2015 IEEE International Conference on Data Mining

On the Connectivity of Multi-layered Networks: Models, Measures and Optimal Control

Chen Chen, Jingrui He, Nadya Bliss and Hanghang Tong
Arizona State University
Tempe, Arizona 85287, USA
Email: {chen_chen, jingrui.he, nadya.bliss, hanghang.tong}@asu.edu

***Abstract*—Networks appear naturally in many high-impact real-world applications. In an increasingly connected and coupled world, the networks arising from many application domains are often collected from different channels, forming the so-called *multi-layered networks*, such as cyber-physical systems, organization-level collaboration platforms, critical infrastructure networks and many more. Compared with single-layered networks, multi-layered networks are more vulnerable as even a small disturbance on one supporting layer/network might cause a ripple effect to all the dependent layers, leading to a catastrophic/cascading failure of the entire system. The state-of-the-art has been largely focusing on modeling and manipulating the cascading effect of *two-layered* interdependent network systems for some *specific* type of network connectivity measure.**

This paper generalizes the challenge to multiple dimensions. First, we propose a new data model for multi-layered networks (MULAN), which admits an arbitrary number of layers with a much more flexible dependency structure among different layers, beyond the current pair-wise dependency. Second, we unify a wide range of classic network connectivity measures (SUBLINE). Third, we show that for any connectivity measure in the SUBLINE family, it enjoys the *diminishing returns property* which in turn lends itself to a family of provable near-optimal control algorithms with linear complexity. Finally, we conduct extensive empirical evaluations on real network data, to validate the effectiveness of the proposed algorithms.

Keywords*—*multi-layered network; connectivity control;

I. INTRODUCTION

Networks are ubiquitous and naturally appear in many high-impact applications. Moreover, in a way reminiscent of the famous quote from *Leonardo da Vinci*[1], the networks arising from these application domains are often interconnected/interwined with each other, forming the so-called *multi-layered networks* [3], [8], [16], [18]. Cyber-physical systems are a classic example of multi-layered networks, where the *control layer* controls the *physical layer* (e.g., power grid) through the *communication layer* (e.g., computer networks); and in the meanwhile, the fully functioning of the *communication layers* depends on the sufficient power supply from the *physical layer*. Here, these three interdependent layers naturally form a line-structured dependency graph. Another example is the organization-level collaboration platforms (Fig. 1), where the *team network* is supported by the *social network*, connecting its employee pool, which further interacts with the *information network*, linking to its knowledge base. Furthermore, the *social network* layer could have an embedded multi-layered structure (e.g., each of its layers represents a different collaboration type among different individuals); and so does the *information network*. In this application, the different layers form a tree-structured dependency graph.

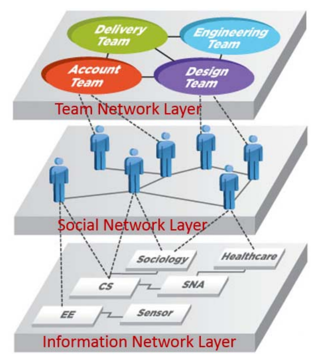


Fig. 1. A simplified example of multi-layered network.

Compared with single-layered networks, multi-layered networks are even more vulnerable to external attacks analogous to the *Butterfly Effect* in the atmosphere system. That is, even a small disturbance on one supporting layer/network might cause a ripple effect to all the dependent layers, leading to a catastrophic/cascading failure of the entire system. In 2012, Hurricane Sandy disabled several major power generator facilities in the New York area, which not only put tens of thousands of people in dark for a long time, but also paralyzed the telecom network and caused a great interruption on the transportation network. Therefore, it is of key importance to identify crucial nodes in the supporting layer/network, whose loss would lead to a catastrophic failure of the entire system, so that counter measures can be taken proactively.

In response to such an imminent need, a recent trend in multi-layered networks research community has been focusing on modeling and manipulating the cascading effect of *two-layered* interdependent network systems [3], [15], [18], [19], [8]. Although much progress has been made, several key challenges have largely remained open. First (**modeling**), most, if not all, of these existing work is devoted to *two-layered* networks with a pair-wise dependency structure; and thus it is not clear how to represent and model multiple (more than two) layers with a more generic dependency structure. Second (**connectivity measures**), there does not exist one single network connectivity measure that is superior to all other measures; but rather several connectivity measures are prevalent in the literature (e.g., robustness, vulnerability, triangle counts). Each of the existing controlling algorithms on multi-layered networks is tailored for one specific connectivity measure. It is not clear if an algorithm designed for one specific connectivity measure is still applicable to other measures. So how can we design a generic control strategy that applies to a variety of prevalent network connectivity measures? Third (**optimal control**), an optimal control strategy tailored for two-layered networks might be sub-optimal, or even misleading to

[1] "Learn how to see. Realize that everything is connected to everything else."

1550-4786/15 $31.00 © 2015 IEEE
DOI 10.1109/ICDM.2015.104

multi-layered networks, e.g., in case we want to simultaneously optimize the connectivity of multiple layers by manipulating one common supporting layer. On the theoretic side, the optimality of the connectivity control problem of generic multi-layered networks is largely unknown.

