Survival Value of Communication Networks

Ali Tizghadam
School of Electrical and Computer Engineering
University of Toronto, Toronto, Canada
Email: ali.tizghadam@utoronto.ca

Alberto Leon-Garcia
School of Electrical and Computer Engineering
University of Toronto, Toronto, Canada
Email: alberto.leongarcia@utoronto.ca

***Abstract*—This paper looks at the network robustness problem from a new perspective. Inspired by Darwin's survival value, a graph-theoretical metric, betweenness, in combination with network weight matrix is used to define a global quantity, network criticality, to characterize the adaptability of a network to the changes in network conditions. We show that network criticality can be interpreted as the average cost of a journey between any two nodes of a network, or as the average of link betweenness sensitivity of a network. We investigate communication networks in particular, and show that in order to maximize the carried load of a network, one needs to minimize network criticality. We show that network criticality is a monotone decreasing and strictly convex function of weight matrix. This leads to a well-defined convex optimization problem to find the optimal weight matrix assignment. We investigate the solution of this optimization problem for the weight assignment and compare our results with existing methods.**

I. Introduction

Darwin's theory describes the process of natural selection by which each slight variation, if useful, is preserved. Every process receives a survival value as a result of natural selection that quantifies its overall sensitivity or robustness to the external variations. In this paper we are looking for an appropriate survival value for communication networks. The survival value indicates how adaptable a system is to unexpected events.

Darwin's theory does not consider any "final target" for the evolutionary changes in the nature, but one can see that viewing survival as the goal can lead to an implicit optimization problem. Therefore we arrive at the view that the first goal is to keep the system alive under unforeseen circumstances.

In any network, from small designed networks, to large-scale social networks, and even to the Internet, connectivity is a crucial factor as it is essential for communication. Therefore, the first parameter to consider as a candidate for "survival value" is the connectivity of the graph. Any communication network should evolve in a way that maximizes the probability of future connectivity. This implies that the optimization must address the real-time efficiency and performance of the whole network as a short-term goal, while striving to maintain and improve the survival value of the network as a long-term goal. Another important factor in determining the robustness of a network is the network's response to traffic shifts. This includes the effect of changing in the sources and sinks of traffic. This paper tries to find such survival value and investigate its main attributes.

A wealth of literature is available on network robustness and its different aspects. [1] investigates the relationship between node similarity and optimal connectivity, and arrives at the result that a network provides maximum resistance to node destruction if it is both node-similar and optimally connected. The paper then describes a number of ways to design robust networks satisfying these conditions. But this paper considers only the effect of topology in the robustness of a network.

In [2] a way to design backbone networks is proposed that is insensitive to the traffic matrix (i.e., that works equally well for all valid traffic matrices), and that continues to provide guaranteed performance under a user-defined number of link and router failures. The main idea is that the traffic destined for a sink d is forwarded to intermediate hops with equal splits to all nodes, and then it is forwarded to the destination d. Delay propagation is one of the shortcomings of this method.

In [3] a framework for robust routing in core networks is proposed based on the idea of "link criticality" and "path criticality". In [4] we presented an analysis of betweenness centrality, and provided a framework for robustness analysis. Further development of the idea of criticality is provided in [5], where a mathematical framework for the definition of criticality is proposed within the context of Markov chain theory. In this paper we quantify the robustness using the concept of criticality from [5]. We will show that some critical features and metrics of real networks are directly related to the network criticality. We consider network criticality as the survival value and study its robustness properties.

The rest of this paper is structured as follows. Section II summarizes our previous findings about network criticality. Section III provides some properties of network criticality. Section IV establishes the importance of network criticality as survival value by introducing the relationship between network criticality and some other important graph properties, i.e. average cost, average path length, and connectivity. In section V we propose a convex optimization problem for network criticality. Section VI provides a case study on one of the classical problems in communication networks, capacity assignment, by using the proposed optimization problem to design optimal capacities, and compares the results with two well-known solutions in the literature. The paper is concluded in section VII.

II. NETWORK CRITICALITY

In this section we summarize the results of our previous work on robustness [5].

