Covert Network Detection

Steven T. Smith, Kenneth D. Senne, Scott Philips,

Edward K. Kao, and Garrett Bernstein

Covert network detection is an important capability in areas of applied research in which the data of interest can be represented as a relatively small subgraph in an enormous, potentially uninteresting background. This aspect characterizes covert network detection as a "Big Data" problem. In this article, a new Bayesian network detection framework is introduced that partitions the graph on the basis of prior information and direct observations. We also explore a new generative stochastic model for covert networks and analyze the detection performance of both classes of optimal detection techniques.

Network analysis has been a major research area over the last ten years, driven by interest in biological networks, cyber attacks, social networks, and criminal or terrorist organizations. This range of applications is illustrated in Figure 1. Detection of a covert community is most likely to be effective if the community exhibits high levels of connection activity. However, the covert networks of interest to many applications are unlikely to cooperate with this optimistic assumption. Indeed, a "fully connected network is an unlikely description of the enemy insurgent order of battle [1]." A clandestine or covert community is more likely to appear cellular and distributed [2]. Communities of this type can be represented with "small world" models [3]. The covert networks of interest in this paper exist to accomplish nefarious, illegal, or terrorist goals while "hiding in plain sight [4, 5]."

Covert networks necessarily adopt operational procedures to remain hidden and robustly adapt to losses of parts of the network. For example, during the Algerian Revolution, the National Liberation Front's (FLN) Autonomous Zone of Algiers (ZAA) military command was "carefully kept apart from other elements of the organization, the network was broken down into a number of quite distinct and compartmented branches, in communication only with the network chief," allowing ZAA leader Yassef Saadi to command "within 200 yards from the office of the [French] army commandant... and remain there several months [6]." Valdis Krebs' reconstruction of the 11 September 2001 terrorist network details the strategy for keeping cell members distant from each other and from other cells, and notes Osama

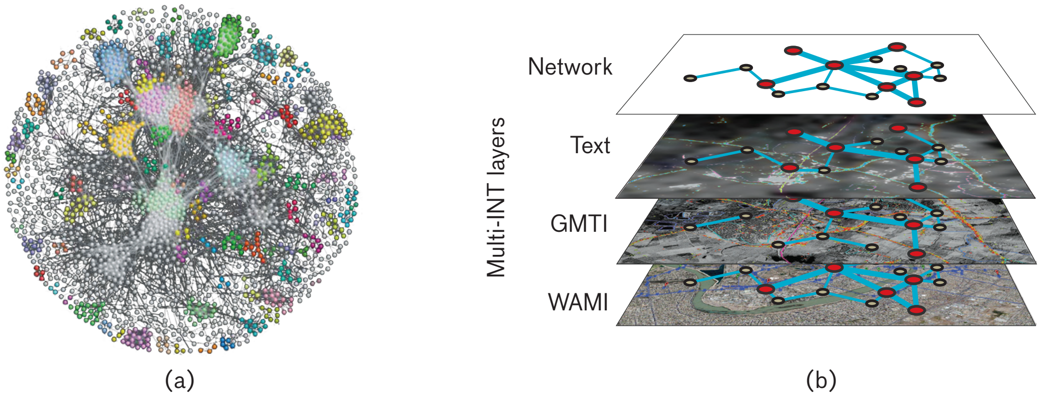


FIGURE 1. Network detection or discovery is the primary objective in many applications, including problems in intelligence, surveillance, and reconnaissance (ISR) and the frontiers of biological knowledge. This figure illustrates how networks of interactions are contructed from (a) protein-protein interactions and (b) multi-intelligent datasets, such as text communications, ground moving-target indication (GMTI), and wide-area motion imagery (WAMI).

bin Laden's description of this organization: "those… who were trained to fly didn't know the others. One group of people did not know the other group [7]." This type of organization is characterized by a *tree*, as shown in Figure 2. A covert network does not have to be human to be nefarious; the widespread Flashback malware attack on Apple's OS~X computers employed switched load balancing between servers to avoid detection [8], mirroring the ZAA's tree structure for robust covert network organization.

In order to accomplish its goals, the covert network must judiciously use "transitory shortcuts [9]." For example, in the 9/11 terrorism operation, after coordination meetings connected distant parts of the network, the "cross-ties went dormant [7]." It is during these occasional bursts of activity that a covert community may be most vulnerable to detection [1].

Because the connections between nodes are observable only when they are active, there are two basic strategies for detecting a covert threat: (1) subject-based Bayesian models that correlate a priori information or observations of the observed network connections; (2) pattern-based (predictive) methods that look for known patterns of organization and behavior to infer nefarious activity [4, 10]. Subject-based methods follow established principles of police investigations to accrue evidence based upon observed connections and historical data. The dependency of predictive methods on known patterns, however, makes them difficult to apply to rare and widely different covert threats: "there are no meaningful patterns