This paper aims to address *all* these challenges, and the main contributions can be summarized as

- *New Data Models*. We propose a novel multi-layered network model (MULAN), which admits an arbitrary number of layers with a much more flexible node-level dependency structure among different layers, beyond the current pair-wise dependency (Section II).
- *Connectivity Measures*. We unify a family of prevalent network connectivity measures (SUBLINE), in close relation to a variety of important network parameters (e.g., epidemic threshold, network robustness, triangle counting) (Section III).
- *Optimal Control*. We show that for *any* network connectivity measure in the SUBLINE family, the optimal connectivity control problem with the proposed MULAN model enjoys the diminishing returns property, which naturally lends itself to a family of provable near-optimal control algorithms with linear complexity (Section IV).
- *Empirical Evaluations*. We perform extensive experiments based on real data sets to validate the effectiveness of the proposed algorithms. (Section V).

II. A NEW MULTI-LAYERED NETWORK MODEL

In this section, we propose our new multi-layered network model that admits an arbitrary number of layers with a more generic dependency structure among different layers. We start with the main symbols used throughout the paper (Table I). We use bold upper case letters for matrices (e.g., $\mathbf{A}, \mathbf{B}$), bold lower case letters for column vectors (e.g., $\mathbf{a}$, $\mathbf{b}$) and calligraphic font for sets (e.g., $\mathcal{A}$, $\mathcal{B}$). The transpose of a matrix is denoted with a prime, i.e., $\mathbf{A}'$ is the transpose of matrix $\mathbf{A}$.

TABLE I. MAIN SYMBOLS.

Symbol	Definition and Description
$\mathbf{A}, \mathbf{B}$	the adjacency matrices (bold upper case)
$\mathbf{a}, \mathbf{b}$	column vectors (bold lower case)
$\mathcal{A}, \mathcal{B}$	sets (calligraphic)
$\mathbf{A}(i, j)$	the element at i^{th} row j^{th} column in matrix $\mathbf{A}$
$\mathbf{A}(i, :)$	the i^{th} row of matrix $\mathbf{A}$
$\mathbf{A}(:, j)$	the j^{th} column of matrix $\mathbf{A}$
$\mathbf{A}'$	transpose of matrix $\mathbf{A}$
$\mathbf{G}$	the layer-layer dependency matrix
$\mathcal{A}$	networks at each layer of MULAN $\mathcal{A} = \{\mathbf{A}_1, \dots, \mathbf{A}_g\}$
$\mathcal{D}$	inter-layer node-node dependency matrices
θ, ψ	one to one mapping functions
Γ	multi-layered network MULAN $\Gamma =< \mathbf{G}, \mathcal{A}, \mathcal{D}, \theta, \psi >$
$\mathcal{S}_i, \mathcal{T}_i, \dots$	node sets in layer $\mathbf{A}_i$(calligraphic)
$\mathcal{S}_{i\to j}$	nodes in $\mathbf{A}_j$ that depend on nodes $\mathcal{S}$ in $\mathbf{A}_i$
$\mathcal{N}(\mathcal{S}_i)$	nodes and inter-layer links that depend on $\mathcal{S}_i$
m_i, n_i	number of edges and nodes in layer $\mathbf{A}_i$
$\lambda_{<\mathbf{A},j>}, \mathbf{u}_{<\mathbf{A},j>}$	j^{th} largest eigenvalue (in module) and its corresponding eigenvector of network $\mathbf{A}$
$\lambda_{\mathbf{A}}, \mathbf{u}_{\mathbf{A}}$	first eigenvalue and eigenvector of network $\mathbf{A}$
$C(\mathbf{A})$	connectivity function of network $\mathbf{A}$
$I_{\mathbf{A}}(\mathcal{S}_i)$	impact of node set $\mathcal{S}_i$ on network $\mathbf{A}$
$\mathbb{I}(\mathcal{S}_i)$	overall impact of node set $\mathcal{S}_i$ to MULAN

With the above notation, we introduce a new data model for multi-layered networks as follows.

Definition 1. ***A Multi-layered Network Model (*MULAN*).*** *Given (1) a binary $g \times g$ abstract layer-layer dependency network $\mathbf{G}$, where $\mathbf{G}(i, j) = 1$ indicates layer j depends on layer i (or layer i supports layer j), $\mathbf{G}(i, j) = 0$ means no direct dependency from layer i to layer j; (2) a set of within-layer adjacency matrices $\mathcal{A} = \{\mathbf{A}_1, \dots, \mathbf{A}_g\}$; (3) a set of inter-layer node-node dependency matrices $\mathcal{D}$, indexed by pair (i, j), $i, j \in [1, \dots, g]$, such that for a pair (i, j), if $\mathbf{G}(i, j) = 1$, then $\mathbf{D}_{(i,j)}$ is an $n_i \times n_j$ matrix; otherwise $\mathbf{D}_{(i,j)} = \Phi$ (i.e., an empty set); (4) θ is a one-to-one mapping function that maps each node in layer-layer dependency network $\mathbf{G}$ to the corresponding within-layer adjacency matrix $\mathbf{A}_i$ $(i = 1, ..., g)$; (5) φ is another one-to-one mapping function that maps each edge in $\mathbf{G}$ to the corresponding inter-layer node-node dependency matrix $\mathbf{D}_{(i,j)}$. We define a multi-layered network as a quintuple $\Gamma =< \mathbf{G}, \mathcal{A}, \mathcal{D}, \theta, \varphi >$.*

