s_{t+1})}\}
\]

which can be equivalently written as

\[
V_t(s(t)) = \max_{a(t)} \{z(s(t), a(t)) + y(s(t), a(t)) + \sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)V_{t+1}(s_{t+1})\}
\]

solved recursively backward through in time. Here \( V(\cdot) \) is the value function. \( s_{t+1} = f(s_t, a_t, h_{t+1}) \), i.e. the state transitions from time \( t \) to \( t + 1 \) depend on the previous state \( s_t \), action \( a(t) \), and the new information (hostility) \( h_{t+1} \).

For instance, under HM-F with hostility following a Bernoulli distribution with hostility parameter (probability) \( h \), and \( L = 1 \), when \( a(t) \) is restricted to adding a single link or not, we can express the transition probabilities as follows:

\[
p(s_{t+1}|s_t, a_t) = \begin{cases}
  h, & \text{if } s_{t+1} = s_t - 1 \\
  1 - h, & \text{if } s_{t+1} = s_t
\end{cases}
\]

(4)

when \( a_t = 0 \), and

\[
p(s_{t+1}|s_t, a_t) = \begin{cases}
  h, & \text{if } s_{t+1} = s_t \\
  1 - h, & \text{if } s_{t+1} = s_t + 1
\end{cases}
\]

(5)

if \( a(t) = 1 \).

Dynamic programming is well known to suffer from the curse of dimensionality, and evaluating the expectations tends to be both difficult and also require full statistics a priori. Next, we present two algorithms which do not suffer from these complications.

### IV. Algorithms

In this section we discuss two low-complexity algorithms which act on the instantaneous topology states.

**A. Myopic**

The myopic policy only focuses on the instantaneous costs without consideration of future cost states.

\[
\min_{a(t)} y(s(t), a(t)) + z(s(t), a(t))
\]  

(6)

Solving the equation, we will demonstrate that, the network can tolerate link disability for dense topologies but reacts for sparser topologies.
B. Modified Myopic

The modified myopic policy we propose tends to be horizon length-aware. Assuming that the activation cost is not very high and the main component is the edit cost, to keep a lower time average cost it would benefit to keep a denser topology at the beginning and “ride the wave” of lower network property cost. To facilitate this, we discount edit costs for earlier stages of a frame, hence motivate network improvements. On the other hand, close to the end of the frame, since the number of future slots left is small, we consider a policy closer to the myopic one. Hence, the overall cost is as follows:

$$
\min_a \gamma(s(t), a(t)) + z(s(t), a(t)) - \beta(T-t)\delta(a(t)-1)
$$

(9)

where \(a(t) = 1\) if the decision is to add/repair links, or \(a(t) = 0\) when the network decides not to add links/backtrack, and \(\delta(u) = 1\) for \(u = 0\), \(\delta(u) = 0\) elsewhere. \(\beta(u)\) is a monotonic non-decreasing function of \(u\) to be specified/instantiated which might also depend on parameters like hostility \(h\) and cost \(\alpha_a\).

Even if the classic myopic policy would possibly not add links for dense topologies, the modified myopic policy might still add seemingly “redundant” links to benefit future slots.

**Remark 1:** While we acknowledge that in the (approximate) dynamic programming literature [3] there exist several other control techniques such as lookahead policies, rollout policies and value/policy iteration based methods, these algorithms typically have significantly more complexity. In this work, we focus on more practical methods which perform optimization using only the current slot, leaving the treatment of the aforementioned algorithms for future work.

V. Analytical Results

In this section, we provide several analytical results and properties for the models under HM-F and HM-R. While some of these results might seem rather straightforward, they do provide insight on the decision structures and how different parameters explicitly effect the decision rules.

A. Decision Rules

1) Hostility compromising freshest links: The myopic policy simply compares the total costs of the various decisions and chooses the action that minimizes the instantaneous cost. For the most basic model, with HM-F, \(L = 1\) and the action is whether to add or not, the decision is to recover (if link was disabled in this slot) or upgrade (if hostility did not effect topology) at topology index \(I + 1\) given that we are at topology index \(I\). Here, \(z(s, a) = \alpha_a + \alpha_m\) if a link is added.