A. Network Model

We model a network with an undirected weighted graph $G = (N, E, W)$ where N is the set of nodes, E is the set of graph links, and W is the weight matrix of the graph. Throughout this paper we assume that G is a connected graph. We assume that SLA (Service Level Agreement) parameters are already mapped to the weights by some appropriate method. Some of these methods are discussed in [6]. This permits us to abstract different business policies and/or SLA's as parts of the weight definition. In this paper we are interested in the study of the weights and their effect on robustness.

Consider a finite-state irreducible Markov Chain with transition probabilities p_{ij} of transitioning from state i at time t to state j at time $t+1$ (discrete time). The Markov chain can be represented by a state transition diagram with states as nodes in a graph and edges corresponding to allowable transitions, and labels associated with the edges denoting the transition probabilities. The Markov chain can also be viewed as a random walk on the n-node graph with next-step transition probabilities p_{ij} according to the following rule:

$$p_{ij} = \begin{cases} \frac{w_{ij}}{\sum_{k \in A(i)} w_{ik}} & if \quad j \in A(i) \\ 0 & otherwise \end{cases} \quad (1)$$

where $A(i)$ is the set of adjacent nodes of i and $w_{ik} \geq 0$ is the weight of link (i, k).

We are interested in quantifying the betweenness of a node in the random-walk corresponding to a Markov chain. Consider the set of trajectories that begin at node s and terminate when the walk first arrives at node d, that is, destination node d is an absorbing node. We define the betweenness $b_{sk}(d)$ of node k for the $s-d$ trajectories as the average number of times node k is visited in trajectories from s to d.

Let $B_d = [b_{sk}(d)]$ be the $n \times n$ matrix of betweenness metrics of node k for walks that begin at node s and end at node d. Further, let P_d be the matrix of transition probabilities when the random walk is modified so that state d is an absorbing state. We use $P(i|j)$ to show the truncated $(n-1)\times(n-1)$ matrix that results from removing i^{th} row and j^{th} column of matrix P. It is shown in [5] that:

$$B_d(d|d) = (I - P_d(d|d))^{-1} \quad (2)$$

B. Definition of Network Criticality

We now introduce *network criticality*, the metric that we proposed in [5], to quantify the robustness of a network. We start by defining node/link criticality.

Node criticality is defined as the random-walk betweenness of a node over its weight (weight of a node is defined as the sum of the weights of its incident links). Link criticality is also defined as the betweenness of a link over its weight.

Let η_k be the criticality of node k and η_{ij} be the criticality of link $l = (i, j)$. It is shown in [5] that η_i and η_{ij} can be obtained by the following expressions:

$$\begin{aligned} \tau_{sd} &= l^+_{ss} + l^+_{dd} - 2l^+_{sd} \quad or \quad \tau_{sd} = u^t_{sd} L^+ u_{sd} && (3) \\ \tau &= \sum_s \sum_d \tau_{sd}, \quad \hat{\tau} = \frac{1}{n(n-1)} \tau && (4) \\ \eta(k) &= \frac{b_k}{W_k} = \frac{1}{2}\tau = \frac{n(n-1)}{2}\hat{\tau} && (5) \\ \eta_{ij} &= \frac{b_{ij}}{w_{ij}} = \tau = n(n-1)\hat{\tau} && (6) \\ \frac{b_{sk}(d)}{W_k} &= l^+_{sk} - l^+_{sd} - l^+_{dk} + l^+_{dd} && (7) \end{aligned}$$

where $L^+ = [l^+_{ij}]$ is the Moore-Penrose inverse of graph Laplacian matrix L, n is the number of nodes, and $u_{ij} = [0 \; ... \; 1 \; ... \; -1 \; ... \; 0]^t$ (1 and -1 are in i^{th} and j^{th} position).

Observation 2.1: Equations 3 to 6 show that node criticality (η_k) and link criticality (η_{ij}) are independent of the node/link position and only depend on τ (or $\hat{\tau}$) which is a global quantity of the network.

Definition 2.2: We refer to τ as the *network criticality* and $\hat{\tau}$ as *normalized network criticality*.