For simplicity, we restrict the within-layer adjacency matrices $\mathbf{A}_i$ to be simple (i.e., no self-loops), symmetric and binary; and the extension to the weighted, asymmetric case is straight-forward. In this paper, we require inter-layer dependency network $\mathbf{G}$ to be an un-weighted directed acyclic graph (DAG). Notice that compared with the existing pair-wise two-layered model, it allows a much more flexible and complicated dependency structure among different layers. For the inter-layer node-node dependency matrix $\mathbf{D}_{(i,j)}$, $\mathbf{D}_{(i,j)}(s, t) = 1$ indicates that node s in layer i supports node t in layer j.

Fig. 2(a) presents an example of a four-layered network. In this example, Layer 1 (e.g., the control layer) is the supporting layer (i.e., the root node in the layer-layer dependency network $\mathbf{G}$). Layer 2 and Layer 3 directly depend on Layer 1 (e.g., one represents a communication layer by satellites and the other represents another communication layer in landlines, respectively), while Layer 4 (e.g., the physical layer) depends on both communication layers (Layer 2 and Layer 3). The abstracted layer-layer dependency network ($\mathbf{G}$) is shown in Fig. 2(b). $\mathcal{A} = \{\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3, \mathbf{A}_4\}$ denotes the within-layer adjacency matrices, each of which describes the network topology in the corresponding layer. In this example, $\mathcal{D}$ is a set of matrices containing only four non-empty matrices: $\mathbf{D}_{(1,2)}$, $\mathbf{D}_{(1,3)}$, $\mathbf{D}_{(2,4)}$, and $\mathbf{D}_{(3,4)}$. For example, $\mathbf{D}_{(3,4)}$ describes the node-node dependency between Layer 3 and Layer 4. The one-to-one mapping function θ maps node 1 (i.e., Layer 1) in $\mathbf{G}$ to the within-layer adjacency matrix of Layer 1 ($\mathbf{A}_1$); and the one-to-one mapping function φ maps edge $< 3, 4 >$ in $\mathbf{G}$ to the inter-layer node-node dependency matrix $\mathbf{D}_{(3,4)}$ as shown in Fig. 2(b).

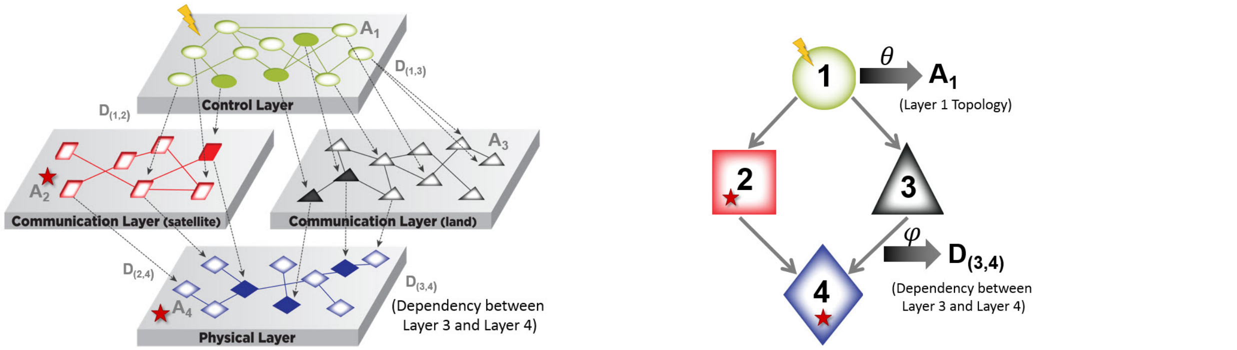
III. UNIFICATION OF CONNECTIVITY MEASURES

In this section, we present a unified view for a variety of prevalent network connectivity measures.