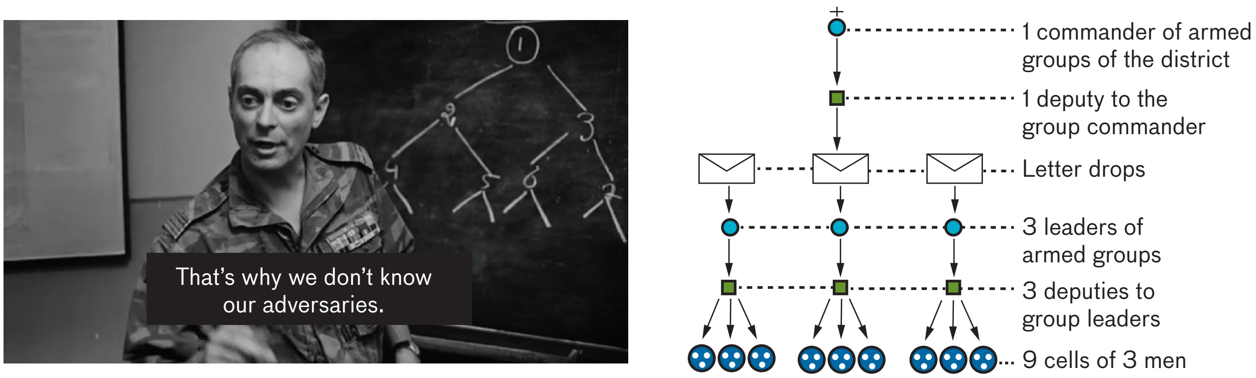

That's why we don't know our adversaries.
1 commander of armed groups of the district
1 deputy to the group commander
Letter drops
3 leaders of armed groups
3 deputies to group leaders
9 cells of 3 men

FIGURE 2. Covert networks are often organized as tree structures to avoid detection. The network illustrated is the FLN's Autonomous Zone of Algiers (ZAA) from Trinquier's monograph and the film Battle of Algiers [6, 11].

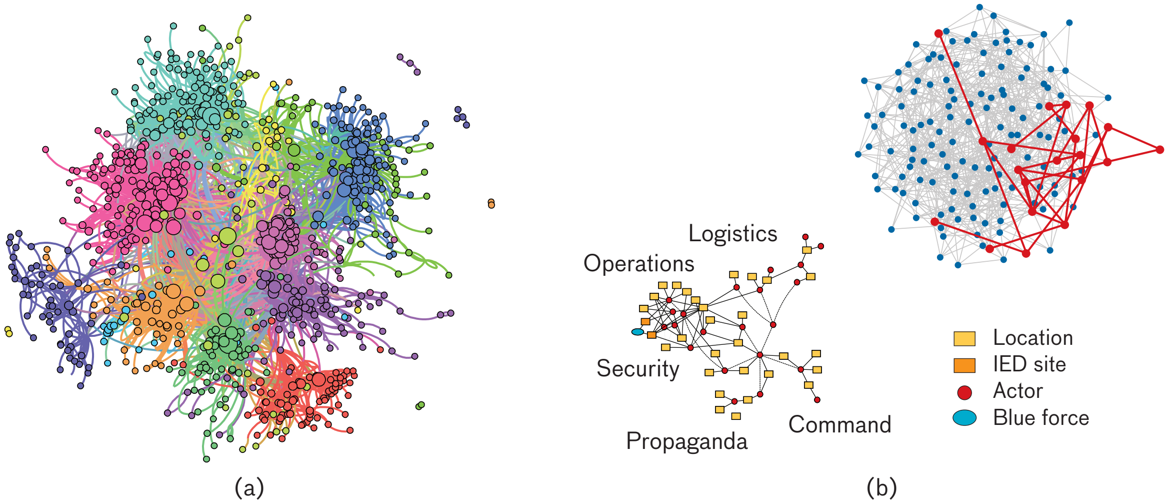


FIGURE 3. Network detection is a special case of the graph partitioning problem in which a graph must be divided into related subgraphs. The examples shown here are (a) the European Science in Society study [13] and (b) the foreground and representative background from the simulated insurgent network illustrated later in Figure 10.

that show what behavior indicates planning or preparation for terrorism [4].” The real-world consequences of applying an inappropriate model to detect a threat may include an unacceptable number of false positives and an erosion of individual privacy rights and civil liberties [4, 10].

The general problem of network detection is referred to as graph partitioning, wherein a large graph is subdivided into all of its component subgraphs. Covert network detection for discovering clandestine behavior is a special case in which the desired graph partition is binary: the covert subgraph is discovered on the basis of the existence of observed activities (see Figure 3). In this article, the focus will be on observations of network activities using intelligence, surveillance, and reconnaissance (ISR) sensors, such as wide-area motion imagery (WAMI). Covert networks engaged in terrorist attacks with improvised explosive devices (IEDs) comprise loosely connected cells with various functions, such as finance, planning, operations, logistics, security, and propaganda.