One can see that $\hat{\tau}$ is a global quantity on network graph G. Equations 5 and 6 show that node (link) betweenness consists of a local parameter (weight) and a global metric (network criticality). $\hat{\tau}$ can capture the effect of topology and community of interest via betweenness, and the effect of traffic via weight (by appropriate definition of weight). The higher the betweenness of a node/link, the higher the risk of using the node/link. Furthermore, one can define node/link capacity as the weight of a node/link, then the higher the weight of a node/link, the lower the risk of using the node/link. Therefore network criticality can quantify the risk of using a node/link in a network which in turn indicates the degree of robustness of the network.

This motivates the rest of our work in this paper. We consider network criticality as the survival value of a network. This survival value is a network-wide metric to capture and optimize network robustness. In this paper our goal is to investigate $\hat{\tau}$ as a function of weight matrix (W). We aim to find an appropriate weight matrix that can optimize the survival value (and robustness as a result) of a network.

III. SOME FACTS ABOUT NETWORK CRITICALITY

In this section we establish some lemmas which will be central to our other derivations.

Lemma 3.1: Network Criticality τ is equal to $2nTr(L^+)$. Equivalently, normalized network criticality $\hat{\tau}$ is $\frac{2}{n-1}Tr(L^+)$.

Proof: Since $\tau_{sd} = l^+_{ss} + l^+_{dd} - 2l^+_{sd}$, we have

$$\begin{aligned} \tau &= \sum_{s,d} \tau_{sd} \\ &= \sum_d \sum_s l^+_{ss} + \sum_s \sum_d l^+_{dd} - 2 \sum_s \sum_d l^+_{sd} \\ &= n \sum_s l^+_{ss} + n \sum_d l^+_{dd} - 2 \times 0 = 2n \sum_i l^+_{ii} \end{aligned}$$

Therefore

$$\begin{aligned}\tau &= 2nTr(L^{+})\\ \hat{\tau} &= \frac{1}{n(n-1)}2nTr(L^{+}) = \frac{2}{n-1}Tr(L^{+})\end{aligned}$$

This completes the proof of lemma 3.1. ▮

■

Lemma 3.2: $\hat{\tau}$ can be written as: $\hat{\tau} = \frac{2}{n-1}\sum_{i=2}^{n}\frac{1}{\lambda_i}$, where $0 = \lambda_1 \le \lambda_2 \le ... \le \lambda_n$ are eigenvalues of graph Laplacian L.

Proof: We know from linear algebra that the trace of a square matrix is equal to the sum of its eigenvalues. On the other hand, the non-zero eigenvalues of L^{+} are reciprocals of the non-zero eigenvalues of L. Lemma 3.2 is then a direct result of Lemma 3.1. ▮

■

Lemma 3.2 will establish a connection between network criticality and the spectrum of graph Laplacian.

Lemma 3.3: For any weight matrix W of links of a graph: $Vec(W)^t\nabla\tau + \tau = 0$, where $Vec(W)$ is a vector obtained by concatenating all the rows of matrix W.

Proof: Suppose we scale all the link weights in a graph with factor t, then by equation 6:

$$\tau(tVec(W)) = \frac{b_{ij}}{tw_{ij}}$$

Note that the transition probabilities $p_{sk}(d)$ are invariant to the scaling of the weights based on their definition in equation 1. Therefore matrix P_d is also invariant to the scaling of the weights. As a direct result matrix B_d (and each one of its elements $b_{sk}(d)$) is invariant to the scaling of the weights based on equation 2. This verifies that any node betweenness $b_k = \sum_{s,d} b_{sk}(d)$ is invariant to the scaling and finally equation 6 asserts that any link betweenness b_{ij} is invariant to the scaling of the link weights. Therefore

$$\tau(tVec(W)) = \frac{1}{t}\tau(Vec(W)) \tag{8}$$

By taking the derivative of τ with respect to t, we have

$$Vec(W)^t\nabla\tau = \frac{-1}{t^2}\tau(W) \tag{9}$$

It is enough to consider equation 9 at $t = 1$ to get $Vec(W)^t\nabla\tau + \tau = 0$. ▮

■

IV. Some Interpretations of Network Criticality

In this section we try to shed more light on the concept of network criticality as the survival value for network by providing some interpretations of it.