The key of our unified connectivity measure (referred to as SUBLINE in this paper) is to view the connectivity of the entire network as an aggregation over the connectivity measures of its sub-networks (e.g., subgraphs), that is,

$$C(\mathbf{A}) = \sum_{\pi \subseteq \mathbf{A}} f(\pi) \tag{1}$$

where π is a subgraph of $\mathbf{A}$. The non-negative function $f : \pi \to \mathbb{R}^+$ maps any subgraph in $\mathbf{A}$ to a non-negative real number and $f(\Phi) = 0$ for empty set Φ. In other words, we view the connectivity of the entire network ($C(\mathbf{A})$) as the sum of the connectivity of all the subgraphs ($f(\pi)$). Based on such a connectivity definition, we further define the impact function



(a) A four-layered network (b) The corrsponding layer-layer dependency network **G**

Fig. 2. An illustrative example of MuLaN model

of a given set of nodes $\mathcal{S}$ as follows, where $\mathbf{A}\backslash\mathcal{S}$ is the residual network after removing the set of nodes $\mathcal{S}$ from the original network $\mathbf{A}$.

$$I(\mathcal{S}) = C(\mathbf{A}) - C(\mathbf{A} \setminus \mathcal{S}) \tag{2}$$

Based on eq. (2), we can define the overall impact of node set $\mathcal{S}_i$ in $\mathbf{A}_i$ on the multi-layered network system as

$$\mathbb{I}(\mathcal{S}_i) = \sum_{j=1}^{g} \alpha_j I(\mathcal{S}_{i\to j}) = \sum_{j=1}^{g} \alpha_j (C(\mathbf{A}_j) - C(\mathbf{A}_j \setminus \mathcal{S}_{i\to j})) \tag{3}$$

where $\alpha = [\alpha_1, ..., \alpha_g]'$ is a $g \times 1$ non-negative weight vector that assigns different weights to different layers in the system. $\mathcal{S}_{i\to j}$ denotes the set of nodes in layer-j that depend on nodes $\mathcal{S}$ in layer-i.

It turns out many prevalent network connectivity measures can be interpreted from this perspective. Examples include path capacity, loop capacity and triangle capacity. We omit the detailed discussions due to space limit.

IV. Optimal Connectivity Control

In this section, we first define the optimal connectivity control problem (Opera) on the proposed multi-layered network model (MuLaN); then unveil its major theoretic properties; and finally propose a generic algorithmic framework to solve it.

A. Opera: *Problem Statement*

We formally define the optimal connectivity control problem (Opera) on the proposed MuLaN model for multi-layered networks as follows.

Problem 1. Opera on MuLaN

Given: *(1) a multi-layered network* $\Gamma =< \mathbf{G}, \mathcal{A}, \mathcal{D}, \theta, \psi >$ *(2) a control layer* $\mathbf{A}_l$*, (3) an impact function* $\mathbb{I}(.)$*, and (4) an integer* k *(budget);*

Output: *a set of* k *nodes* $\mathcal{S}_l$ *from the control layer (*$\mathbf{A}_l$*) such that* $\mathbb{I}(\mathcal{S}_l)$ *(the overall impact of* $\mathcal{S}_l$*) is maximized.*

In the above definition, the control layer $\mathbf{A}_l$ indicates the sources of the 'attack'; and the $g \times 1$ vector α indicates the target layer(s) as well as their relative weights. For instance, in Figure 2(a), we can choose Layer 1 as the control layer (indicated by the strike sign); and set $\alpha = [0\ 1\ 0\ 1\ ]'$, which means that both Layer 2 and Layer 4 are the target layers (indicated by the star signs) with equal weights between them. In this example, once a subset of nodes $\mathcal{S}$ in Layer 1 are attacked/deleted (e.g., shaded circle nodes), all the nodes from Layer 2 and Layer 3 that are dependent on $\mathcal{S}$ (e.g., shaded parallelogram and triangle nodes) will be disabled/deleted, which will in turn cause the disfunction of the nodes in Layer 4 (e.g., shaded diamond nodes) that depend on these affected nodes in Layer 2 or Layer 3. Our goal is to choose k nodes from Layer 1 that have the maximal impact on both Layer 2 and Layer 4, i.e., to simultaneously decrease the connectivity $C(\mathbf{A}_2)$ and $C(\mathbf{A}_4)$ as much as possible.

B. Opera: *Theory*

In this subsection, we present the major theoretic results of the optimal connectivity control problem (Opera) on multi-layered networks defined in Problem 1. It says that for **any** connectivity function $C(\mathbf{A})$ in the SubLine family (eq. (1)), for **any** multi-layered network in the MuLaN family (Definition 1), the optimal connectivity control problem (Opera, Problem 1) bears **diminishing returns property**.

Theorem 1. Diminishing Returns Property of MuLaN. *For* **any** *connectivity function* $C(\mathbf{A})$ *in the* SubLine *family (eq.* (1)*), for* **any** *multi-layered network in the* MuLaN *family (Definition 1); the overall impact of node set* $\mathcal{S}_l$ *in the control layer* l*,* $\mathbb{I}(\mathcal{S}_l) = \sum_{i=1}^{g} \alpha_i I(\mathcal{S}_{l\to i})$*, is (a) monotonically non-decreasing; (b) sub-modular; and (c)normalized.*

Proof: Omitted for space. ■

C. Opera: *Algorithms*

In this subsection, we introduce our algorithm to solve Opera (Problem 1).