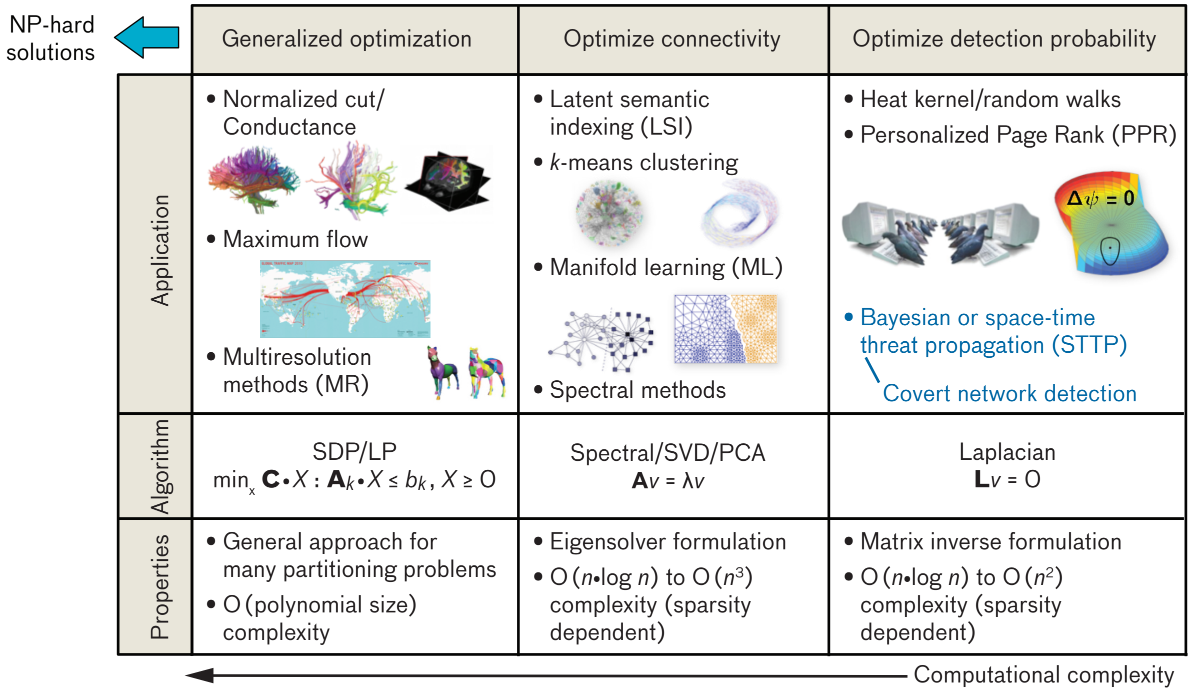
A new model of covert threat for detection analysis that accounts for the realities of dynamic foreground networks in large backgrounds is a specially adapted version of a mixed-membership stochastic block model [12]. The terrorist cells of interest are embedded into a background consisting of many “neutral” communities that represent business, homes, industry, religion, sports, etc. Because in real life people wear different “hats” depending upon the communities with which they interact, their proportions of membership in multiple communities (lifestyles) can be adjusted to control the occasional coordination between the foreground and background networks. The new generative block model approach introduced in the section entitled “Network Models and Performance” leads to an analytically tractable tool with sufficient parameters to exhibit realistic coordinated activity levels and interactions.

Network Detection

Network Detection is Graph Partitioning

Network detection is a special case of the general graph-partitioning (GP) problem in which the parts of a graph must be divided into a set of similar classes. There are simply two classes for network detection: membership or non-membership. In general, there could be many classes, and typically graph partitioning is an “NP-hard” problem, meaning that GP almost certainly cannot be solved in “polynomial time” because of GP’s intrinsically combinatoric characteristics, and that GP has prohibitively terrible scaling properties as the size of the graph grows. Fortunately, the solutions to a great many GP problems may be cast as approximate solutions to various optimization problems that are solvable in polynomial time and thereby possess practical computational and scaling properties (Figure 4).

The solution to many special cases of graph partitioning may be cast as a semidefinite-programming



	Generalized optimization	Optimize connectivity	Optimize detection probability
Application	• Normalized cut/Conductance • Maximum flow • Multiresolution methods (MR)	• Latent semantic indexing (LSI) • *k*-means clustering • Manifold learning (ML) • Spectral methods	• Heat kernel/random walks • Personalized Page Rank (PPR) • Bayesian or space-time threat propagation (STTP)
Algorithm	SDP/LP $\min_x \mathbf{C}\bullet X : \mathbf{A}_k\bullet X \le b_k,\ X \ge 0$	Spectral/SVD/PCA $\mathbf{A}v = \lambda v$	Laplacian $\mathbf{L}v = 0$
Properties	• General approach for many partitioning problems • O (polynomial size) complexity	• Eigensolver formulation • $O(n \bullet \log n)$ to $O(n^3)$ complexity (sparsity dependent)	• Matrix inverse formulation • $O(n \bullet \log n)$ to $O(n^2)$ complexity (sparsity dependent)

FIGURE 4. A taxonomy of network-detection algorithms, ranging from general graph partitioning methods that solve a semidefinite-programming problem, to spectral methods that optimize connectivity, to Bayesian or Laplacian methods that optimize detection probability. This article focuses on local spectral and harmonic methods for network detection. (Images used from various sources; clockwise from upper left [19–24]).

(SDP) method, offering both practical and oftentimes theoretically attractive approximation to GP [14, 15]. In general, practical GP approaches exploit a variety of global and local connectivity properties to divide a graph into many subgraphs. Decreasing algorithmic complexity is achieved in certain domains that may be cast as quadratic optimization problems that yield eigenvalue- or spectral-based methods. The lowest-cost graph partitioning problems involve diffusion on the graph, which is solved with a set of linear equations. Network detection may be cast as either an optimization problem in which a connectivity measure of the detected network is optimized, or as a Bayesian detection problem in which the probability of network detection is optimized given a set of observations. Approaches based on optimizing connectivity are generally called *community detection*, and Bayesian approaches based on diffusion models and optimizing detection probability are called *threat propagation* [16–18].

This article describes these approaches to network detection applied to covert networks. It is demonstrated that space-time threat propagation (STTP) optimizes the probability of network detection in a Neyman-Pearson sense, given prior information and/or direct observations, meaning that the detection probability is maximized given a fixed false alarm, also known as a false-positive probability. This property is important because it provides a practical optimum algorithm in many settings, and it provides a performance bound on detection performance. Remarkably, the two apparently different optimal network detection approaches are related to each other by using insights from algebraic graph theory. Both spectral-based and Neyman-Pearson network-detection methods are described in the following section. Network-detection performance is assessed using a new stochastic block model for small, dynamic foreground networks embedded within a large background [12].