A. Network Criticality and Average Path Cost

We assume certain cost to travel along a path and study the effect of network criticality τ on average cost incurred by a message during its walk from source s to destination d.

We consider the following scenario. For each link $l = (i, j)$ there is a cost $z_l = z(i, j)$. Note that this cost is different from the weight of the link. After a random-walk starts from source node s, at each step it traverses one link, incurs a cost, and continues until it is absorbed by destination d. We wish to calculate the average cost of this journey. Theoretically, the number of hops of a path that is taken by random-walk can be infinite.

Using the properties of Markov Chains one can calculate the average cost $\varphi(s, d)$ incurred by paths between source s and destination d:

$$\varphi(s,d) = E\{\sum_{k=0}^{\infty} z(d_k, d_{k+1})\} \quad where \quad s = d_0 \tag{10}$$

We expand equation 10 using elementary probability.

$$\begin{aligned}\varphi(s,d) &= \sum_{d_1,d_2,...} p(d_1,d_2,...|d_0=s)(\sum_{k=0}^{\infty} z(d_k,d_{k+1}))\\ &= \sum_{d_1} p_{sd_1}\{z(s,d_1) + \sum_{k=1}^{\infty} p(d_2,...|d_0)z(d_k,d_{k+1})\}\\ \varphi(s,d) &= \sum_j p_{sj}z(s,j) + \sum_j p_{sj}\varphi(j,d) \qquad (11)\end{aligned}$$

Equation 11 has a recursive form, which we will express in matrix form. We relabel the nodes so that node d is the last node. We also define $f_s = \sum_j p_{sj}z(s,j)$. Equation 11 can be written as:

$$\begin{aligned}\overrightarrow{\phi}_d(d|d) &= \overrightarrow{f}(d|d) + P_d(d|d)\overrightarrow{\phi}_d(d|d)\\ \overrightarrow{\phi}_d(d|d) &= (I - P_d(d|d))^{-1}\overrightarrow{f}(d|d) \qquad (12)\end{aligned}$$

where $\overrightarrow{\phi}_d = [\phi(s_1,d), \phi(s_2,d), ..., \phi(s_n,d)]$, $\overrightarrow{f} = [f_{s_1}, f_{s_2}, ..., f_{s_n}]$. Now we can use equation 2 to write the cost as a function of betweenness. In order to simplify the notation, we substitute $p_{sk}(d)$ with p_{sk} in the following equations. This is safe because node d is an absorbing node and it does not have any effect on the transition probability of other node pairs. In the following equations we work with reduced matrices where the effect of absorbing state is already considered by removing the row and column corresponding to node d.

$$\begin{aligned}\overrightarrow{\phi}_d(d|d) &= B_d(d|d)\overrightarrow{f}(d|d) \quad or\\ \varphi(s,d) &= \sum_{k=1}^{n} b_{sk}(d)f_k = \sum_{k=1}^{n} b_{sk}(d)\sum_j p_{kj}z(k,j) \qquad (13)\end{aligned}$$

Now we are ready to calculate the average cost over all node-pairs.

$$\begin{aligned}\bar{\varphi} &= \frac{1}{n(n-1)}\sum_{s,d}\varphi(s,d)\\ &= \frac{1}{n(n-1)}\sum_k(\sum_{s,d} b_{sk}(d)\sum_j p_{kj}z(k,j))\\ &= \frac{1}{n(n-1)}\sum_k(\sum_j p_{kj}z(k,j)b_k)\end{aligned}$$

One can use relation 5 to find the relationship between average cost and criticality.

$$\begin{aligned}\bar{\varphi} &= \frac{1}{n(n-1)}\sum_k(\sum_j p_{kj}z(k,j)\frac{1}{2}\tau W_k)\\ &= \frac{1}{2n(n-1)}\tau\sum_k(\sum_j \frac{w_{kj}}{W_k}z(k,j)W_k)\\ &= \frac{1}{2}\hat{\tau}\sum_k(\sum_j w_{kj}z(k,j)) \qquad (14)\end{aligned}$$

Observation 4.1: Equation 14 shows that the average network cost is the product of normalized network criticality and total weighted graph cost ($\sum_k(\sum_j w_{kj} z(k,j))$). If $\sum_k(\sum_j w_{kj} z(k,j))$ is fixed at constant value C (maximum budget) then *the average network cost is proportional to the criticality of the network.*

This interpretation of network criticality is important because in many practical situations we aim to minimize the average cost of a network. For example most of the traffic engineering algorithms try to minimize a kind of cost in the system. Another example is network planning (or re-planning). In network design we have an optimization criteria where a cost metric is minimized.