A Generic Solution Framework. Finding out the global optimal solution for Problem 1 by a brute-force method would be computationally intractable, due to the exponential enumeration. Nonetheless, the diminishing returns property of the impact function $\mathbb{I}(.)$ (Theorem 1) immediately lends itself to a greedy algorithm for solving Opera with any arbitrary connectivity function in the SubLine family and an arbitrary member in the MuLaN family, summarized in Algorithm 1.

In Algorithm 1, Steps 2-4 calculate the impact score $\mathbb{I}(v_0)$ ($v_0 = 1, 2, ...$) for each node in the control layer $\mathbf{A}_l$. Step 5 selects the node with the maximum impact score. In each iteration in Steps 7-19, we select one of the remaining $(k-1)$ nodes, which would make the maximum marginal increase in terms of the current impact score (Step 12, $\text{margin}(v_0) = \mathbb{I}(\mathcal{S} \cup \{v_0\}) - \mathbb{I}(\mathcal{S})$). In order to further speed-up the computation, the algorithm admits an *optional* lazy evaluation strategy (adopted from [11]) by activating an optional 'if' condition in Step 11.

We can show that Algorithm 1 leads to a near-optimal solution with linear complexity, thanks to the diminishing returns property in Theorem 1. We omit the detailed algorithm analysis due to space limit.

Algorithm 1 OPERA: A Generic Solution Framework

Input: (1) A multi-layered network Γ, (2) a control layer $\mathbf{A}_l$, (3) an overall impact function $\mathbb{I}(\mathcal{S}_l)$ and (4) an integer k
Output: a set of k nodes $\mathcal{S}$ from the control layer $\mathbf{A}_l$.

1: initialize $\mathcal{S}$ to be empty
2: **for** each node v_0 in layer $\mathbf{A}_l$ **do**
3: calculate margin$(v_0) \leftarrow \mathbb{I}(v_0)$
4: **end for**
5: find $v = \text{argmax}_{v_0}\text{margin}(v_0)$ and add v to $\mathcal{S}$
6: set margin$(v) \leftarrow -1$
7: **for** $i = 2$ to k **do**
8: set maxMargin $\leftarrow -1$
9: **for** each node v_0 in layer $\mathbf{A}_l$ **do**
10: `/*an optional 'if' for lazy eval.*/`
11: **if** margin$(v_0) >$ maxMargin **then**
12: calculate margin$(v_0) \leftarrow \mathbb{I}(\mathcal{S} \cup \{v_0\}) - \mathbb{I}(\mathcal{S})$
13: **if** margin$(v_0) >$ maxMargin **then**
14: set maxMargin $\leftarrow$ margin(v_0) and $v \leftarrow v_0$
15: **end if**
16: **end if**
17: **end for**
18: add v to $\mathcal{S}$ and set margin$(v) \leftarrow -1$
19: **end for**
20: return $\mathcal{S}$

V. EXPERIMENTAL RESULTS

In this section, we empirically evaluate the proposed OPERA algorithms. All experiments are designed to show the effectiveness of the proposed OPERA algorithms at optimizing the connectivity measures (defined in the proposed SUBLINE family) of a multi-layered network (from the proposed MULAN family).

A. Experimental Setup

Data Sets Summary. We perform the evaluations on three application domains, including (D1) a multi-layered Internet topology at the autonomous system level (MULTIAS); and (D2) critical infrastructure networks (INFRANET). For each application domain, we use real networks to construct the within-layer networks (i.e., $\mathcal{A}$ in the MULAN model) and construct one or more inter-layer dependency based on real application scenarios (i.e., $\mathbf{G}$ and $\mathcal{D}$ in the MULAN model). A summary of these data sets is shown in Table II. We will present the detailed description of each application domain in Subsection V-B.

TABLE II. DATA SETS SUMMARY.

Data Sets	Application Domains	# of Layers	# of Nodes	# of Links
D1	MULTIAS	2∼4	5,929∼24,539	11,183∼50,778
D2	INFRANET	3	19,235	46,926

Baseline Methods. To our best knowledge, there is no existing method which can be directly applied to the connectivity optimization problem (Problem 1) of the proposed MULAN model. We generate the baseline methods using two complementary strategies, including *forward propagation* ('FP' for short) and *backward propagation* ('BP' for short). The key idea behind the *forward propagation* strategy is that an important node in *control layer* might have more impact on its dependent networks as well. On the other hand, for the *backward propagation strategy*, we first identify important nodes in *target layer(s)*, and then trace back to its supporting layer through the inter-layer dependency links (i.e., $\mathcal{D}$). For both strategies, we need a node importance measure. In our evaluations, we compare three such measures, including (1) node degree; (2) pagerank measure [13]; and (3) *Netshield* values [21]. In addition, for comparison purposes, we also randomly select nodes either from the control layer (for the forward propagation strategy) or from the target layer(s) (for the backward propagation strategy). Altogether, we have eight baseline methods (four for each strategy, respectively), including (1) 'Degree-FP', (2) 'PageRank-FP', (3) 'Netshield-FP', (4) 'Rand-FP', (5) 'Degree-BP', (6) 'PageRank-BP', (7) 'Netshield-BP', (8) 'Rand-BP'.