Optimum Network Detection

The objective of any network-detection method is the computation of the degree to which each vertex in a graph belongs to a network. In Bayesian network detection, for example, the probability of threat at a particular vertex is determined. Because the likelihood of threat at every ver-

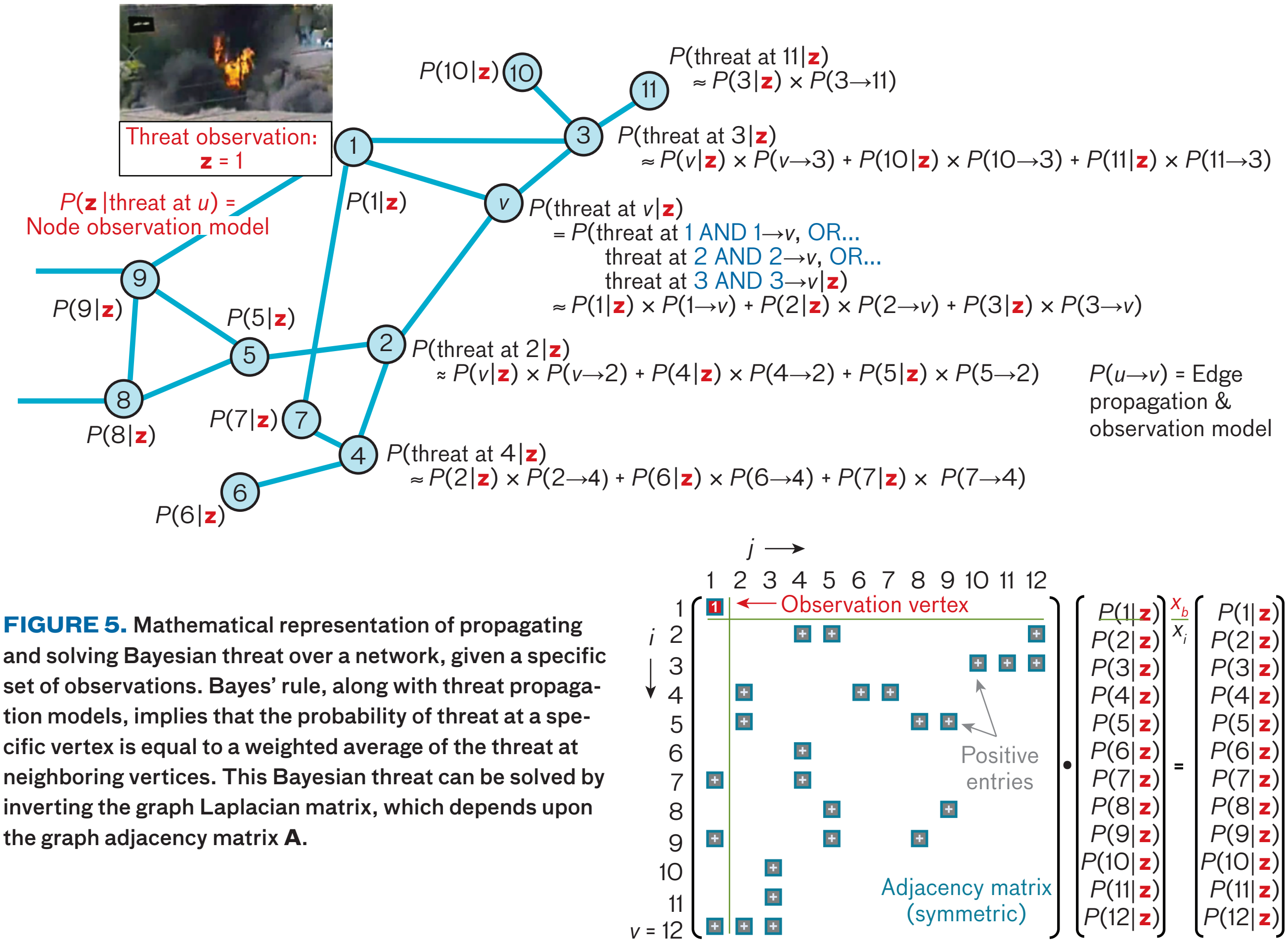

Threat observation:
z = 1
P(z |threat at u) =
Node observation model
P(10|z)
P(threat at 11|z)
≈ P(3|z) × P(3→11)
P(threat at 3|z)
≈ P(v|z) × P(v→3) + P(10|z) × P(10→3) + P(11|z) × P(11→3)
P(1|z)
P(threat at v|z)
= P(threat at 1 AND 1→v, OR...
threat at 2 AND 2→v, OR...
threat at 3 AND 3→v|z)
≈ P(1|z) × P(1→v) + P(2|z) × P(2→v) + P(3|z) × P(3→v)
P(9|z)
P(5|z)
P(threat at 2|z)
≈ P(v|z) × P(v→2) + P(4|z) × P(4→2) + P(5|z) × P(5→2)
P(u→v) = Edge propagation & observation model
P(8|z)
P(7|z)
P(threat at 4|z)
≈ P(2|z) × P(2→4) + P(6|z) × P(6→4) + P(7|z) × P(7→4)
P(6|z)
1 2 3 4 5 6 7 8 9 10 11 v
j →
1 2 3 4 5 6 7 8 9 10 11 12
i ↓
1 2 3 4 5 6 7 8 9 10 11 v = 12
Observation vertex
Positive entries
Adjacency matrix (symmetric)
P(1|z) P(2|z) P(3|z) P(4|z) P(5|z) P(6|z) P(7|z) P(8|z) P(9|z) P(10|z) P(11|z) P(12|z)
x_b
x_i
=
P(1|z) P(2|z) P(3|z) P(4|z) P(5|z) P(6|z) P(7|z) P(8|z) P(9|z) P(10|z) P(11|z) P(12|z)