That is, the controller compares:

$$
\frac{\gamma}{I+1} + \gamma I + I + \alpha_m \leq \frac{\gamma}{\alpha_a + \alpha_m} \leq C + \frac{\gamma}{I} + I
$$

(10)

$$
\alpha_a + \alpha_m \leq \frac{C + \gamma}{\alpha_a + \alpha_m} \leq \frac{\gamma}{I(I+1)}
$$

(11)

which results in an index policy given parameters \(\gamma\), \(\alpha_a\) and \(\alpha_m\). We can also rewrite (11) as

$$
I^2 + I - \frac{\gamma}{\alpha_a + \alpha_m} \leq \frac{\gamma a = 1}{\gamma a = 0} \leq 0
$$

(12)

hence \(I = \sqrt{0.25 + \frac{\gamma}{\alpha_a + \alpha_m} - 0.5}\) being the positive root which reveals the value where the decision switches from add edges (\(a = 1\)) to don’t add edges (\(a = 0\)). Note that this requires \(\gamma\) to be notably larger than \(\alpha_a\) in order for the myopic policy to decide to add anything.

For the modified myopic algorithm, we can similarly compare the costs from the two actions:

$$
\alpha_a + C + \frac{\gamma}{I+1} + \alpha_m - \beta(T-t) \leq C + \frac{\gamma}{I} 
$$

$$
I^2 + I - \alpha_a + \alpha_m - \beta(T-t) \leq 0
$$

(13)

(14)

with \(I = \sqrt{0.25 + \frac{\gamma}{\alpha_a + \alpha_m} - 0.5}\) (assuming \(\beta(T-t)\)) being the positive root which reveals the value where the decision switches from add edges (\(a = 1\)) to don’t add edges (\(a = 0\)). We observe as expected that the term \(\beta(T-t)\) has increased the threshold, meaning the controller can decide to add edges even if the myopic/instantaneous cost based controller would not. Notice that in contrast with the pure myopic policy, here the switching index depends upon the time to the horizon \(T-t\).

2) Random Hostility: The main difference between the HM-F and HM-R is that a disabled link may result in an unallowed topology. If this is the case (which we will next show is quite likely), the controller has two options: (i) Fix/repair the disabled link and recover to the original topology before hostility. (ii) Do not fix the link but “backtrack” (reduce the topology by deactivating links) to the densest allowed topology which can be obtained without any link additions and the fact that the link is disabled. In this case, given that the disabled link is \(J\), the myopic policy would decide the following at the original state \(I > J\):

$$
\alpha_a + C + \frac{\gamma}{\alpha_a + \alpha_m} \leq C + \frac{\gamma}{I} + \frac{\gamma}{J} = \alpha_d - \alpha_m \times (I - J)
$$

(15)

which leads to a decision rule given \(I\) and \(J\). If deactivation is free (at no cost), \(\alpha_d\) is simply \(0\).

Next, we investigate the expected cost for the myopic/modified policy when at topology index \(I\). To derive the expected loss, we first observe that at state \(I\), the probability that the disabled link is the link between topologies \(J\) and \(J+1\) (where \(J < I\)) is identical \(\forall J\) as \(\frac{1}{I}\). On the other hand, the loss in network property in the case of backtracking is given as \(\frac{\gamma}{I+1}\). We can readily find the threshold \(J’\) where the backtrack decision is accepted from \(\gamma\) \(\frac{1}{I+1}\) \(\leq \alpha_a\)

as \(J’ = \alpha_a - \frac{\gamma}{\alpha_a}\). Hence, the probability of landing in one of the states where links will be fixed is \(\frac{1}{I+1}\). Accordingly, we can express the expected cost conditioned on a link being disabled as

$$
\frac{\gamma}{I+1} + \frac{1}{I} \sum_{k=\lfloor J’ \rfloor}^{I} \gamma \left( \frac{1}{k} \right) \approx \frac{\alpha_d |J’|}{I} + \gamma |\ln I - \ln |J’| |
$$

(16)

where \(J’ = \frac{\alpha_a a = 1}{\alpha_a a = 0}\) for the myopic policy when \(\alpha_d = 0\).