OPERA *Algorithms and Variants*. We evaluate three prevalent network connectivity measures, including (1) the leading eigenvalue of the (within-layer) adjacency matrix, which relates to the epidemic threshold of a variety of cascading models; (2) the loop capacity (LC), which relates to the robustness of the network; and (3) the triangle capacity (TC), which relates to the local connectivity of the network. As mentioned in Section III, both the loop capacity and the triangle capacity are members of the SUBLINE family. Strictly speaking, the leading eigenvalue does *not* belong to the SUBLINE family. Instead, it approximates the path capacity (PC), and the latter (PC) is a member of the SUBLINE family. Correspondingly, we have three instances of the proposed OPERA algorithm (each corresponding to one specific connectivity measures) including 'OPERA-PC', 'OPERA-LC', and 'OPERA-TC'. Recall that there is an optional lazy evaluation step (Step 11) in the proposed OPERA algorithm, thanks to the diminishing returns property of the SUBLINE connectivity measures. When the leading eigenvalue is chosen as the connectivity function, such a diminishing returns property does not hold any more. To address this issue, we introduce a variant of OPERA-PC as follows. At each iteration, after the algorithm chooses a new node v (Step 18, Algorithm 1), we (1) update the network by removing all the nodes that depend on node v, and (2) update the corresponding leading eigenvalues and eigenvectors. We refer to this variant as 'OPERA-PC-Up'. For each of the three connectivity measures, we run all four OPERA algorithms.

Machines and Repeatability. All the experiments are performed on a machine with 2 processors Intel Xeon 3.5GHz with 256GB of RAM. The algorithms are programmed with MATLAB using single thread. All data sets used in this paper are publicly available. Due to the space limit, we omit the actual figures for some experimental results. We will include these additional results in an extended technical report.

B. Effectiveness Results

D1 - MULTIAS. This data set contains the Internet topology at the autonomous system level. The data set is available at http://snap.stanford.edu/data/. It has 9 different network snapshots, with $633 \sim 13,947$ nodes and $1,086 \sim 30,584$ edges. In our evaluations, we treat these snapshots as the within-layer adjacency matrices $\mathcal{A}$. For a given supporting layer, we generate the inter-layer node-node dependency matrices $\mathcal{D}$ by randomly choosing 3 nodes from its dependent layer as the direct dependents for each supporting node. For this application domain, we have experimented with different layer-layer dependency structures ($\mathbf{G}$), including a two-layered network, a three-layered line-structured network, a three-layered

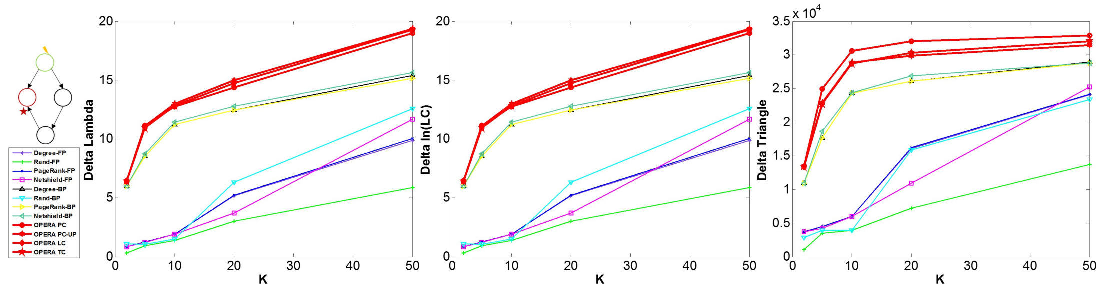


Fig. 3. Evaluations on the MULTIAS data set, with a four-layered diamond-shaped dependency network. The connectivity change vs. budget. Larger is better. All the four instances of the proposed OPERA algorithm (in red) outperform the baseline methods.

tree-structured network and a four-layered diamond shaped network. Figure 3 shows the results on the diamond shaped network. All the four instances of the proposed OPERA algorithm perform better than the baseline methods. Among the baseline methods, the *backward propagation* methods are better than the forward propagation methods. This is because the length of the back tracking path on the dependency network **G** (from the target layer to the control layer) is short. Therefore compared with other baseline methods, the node set returned from the BP strategy is able to affect more important nodes in the target layer. The results on the other dependent networks are similar and omitted due to the space limit. In all these scenarios, the proposed OPERA algorithms perform best consistently.