FIGURE 5. Mathematical representation of propagating and solving Bayesian threat over a network, given a specific set of observations. Bayes' rule, along with threat propagation models, implies that the probability of threat at a specific vertex is equal to a weighted average of the threat at neighboring vertices. This Bayesian threat can be solved by inverting the graph Laplacian matrix, which depends upon the graph adjacency matrix **A**.

tex necessarily depends upon its neighbors connected to that vertex in the graph, the probability of threat at every vertex is a function of the probability of threat of the vertices' neighbors. A graph $G = (V, E)$ is defined by two sets, the vertices V of G, and the edges $E \subset [V]^2 \subset 2^V$ of G, in which $[V]^2$ denotes the set of 2-element subsets of V [25]. For example, the sets $V = \{1, 2, 3\}$ and $E = \{\{1, 2\}, \{2, 3\}\}$ describe a simple graph with undirected edges between vertices 1 and 2, and 2 and 3: ①–②–③.

SPACE-TIME THREAT PROPAGATION

Consider the probability of threat at vertex v in Figure 5. Under independence and first-order approximation assumptions [17], the probability of threat at vertex v given an observation **z**, denoted $P(v \mid \mathbf{z})$, is represented by the linear approximation

$$P(v \mid \mathbf{z}) \approx P(1 \mid \mathbf{z}) P(1 \rangle v) + P(2 \mid \mathbf{z}) P(2 \rangle v) + P(3 \mid \mathbf{z}) P(3 \rangle v),$$

where, as illustrated in the example in Figure 5, v is connected to the vertices 1, 2, and 3. The probability $P(u \rangle v)$ represents the probability that the threat moves along the connection from u to v and is called the propagation model. By applying this argument to the remaining vertices in Figure 5, the probability of threat at vertices 2, 3, ... is given by the linear equations

$$P(2 \mid \mathbf{z}) \approx P(v \mid \mathbf{z}) P(v \rangle 2) + P(4 \mid \mathbf{z}) P(4 \rangle 2) + P(5 \mid \mathbf{z}) P(5 \rangle 2)$$

$$P(3 \mid \mathbf{z}) \approx P(v \mid \mathbf{z}) P(v \rangle 3) + P(10 \mid \mathbf{z}) P(10 \rangle 3) + P(11 \mid \mathbf{z}) P(11 \rangle 3)$$

$$\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots$$

At the vertex or vertices with an observed threat, the threat is fixed by using the observation model $P(1 \mid \mathbf{z}) \propto P(\mathbf{z} \mid 1)$. The most basic model is $P(1 \mid \mathbf{z}) \equiv 1$ if there is a threat observed at node 1. There are, in general, time stamps associated with each edge, and space-time threat propagation over these edges and times is modeled with a jump process, yielding a space-time threat kernel [17]. This system of linear equations determines the unknown threat probabilities at vertices for which no threats are directly observed but are connected by one or more hops through the graph to observed threats.

GRAPH THEORY

The set of linear equations that defines the threat probabilities is easily solved by expressing these equations using matrix notation (Figure 5). The set of threat probabilities forms the column vector

$$\vartheta = \begin{pmatrix} P(1\,|\,\mathbf{z}) \\ P(2\,|\,\mathbf{z}) \\ \vdots \\ P(12\,|\,\mathbf{z}) \end{pmatrix},$$

where in the example the vertex v is numbered as vertex 12. As illustrated in Figure 5, the square matrix $\mathbf{A}$ is formed whose elements $(i\,,j\,)$ are set to 1 along row i if vertex i is connected to vertex j. Otherwise, the matrix is zero if there is no direct connection between i and j. This matrix is called the adjacency matrix of the graph. It is a symmetric matrix ($\mathbf{A}_{ij} = \mathbf{A}_{ji}$); if vertex i is connected to vertex j implies vertex j is connected to vertex i. It is also typically a sparse matrix because most elements of this matrix equal zero. A simple threat propagation model is the nearest-neighbor average assumption $P(u\,\rangle\,v\,) \propto d(u)^{-1}$ in which $d(u)$ is the number of neighbors of vertex u called the degree of u. The diagonal degree matrix $\mathbf{D} = \operatorname{diag}(d(1), d(2), \ldots, d\,(12))$ accounts for the degrees of all vertices in the graph. Using the adjacency and degree matrices, the system of linear equations that determines the threat probabilities is written as the matrix-vector multiplication

$$\mathbf{D}^{-1}\mathbf{A}\,\vartheta = \vartheta\ ,$$

where $\mathbf{D}^{-1} = \operatorname{diag}(d(1)^{-1}, d(2)^{-1}, \ldots, d\,(12)^{-1})$ is the inverse of the diagonal matrix such that $\mathbf{D}^{-1}\mathbf{D} = \mathbf{I}$ and $\mathbf{I} = \operatorname{diag}(1, \ldots, 1)$ is the identity matrix. This matrix equation is solved for the threat probability vector ϑ by subtracting $\mathbf{D}^{-1}\mathbf{A}\vartheta$ from both sides of the equation to obtain the equation

$$\vartheta - \mathbf{D}^{-1}\mathbf{A}\,\vartheta = (\mathbf{I} - \mathbf{D}^{-1}\mathbf{A})\,\vartheta = \mathbf{\L}\,\vartheta.$$