B. Network Criticality and Average Hop Length

Network Criticality is also related to the average hop length of a walk. The following important result relates the average length of random-walk to the network criticality.

Lemma 4.2: Let T be the average length (number of hops) of a random-walk over all source-destination pairs, and $\bar{W}$ be the average weight of all nodes. Then:

$$T = \frac{n}{2}\bar{W}\hat{\tau}$$

Proof: It is enough to consider the special case of unit cost $z(i,j) = 1$ for all the links of the network in equation 14.

Lemma 4.2 reveals that the average hop length of a random-walk is proportional to the product of normalized network criticality and average node weights. If we fix the total weight of a network at a budget C, then the average hop length of a walk would be proportional to the normalized network criticality, therefore, the normalized network criticality can quantify the average path length for the network flows.

C. Network Criticality and Average Betweenness Sensitivity

Another interpretation for network criticality is based on the betweenness of different network links. Since $\tau = \frac{b_{ij}}{w_{ij}}$, we have for $w_{ij} > 0$:

$$\frac{\partial \tau}{\partial w_{ij}} = \frac{1}{w_{ij}}\frac{\partial b_{ij}}{\partial w_{ij}} - \frac{\tau}{w_{ij}} \quad or \quad w_{ij}\frac{\partial \tau}{\partial w_{ij}} = \frac{\partial b_{ij}}{\partial w_{ij}} - \tau \qquad (15)$$

By adding the results of equation 15 for different links of the network one can see:

$$\sum_{(i,j)\in E} w_{ij}\frac{\partial \tau}{\partial w_{ij}} = \sum_{(i,j)\in E}\frac{\partial b_{ij}}{\partial w_{ij}} - m\tau \qquad (16)$$

Combining equation 16 and Lemma 3.3 results in:

$$\tau = \frac{1}{m-1}\sum_{(i,j)\in E}\frac{\partial b_{ij}}{\partial w_{ij}} \qquad (17)$$

where m is the number of links of the network.

Observation 4.3: According to equation 17 network criticality τ can be interpreted as the average of link betweenness derivatives or sensitivities with respect to link weight.

Equation 17 suggests an effective approach to design routing and flow assignment algorithms. If we can estimate the variation of each link betweenness with respect to its weight (i.e. $\frac{\partial b_{ij}}{\partial w_{ij}}$), then we can use this variation as a cost to develop routing strategies to find min-cost paths.

D. Network Criticality and Algebraic Connectivity

Fiedler [7] defined algebraic connectivity as the first non-zero eigenvalue (λ_2) of the Laplacian matrix of a connected graph (recall that the first eigenvalue of Laplacian matrix for a connected graph is zero). Algebraic connectivity is an upper bound for node connectivity and link connectivity. Therefore, *the further λ_2 is from zero, the higher the node and link connectivity of a graph.*

We now establish lower and upper bounds for network criticality based on algebraic connectivity.

Theorem 4.4: Normalized network criticality satisfies the following bounds : $\frac{2}{(n-1)\lambda_2} \leq \hat{\tau} \leq \frac{2}{\lambda_2}$.

Proof: Lemma 3.2 can be used to obtain spectral bounds for network criticality. Since λ_2 is the smallest non-zero eigenvalue of graph Laplacian and all the eigenvalues are non-negative, we have:

$$\hat{\tau} = \frac{2}{n-1}\sum_{i=2}^{n}\frac{1}{\lambda_i} \leq \frac{2}{n-1}\times\frac{n-1}{\lambda_2} \leq \frac{2}{\lambda_2} \qquad (18)$$

This establishes the upper bound for normalized network criticality. To get the lower bound we observe that:

$$\hat{\tau} = \frac{2}{n-1}\sum_{i=2}^{n}\frac{1}{\lambda_i} \geq \frac{2}{n-1}\frac{1}{\lambda_2} \qquad (19)$$

combining inequalities 18 and 19 completes the proof of theorem 4.4.