D2 - INFRANET. This data set contains three types of critical infrastructure networks, including (1) the power grid, (2) the communication network; and (3) the airport networks. The power grid is an undirected, un-weighted network representing the topology of the Western States Power Grid of the United State [23]. It has 4,941 nodes and 6,594 edges. We use one snapshot from the MULTIAS data set as the communication network with 11,461 nodes and 32,730 edges. The airport network represents the internal US air traffic lines between 2,649 airports and has 13,106 links (available at http://www.levmuchnik.net/Content/Networks/NetworkData.html). We construct a triangle-shaped layer-layer dependency network **G** (see the icon of Figure 4) based on the following observation. The operation of an airport depends on both the electricity provided by the power grid and the Internet support provided by the communication network. In the meanwhile, the full functioning of the communication network depends on the support of power grid. We use the similar strategy as MULTIAS to generate the inter-layer node-node dependency matrices $\mathcal{D}$. The results are summarized in Figure 4. Again, the proposed OPERA algorithms outperform all the baseline methods. Similar to the MULTIAS network, the back tracking path from the airport layer to the power grid layer is also very short. Therefore the backward propagation strategies perform relatively better than other baseline methods. In addition, we also change the density of the inter-layer node-node dependency matrices and evaluate its impact on the optimization results (detailed results are omitted for space). We found that (1) across different dependency densities, the proposed OPERA algorithms still outperform the baseline methods; and (2) when the dependency density increases, the algorithms lead to a larger decrease of the corresponding connectivity measures with the same budget.

VI. RELATED WORK

In this section, we review the related work, which can be categorized into two groups: (a) network connectivity control, and (b) multi-layered network analysis.

Network Connectivity Control. Connectivity is a fundamental property of networks, and has been a core research theme in graph theory and mining for decades. Depending on the specific applications, many network connectivity measures have been proposed in the past. Examples include the size of giant connected component (GCC), graph diameter, the mixing time [9], the vulnerability measure [1], the epidemic thresholds [4], the natural connectivity [10] and number of triangles in the network, each of which often has its own, different mathematical definitions.

From algorithm's perspective, network connectivity control aims to optimize (e.g., maximize or minimize) the corresponding connectivity measure by manipulating the underlying topology (e.g., add/remove nodes/links). Recent work tries to solve this problem by collectively finding a subset of nodes/links with the highest impact on the network connectivity measure. For example, Tong et al. [21], [20] proposed both node-level and edge-level manipulation strategies to optimize the leading eigenvalue of the network, which is the key network connectivity measure behind a variety of cascading models. In [5], Chan et al. further generalized these strategies to manipulate the network robustness measure through the truncated loop capacity [10]. Another important aspect of network connectivity control lies in the network dynamics. Chen et al. in [6] proposed an efficient online algorithm to track some important network connectivity measures (e.g., the leading eigenvalue, the robustness measure) on a temporal dynamic network.

Multi-Layered Network Analysis. Multi-layered networks have been attracting a lot of research attention in recent years. In [16] and [7], the authors presented an in-depth introduction on the fundamental concepts of interdependent, multi-layered networks as well as the key research challenges. In a multi-layered network, the failure of a small number of the nodes might lead to catastrophic damages on the entire system as shown in [3] and [22]. In [3], [15], [19], [18], [8], different types of *two-layered* interdependent networks were thoroughly analyzed. In [7], Gao et al. analyzed the robustness of multi-layered networks with star- and loop-shaped dependency structures. Similar to the robustness measures in [17], most of the current works use the size of GCC (giant connected component) in the network as the evaluation standard [14], [12], [2]. Nonetheless, the fine-granulated connectivity details might not be captured by the GCC measure.

VII. CONCLUSION

In this paper, we study the connectivity control problem on multi-layered networks (OPERA). Our main contributions

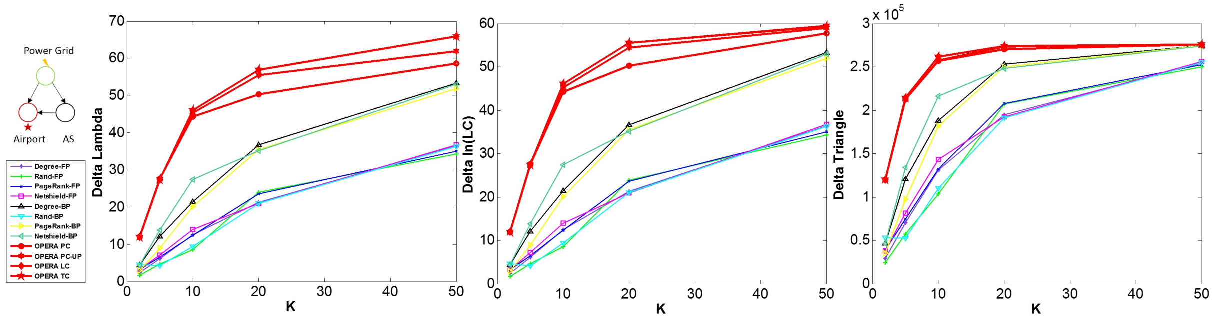


Fig. 4. Evaluations on the INFRANET data set, with a three-layered triangle-shaped dependency network. The connectivity change vs. budget. Larger is better. All the four instances of the proposed OPERA algorithm (in red) outperform the baseline methods.

are as follows. First, we propose a new data model for multi-layered networks (MULAN), which admits an arbitrary number of layers with a much more flexible dependency structure among different layers, beyond the current pair-wise dependency. Second, we unify a family of prevalent network connectivity measures (SUBLINE). Third, we show that for *any* network connectivity measure in the SUBLINE family, the optimal connectivity control problem with the proposed MULAN model enjoys the diminishing returns property, which naturally lends itself to a family of provable near-optimal control algorithms with linear complexity. Finally, we conduct extensive empirical evaluations on real network data, to validate the effectiveness of the proposed algorithms. In the future, we plan to generalize MULAN to an arbitrary layer-layer dependency network as well as the dynamic setting.