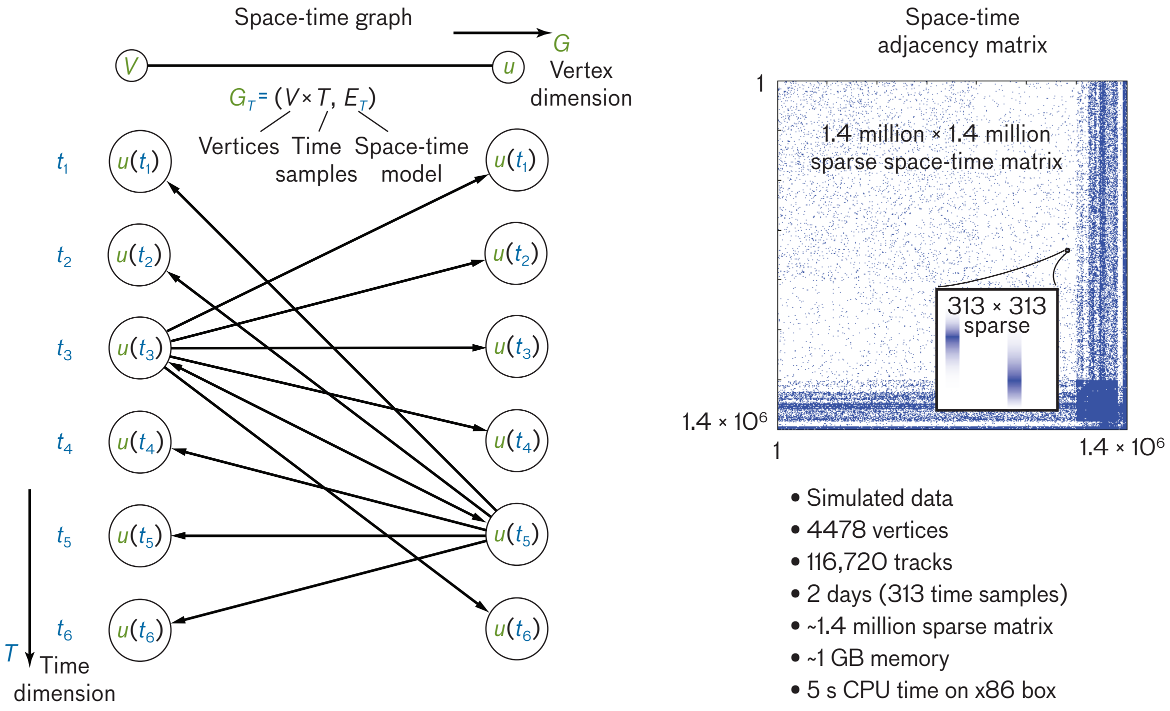
Finally, the unknown threat probabilities are determined by treating this "harmonic" system of equations as a boundary value problem in which the observed cue vertices are treated as known *boundary* values, and the unknown vertices are treated as *interior* points whose values must be computed. This separation between vertices with observations (marked with ⊞) and those without (unmarked) is also depicted in Figure 5.

The matrix $\mathbf{\L} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$ is called the (asymmetric) Laplacian matrix. It is related to the graph Laplacian matrix $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2} = \mathbf{D}^{-1/2}\mathbf{\L}\mathbf{D}^{-1/2}$ and the graph Kirchoff matrix $\mathbf{Q} = \mathbf{D} - \mathbf{A} = \mathbf{D}^{1/2}\,\mathbf{L}\mathbf{D}^{1/2}$ via matrix similarity and congruence transformations [26]. This fact is of fundamental importance because it connects Bayesian threat propagation to a multitude of problems in graph theory, network detection methods, and harmonic analysis. The graph Laplacian $\mathbf{L}$ is the discretized Laplacian operator $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2 + \cdots$ that appears in numerous physical applications, and the asymmetric Laplacian $\mathbf{\L}$ plays an important role in mean-value theorems involving solutions to Laplace's equation $\mathbf{\L}\vartheta = 0$. This harmonic equation will be seen to be the motivating equation behind several network-detection algorithms. Given a cue at boundary vertices represented by the threat vector ϑ_b, the *harmonic threat* at the remaining interior vertices is the solution

$$\vartheta_i = \mathbf{\L}_{ii}^{-1}(\mathbf{\L}_{ib}\vartheta_b).$$

This highly sparse linear system may be solved by iterative methods such as the biconjugate gradients, which provide a practical computational approach that scales well to graphs with thousands of vertices and thousands of time samples, resulting in space-time graphs of order 10 million or more. In practice, significantly smaller subgraphs are encountered in applications such as threat network discovery, for which linear solvers with sparse systems are extremely fast [27].

Many important network detection applications, especially networks based on vehicle tracks and computer communication networks, involve directed graphs in which the edges have departure and arrival times associated with their initial and terminal vertices [17]. In such scenarios, the time-stamped graph $G = (V, E)$ may be viewed as a space-time graph $G_T = (V \times T, E_T\,)$ in which T is the set of sample times and $E_T \subset [V \times T]^2$



- Simulated data
- 4478 vertices
- 116,720 tracks
- 2 days (313 time samples)
- ~1.4 million sparse matrix
- ~1 GB memory
- 5 s CPU time on x86 box

FIGURE 6. A directed space-time graph represents relationships between different vertices at different times. Time-stamped tracks or connections between entities determine a space-time graph. The example space-time graph shown on the right is the insurgent network illustrated later in Figure 10.

is an edge set determined by the temporal correlations between vertices at specific times. Two space-time graphs are illustrated in Figure 6.