Theorem 4.4 shows the relationship between network criticality and connectivity. Since normalized network criticality is upper bounded by the reciprocal of algebraic connectivity, improvement of connectivity (increasing λ_2) improves the robustness as well (decreasing the upper bound of $\hat{\tau}$), but it is important to note that increasing connectivity at the same time decreases the lower bound of network criticality, which in turn causes more variance in network criticality. In other words, we can't uniformly improve the robustness of a network just by increasing the connectivity.

E. Network Criticality in Communication Networks

In this section we show the importance of network criticality in the study of communication networks. Let λ be the average input rate of the network, and let the weight of each link be the capacity of the link $(i,j) = l$ (i.e. $w_{ij} = c_{ij} = c(l)$). Further, let x_k be the average load on node k and c_k be the capacity of node k. By applying Little's formula and using lemma 4.2 we have:

$$x_k = \lambda\pi_k T = \lambda\frac{W_k}{\sum_i W_i}\frac{n}{2}\bar{W}\hat{\tau} = \frac{\lambda}{2}W_k\hat{\tau} \qquad (20)$$

But W_k is the total capacity of node k, therefore

$$x_k \leq W_k \Rightarrow \frac{\lambda}{2}W_k\hat{\tau} \leq W_k \Rightarrow \lambda \leq \frac{2}{\hat{\tau}} \qquad (21)$$

We can summarize these results in theorem 4.5.

Theorem 4.5: To maximize the carried load of a network, one needs to minimize the (normalized) network criticality, where the link weight is defined as the link capacity:

$$\max_{W} \lambda = \frac{2n(n-1)}{\min_W \tau} = \frac{2}{\min_W \hat{\tau}}$$

∎

V. Optimization of Network Criticality

Different interpretations of network criticality, in particular theorem 4.5, show that network criticality should be minimized in order to have robust behavior in different applications. The following theorem proves that the minimization of network criticality is in fact doable.

Theorem 5.1: τ is a strictly convex function of graph weights. Further, τ is a non-increasing function of link weights.

Proof: We note that function $f(X) = Tr(X^{-1})$ is strictly convex on X, if X is positive definite (see [8]). Therefore, considering well-known equation $L^+ = (L + \frac{J}{n})^{-1} - \frac{J}{n}$ [8] (J is an $n \times n$ matrix whose entries are all equal to 1),we can see that $\tau = 2nTr(L^+) = 2nTr(L + \frac{J}{n})^{-1} - 2n$ is strictly convex on matrix $L + \frac{J}{n}$ (since L is positive semi-definite, $L + \frac{J}{n}$ is always positive definite).

It is also not difficult to show that $\frac{\partial \tau}{\partial w_{ij}} = -2n\|L^+ u_{ij}\|^2$, which is always non-positive, therefore, τ is a monotone decreasing function of link weights. ∎

∎

Theorem 5.1 has some direct consequences.

Observation 5.2: The problem of finding graph weights to optimize network criticality is a convex optimization problem and all the related literature can be used to solve it.

Observation 5.3: Due to the fact that the τ is a strictly convex function of the weights, an optimization problem with some constraints has a unique solution. As τ is a non-increasing function of weights, our optimization problem would be to minimize network criticality where some constraints are imposed on weight matrix.

The ultimate goal is to find a method to minimize network criticality. Hence, we consider the minimization of τ under some constraints. Motivated by equation 14. we set $\sum_{(i,j)\in E} z_{ij} w_{ij} = C$ as a reasonable constraint for our optimization problem. One can consider C as the maximum budget for total network weight. The main optimization problem is then:

$$\begin{aligned} & Minimize \quad \bar{\tau} \\ & Subject\ to \quad \textstyle\sum_{(i,j)\in E} z_{ij} w_{ij} = C \quad ,C\ is\ fixed \\ & w_{ij} \geq 0 \quad \forall (i,j) \in E \end{aligned} \qquad (22)$$