ACKNOWLEDGEMENT

This material is supported by the National Science Foundation under Grant No. IIS1017415, by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053, by Defense Advanced Research Projects Agency (DARPA) under Contract Number W911NF-11-C-0200 and W911NF-12-C-0028, by National Institutes of Health under the grant number R01LM011986, Region II University Transportation Center under the project number 49997-33 25.

The content of the information in this document does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

REFERENCES

[1] R. Albert, H. Jeong, and A.-L. Barabási. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[2] A. Bernstein, D. Bienstock, D. Hay, M. Uzunoglu, and G. Zussman. Power grid vulnerability to geographically correlated failuresanalysis and control implications. In *INFOCOM, 2014 Proceedings IEEE*, pages 2634–2642. IEEE, 2014.

[3] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin. Catastrophic cascade of failures in interdependent networks. *Nature*, 464(7291):1025–1028, 2010.

[4] D. Chakrabarti, Y. Wang, C. Wang, J. Leskovec, and C. Faloutsos. Epidemic thresholds in real networks. *ACM Transactions on Information and System Security (TISSEC)*, 10(4):1, 2008.

[5] H. Chan, L. Akoglu, and H. Tong. Make it or break it: manipulating robustness in large networks. In *Proceedings of 2014 SIAM International Conference on Data Mining*, pages 325–333. SIAM, 2014.

[6] C. Chen and H. Tong. Fast eigen-functions tracking on dynamic graphs. In *Proceedings of the 2015 SIAM International Conference on Data Mining*. SIAM, 2015.

[7] J. Gao, S. V. Buldyrev, S. Havlin, and H. E. Stanley. Robustness of a network of networks. *Physical Review Letters*, 107(19):195701, 2011.

[8] J. Gao, S. V. Buldyrev, H. E. Stanley, and S. Havlin. Networks formed from interdependent networks. *Nature physics*, 8(1):40–48, 2012.

[9] M. Jerrum and A. Sinclair. Conductance and the rapid mixing property for markov chains: the approximation of permanent resolved. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 235–244. ACM, 1988.

[10] W. Jun, M. Barahona, T. Yue-Jin, and D. Hong-Zhong. Natural connectivity of complex networks. *Chinese Physics Letters*, 27(7):078902, 2010.

[11] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 420–429. ACM, 2007.

[12] D. T. Nguyen, Y. Shen, and M. T. Thai. Detecting critical nodes in interdependent power networks for vulnerability assessment. *IEEE Trans. Smart Grid*, 4(1):151–159, 2013.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998. Paper SIDL-WP-1999-0120 (version of 11/11/1999).

[14] M. Parandehgheibi and E. Modiano. Robustness of interdependent networks: The case of communication networks and the power grid. In *Global Communications Conference (GLOBECOM), 2013 IEEE*, pages 2164–2169. IEEE, 2013.

[15] R. Parshani, S. V. Buldyrev, and S. Havlin. Interdependent networks: Reducing the coupling strength leads to a change from a first to second order percolation transition. *Physical review letters*, 105(4):048701, 2010.

[16] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly. Identifying, understanding, and analyzing critical infrastructure interdependencies. *Control Systems, IEEE*, 21(6):11–25, 2001.

[17] C. M. Schneider, A. A. Moreira, J. S. Andrade, S. Havlin, and H. J. Herrmann. Mitigation of malicious attacks on networks. *Proceedings of the National Academy of Sciences*, 108(10):3838–3841, 2011.

[18] A. Sen, A. Mazumder, J. Banerjee, A. Das, and R. Compton. Multi-layered network using a new model of interdependency. *arXiv preprint arXiv:1401.1783*, 2014.

[19] J. Shao, S. V. Buldyrev, S. Havlin, and H. E. Stanley. Cascade of failures in coupled network systems with multiple support-dependent relations. *arXiv preprint arXiv:1011.0234*, 2010.

[20] H. Tong, B. A. Prakash, T. Eliassi-Rad, M. Faloutsos, and C. Faloutsos. Gelling, and melting, large graphs by edge manipulation. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 245–254. ACM, 2012.

[21] H. Tong, B. A. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, and D. H. Chau. On the vulnerability of large graphs. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 1091–1096. IEEE, 2010.

[22] A. Vespignani. Complex networks: The fragility of interdependency. *Nature*, 464(7291):984–985, 2010.

[23] D. J. Watts and S. H. Strogatz. Collective dynamics of small-worldnetworks. *nature*, 393(6684):440–442, 1998.