Note that in (16) we have used a discrete sum formula, i.e.
using the Harmonic number. The Harmonic number \( \sum_{v=1}^{I} \frac{1}{v} \) is approximated by \( \ln I + \eta \) where \( \eta \approx 0.577 \) is the Euler constant. We can also derive analogous expressions for the modified myopic by taking into account discounting \( \beta(T - t) \). Note that both terms in (16) reduce with \( I \). Hence, it actually provides an incentive to increase/maintain high \( I \) even for the long run, in line with the objective of the modified myopic policy.

VI. NUMERICAL RESULTS

In this section, we compare average costs realized for different algorithms under different parameters. We compare the myopic algorithm and our modified myopic algorithm with three algorithms: (i) NA: Never add/repair edges, (ii) AA: Always add/repair edges (iii) RA: Randomly add/repair edges with probability \( h \) (which is assumed to be known). For the modified myopic, we next detail the function \( \beta(u, h, \alpha_a) \). First, we note that \( \beta() \) increases with \( T - t \) by design. Next, \( \beta() \) decreases with \( h \) since otherwise the algorithm might discount the cost corresponding to the “add links” action too much. Finally, we adjust the discount by \( \alpha_a \) since the algorithm should be more conservative when the link addition/activation cost is high. Overall, we consider the following function:

\[
\beta(T - t) = \frac{h \times (T - t)}{\alpha_a}
\]  

(17)

Specifically, we vary the hostility parameter \( h \), the horizon \( T \), and the activation cost \( \alpha_a \). It is apparent that as \( \alpha_a \) decreases, it is better to add more links. We use \( C=14, \gamma = 9 \) for the eccentricity parameters from [4], with \( K = 25 \), that is different topologies with up to 25 links added on the base 75-node base topology \( G_0 \). While the relative performances of the algorithms notably depend on the specific parameters, we observe that our modified myopic algorithm outperforms the rest of the algorithms in general, as it adapts to different parameters and gets the best out of different situations. Comparing Figs. 3-5 for the more comprehensive HM-R hostility model with \( \alpha_a = 2.5, \alpha_d = 0.1 \) (deactivation cost), \( \alpha_m = 0.05 \) (maintenance cost), we also observe the effect of the horizon length. While the myopic algorithm performs well for small horizons, as the horizon length increases it turns out to be insufficient in terms of planning for future costs. On the other hand, our modified myopic algorithm has the incentive of benefiting future costs which shows up in the long horizon runs.

Figure 6 demonstrates how the different cost components contribute to the total cost; we observe that despite the fact that the modified myopic algorithm results in larger edit and maintenance costs than most of the algorithms, it eventually makes up for this by keeping a low network property cost.

We observe that the cost of the myopic policy tends to increase linearly with hostility. Our intuitive explanation to this is as follows: Note that there is typically an index \( I_M \) where the myopic controller switches between add/fix and not add/backtrack, which is independent of the hostility. When the index falls below \( I_M \), the myopic tries to shift it to \( \lceil I_M \rceil \) by
taking an add/fix step, hence the network property cost is in effect stabilized to $C + \frac{1}{I_M}$. However, note that bringing the index to $\lceil I_M \rceil$ also results in an edit cost of $\alpha_e$. The frequency with which the myopic scheduler is forced to take such actions increases (linearly) with the hostility. Hence, the overall average cost follows such a behavior.

We next present results when the network property has different cost has different functional forms. In Fig. 7, the network property cost has the form $z(I) = C + e^{-\delta I}$. We observe similar trends in that the modified myopic policy is successful in reducing the cost. Finally, we consider the more simpler network cost model as $z(I) = C + \max(\gamma(1 - I/K), 0)$, i.e. linearly decreases with number of extra edges added in Fig. 8.

VII. CONCLUSIONS

In this work, we have presented a dynamic programming treatment of optimizing network science based objectives. We have provided algorithms which aim to minimize time average cost. The low complexity modified myopic algorithm we propose outperforms other controllers in various scenarios. We also characterize the decision rules as a function of various parameters, as well as expected values of instantaneous network cost based on the control algorithm and hostility model.

Our work opens up a new framework for cost efficient network design and topology control for tactical networks. Future work includes the case where the hostility is targeted (with also possible spatial and temporal correlation of attacks).

REFERENCES