NEYMAN-PEARSON OPTIMALITY

Detection optimality means optimality in the Neyman-Pearson sense in which the probability of detection is maximized at a fixed false-alarm rate. In the context of networks, the probability of detection (PD) refers to the fraction of vertices detected belonging to the threat subgraph, and the probability of false alarm (PFA) refers to the fraction of non-threat vertices detected. As in classical detection theory [28], the optimal detector is a threshold of the log-likelihood ratio (LLR). A new Bayesian framework for network detection is developed in this section. The distinction between classical detection theory and network detection theory is not in the form of the optimal detector—the log-likelihood ratio—but in distinct mathematical formulations. Whereas linear algebra is the foundation for classical detection theory, algebraic graph theory is the foundation for network detection, as evidenced earlier [29].

Network detection of a subgraph within a graph $G = (V, E)$ of order n is treated as n independent binary hypothesis tests to decide which of the graph's n vertices do belong or do not belong (null hypothesis H_0 or hypothesis H_1) to the network. Maximizing the probability of detection (PD) for a fixed probability of false alarm (PFA) yields the Neyman-Pearson test involving the log-likelihood ratio of the competing hypothesis. The optimal Neyman-Pearson detector is given by the likelihood ratio (LR) test,

$$\frac{f(\mathbf{z}|\Theta_v = 1)}{f(\mathbf{z}|\Theta_v = 0)} \underset{H_0(v)}{\overset{H_1(v)}{\gtrless}} \lambda .$$

The numerator $f(\mathbf{z}|\Theta_v = 1)$ is easily computed using standard Bayesian analysis, leading to the threat propagation algorithm for $f(\Theta_v|\mathbf{z})$ and a connection to the Laplacian **Ł** described earlier in this section, and the denominator $f(\mathbf{z}|\Theta_v = 0)$ is determined by prior background information or simply the "principle of insufficient reason" in which this term is a constant [30]. An application of Bayes' theorem to the harmonic threat provides the optimum Neyman-Pearson detector because it results in a threshold of the harmonic space-time threat propagation vector

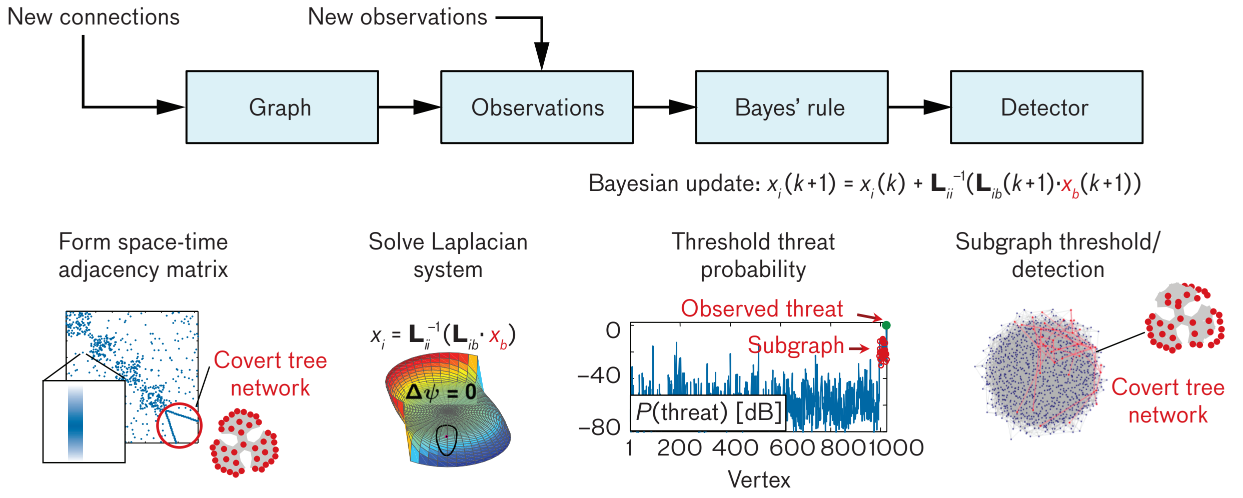

New connections
New observations
Graph
Observations
Bayes' rule
Detector
Bayesian update: $x_i(k+1) = x_i(k) + \mathbf{L}_{ii}^{-1}(\mathbf{L}_{ib}(k+1)\cdot x_b(k+1))$
Form space-time adjacency matrix
Covert tree network
Solve Laplacian system
$x_i = \mathbf{L}_{ii}^{-1}(\mathbf{L}_{ib}\cdot x_b)$
$\Delta\psi = 0$
Threshold threat probability
Observed threat
Subgraph
0
−40
−80
P(threat) [dB]
1
200
400
600
800
1000
Vertex
Subgraph threshold/ detection
Covert tree network

FIGURE 7. The block diagram of space-time threat propagation involves forming a graph, making observations at specific graph vertices and times, applying Bayes' rule to determine the probability of threat over all vertices and time, then applying a threshold to this probability and declaring a threat network.

$$\vartheta \underset{H_0}{\overset{H_1}{\gtrless}} \text{threshold},$$

possibly weighted by a nonuniform null distribution that provides an optimum network detection method [17].

The block diagram for threat propagation is illustrated in Figure 7. The graph is constructed from knowledge of new vertices and edges. Specific observations at any of these vertices are incorporated by using the observation model. Bayes' rule is applied, and the solution to the harmonic space-time threat is computed as a function of both space and time, yielding the (unnormalized) probability of threat for all vertices and times. Finally, this probability is thresholded by using the Neyman-Pearson likelihood ratio test, and the most likely vertices are declared to be the detected network.