Optimization problem 22 can be converted to a semi-definite programming problem. Suppose we let $\Gamma = (L + \frac{J}{n})^{-1}$, then Γ can be written as a semi-definite inequality as follows. We consider matrix $\Theta = \begin{pmatrix} \Gamma & I \\ I & L + \frac{J}{n} \end{pmatrix}$. The necessary and sufficient condition for positive semi-definiteness of Θ is that its

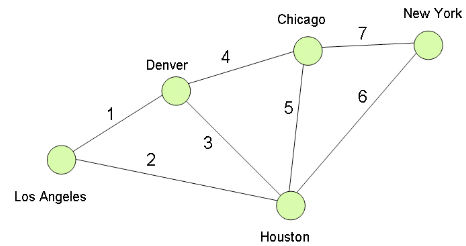

Fig. 1: Kleinrock's Network

Schur complement ([8]) be positive semi-definite. By applying Schur complement for matrix Θ, optimization problem 22 can be converted to the following semi-definite program:

$$\begin{aligned} & Minimize \quad \frac{2}{n-1} Tr(\Gamma) - \frac{2}{n-1} \\ & Subject\ to \quad \sum_{i,j} z_{ij} w_{ij} = C \\ & \begin{pmatrix} \Gamma & I \\ I & L + \frac{J}{n} \end{pmatrix} \geq 0 \end{aligned} \qquad (23)$$

A. Capacity Planning

Let the weight of a link be equal to the link capacity, that is, $w_{ij} = c_{ij}\ \forall (i,j) \in E$ (c_{ij} denotes the capacity of link (i,j)). We investigate the capacity assignment problem in which network topology and traffic load $\gamma_{ij}\ \forall (i,j) \in E$ are assumed known and fixed. This requires that we add constraints $c_{ij} \geq \gamma_{ij}\ \forall (i,j) \in E$ to optimization problem 22. The rest is the same and our semi-definite approach can be used to solve the capacity assignment problem.

VI. Case Study

In order to show the robustness properties of network criticality, we investigate the capacity assignment problem as defined in [9] for two different network topologies.

A. Kleinrock's Network

We compare our capacity planning method with Kleinrock's method for capacity assignment [9] and Meister's extension [10] using the example of telegraph network in Kleinrock illustrated in Fig. 1(see [9], pp. 22-23). In this example the link cost factors z_{ij} are all considered equal to unity. Kleinrock's method finds capacities of the links in such a way to minimize the average delay of the network under the independence assumption and when the link loads are known.

One problem with Kleinrock's approach is that it assigns very long delays to the links with small loads. Meister's method is an alternative approach which assigns equal delays to all the links, of course at the expense of a large deviation from optimal average network delay which can be achieved by Kleinrock's solution. The proposed solution in this paper assigns capacity of the links in a way to balance the individual link delays so as to have acceptable link delays while still we have a good average network delay. Table I shows the capacity assigned to the links using all the methods. The second column of table I shows the individual link loads.

Link	Load	Kleinrock	Meister	Criticality Method
1	3.15	27.93	27.00	29.63
2	3.55	29.85	27.40	33.31
3	0.13	5.16	23.98	12.67
4	3.64	30.28	27.49	32.95
5	0.82	13.46	24.67	13.36
6	3.88	31.38	27.73	33.64
7	9.95	53.99	33.80	36.43

TABLE I: Capacity Assignment using 3 Different Methods

Method	Average Network Delay	Network Criticality
Kleinrock	44.72	1.06
Meister	55.01	0.80
Criticality Method	49.30	0.56

TABLE II: Average Network Delay and Network Criticality using Different Methods

Columns 3, 4, and 5 show the optimal capacity assignment using Kleinrock's method, Meister's method, and our proposed method (which we call it criticality method) respectively. The minimum average network delay for these methods are given in the second column of table II. In the criticality method we actually optimize the robustness (not the average delay as it is the case in Kleinrock and Meister), therefore it is not surprising to see that the average delay obtained by criticality method is between two extremes of Kleinrock (to minimize the average network delay) and Meister (to minimize the maximum link delay). We can see this fact from the values of τ obtained from these capacity assignment methods. The value of network criticality in all of these methods is shown the third column of table II. This table shows that the criticality in the proposed method has the minimum value as expected, and the network criticality in Meister's method is less than Kleinrock's method, therefore, Meister's method is more robust than Kleinrock's. Table III shows individual link delays for all three methods. This table also reveals that the criticality method provides more robust solution.

Link	Kleinrock	Meister	Criticality Method
1	40.36	41.93	37.76
2	38.02	41.93	33.60
3	198.67	41.93	79.71
4	37.54	41.93	34.12
5	79.10	41.93	79.71
6	36.36	41.93	33.60
7	22.71	41.93	37.76

TABLE III: Individual Link Delays using 3 Different Methods

B. Trap Network

Our next example is the trap network shown in Fig. 2. We assume that all link costs are 1 except for the cost of link $(3,4)$ which is equal to 5, that is $z_{34} = z_{43} = 5$. We also assume that the total cost is equal to $C = 1000$. Optimal link weights are shown in Fig. 2. One can see that the link $(3,4)$ is effectively down. This means that the topology of the trap network is changed. In fact, if we set $w_{34} = w_{43} = 0$ in the optimization problem, and if we use the equal cost for all the links, the optimal link weights would be the same as shown in Fig. 2.

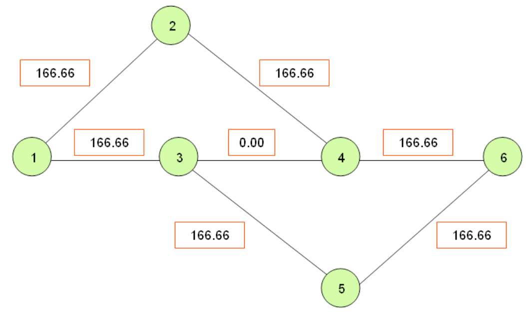

Fig. 2: Trap Network

VII. Conclusion

In this paper we investigated the properties of a global graph metric, network criticality, and considered it as the survival value for a network, because it can quantify some of important network quantities, such as average cost, path length, and connectivity. We showed that network criticality is a strictly convex function of link weights and investigated the convex optimization problem of minimizing the network criticality under some constraints on the weight matrix. We also found a semi-definite programming representation of this problem which permits us to use available literature on semi-definite programming to solve the optimization problem and find the optimal weights. Capacity assignment problem can be considered as a special case of this general problem where the weight of a link is equal to its capacity.

References

[1] A. H. Dekker and B. D. Colbert. Network Robustness and Graph Topology. *Australasian Computer Science Conference*, 26:359–368, Jan. 2004.

[2] R. Zhang-Shen and N. McKeown. Designing a Predictable Internet Backbone with Valiant Load-Balancing. In *Thirteenth International Workshop on Quality of Service (IWQoS)*, Passau, Germany, June 2005.

[3] A. Tizghadam and A. Leon-Garcia. A Robust Routing Plan to Optimize Throughput in Core Networks. *ITC20, Elsvier*, pages 117–128, 2007.

[4] A. Tizghadam and A. Leon-Garcia. On Congestion in Mission Critical Networks. In *Workshop on Mission Critical Networks (MCN), INFOCOM*. IEEE, April 2008.

[5] A. Tizghadam and A. Leon-Garcia. On Robust Traffic Engineering in Core Networks. In *IEEE GLOBECOM*, December 2008.

[6] P. Van Mieghem and F. A. Kuipers. Concepts of Exact QoS Routing Algorithms. *IEEE/ACM TRANSACTIONS ON NETWORKING*, 12(5):851–864, October 2004.

[7] M. Fieler. Algebraic Connectivity of Graphs. *Czechoslovak Math. Journal*, 23(98):298–305, 1973.

[8] Dennis S. Bernstein. *Matrix Mathematics*. Prinston University Press, 2005.

[9] L. Kleinrock. *Communication Nets, Stochastic Message Flow and Delay*. McGraw-Hill, New York, 1964.

[10] B. Meister, H. R. Muller, and H. R. Rudin. New optimization criteria for message switching networks. *IEEE Transactions on Communication Technology*, 19(3):256–260, June 1971.