Community Detection

The spectral methods in this section solve the graph partitioning problem by optimizing various subgraph connectivity properties. Efficient graph-partitioning algorithms and analysis appeared in the 1970s with Donath and Hoffman's eigenvalue-based bounds for graph-partitioning [31] and Fiedler's connectivity analysis and graph-partitioning algorithm [32, 33], which established the connection between a graph's algebraic properties and the spectrum of its Kirchhoff Laplacian matrix $\mathbf{Q} = \mathbf{D} - \mathbf{A}$.

The *cut size* of a subgraph—the number of edges necessary to remove to separate the subgraph from the graph—is quantified by the quadratic form $\mathbf{s}^{\mathrm{T}}\mathbf{Q}\mathbf{s}$, where $\mathbf{s} = (\pm 1, \cdots, \pm 1)^{\mathrm{T}}$ is a ± 1 vector whose entries are determined by subgraph membership [34]. Minimizing this quadratic form over $\mathbf{s}$, whose solution is an eigenvalue problem for the graph Laplacian, provides a network-detection algorithm based on the model of minimal cut size. However, there is a paradox in the application of spectral methods to network detection: the smallest eigenvalue of the graph Laplacian $\lambda_0(\mathbf{Q}) = 0$ corresponds to the eigenvector $\mathbf{1}$ (constant over all vertices) that fails to discriminate between subgraphs. Intuitively, this degenerate constant solution makes sense because the two subgraphs with minimal (zero) subgraph cut size are the entire graph itself, $\mathbf{s} \equiv \mathbf{1}$, or the null graph $\mathbf{s} \equiv -\mathbf{1}$. This property manifests itself in many well-known results from complex analysis, such as the maximum principle.

Fiedler also showed that if instead of using the zeroth eigenvector, rather the eigenvector ξ_1 corresponding to the second smallest eigenvalue $\lambda_1(\mathbf{Q})$ of $\mathbf{Q}$ is used, then for every nonpositive constant $c \le 0$, the subgraph whose vertices are defined by the threshold $\xi_1 \ge c$ is necessarily connected. This algorithm is called *spectral detection*. Given a graph G, the number $\lambda_1(\mathbf{Q})$ is called the *Fiedler value* of G, and the corresponding eigenvector $\xi_1(\mathbf{Q})$ is called the *Fiedler vector*. The Fiedler value is also called the *algebraic connectivity* because the greater the Fiedler value, the smaller the graph diameter, implying greater graph connectivity.

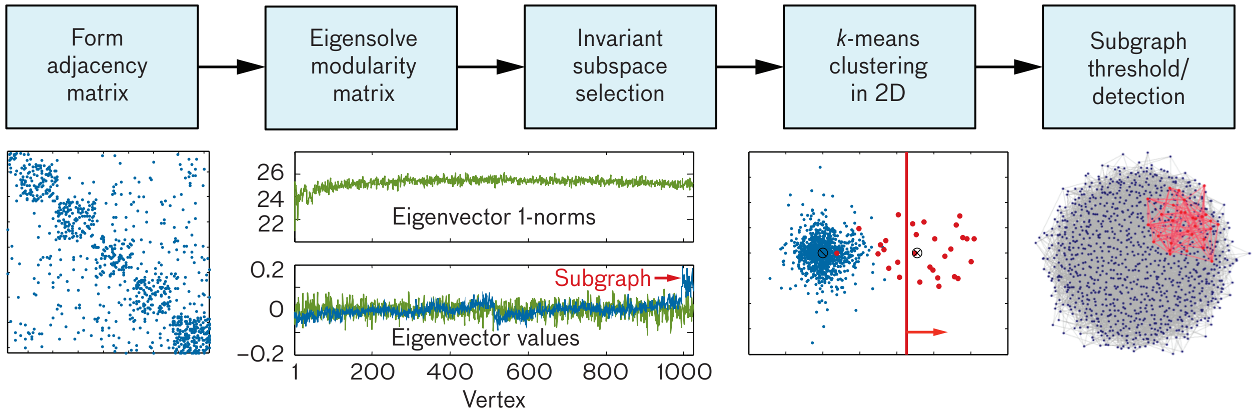


FIGURE 8. The block diagram for a representative community-detection algorithm involves forming the adjacency matrix of a graph, computing its eigenvectors, selecting and projecting onto a two-dimensional (2D) invariant subspace, performing *k*-means clustering in this 2D subspace, then detecting a network by using the smallest *k*-means cluster.

Newman introduced the alternate criterion of subgraph "modularity" for subgraph detection [35] because, in practice, spectral detection with its implicit assumption of minimizing the cut size oftentimes does not detect an intuitively appealing subgraph. Rather than minimizing the cut size, Newman modularity maximizes the subgraph connectivity relative to background graph connectivity, which yields the quadratic maximization problem $\max_{\mathbf{s}} \mathbf{s}^{\mathrm{T}}\mathbf{M}\mathbf{s}$, where $\mathbf{M} = \mathbf{A} - V^{-1}\mathbf{d}\mathbf{d}^{\mathrm{T}}$ is Newman's *modularity matrix*, $\mathbf{A}$ is the adjacency matrix, $(\mathbf{d})_i = d_i$ is the degree vector, and $V = \mathbf{1}^{\mathrm{T}}\mathbf{d}$ is the graph volume [35]. Newman's modularity-based graph-partitioning algorithm, also called community detection, involves thresholding the values of the principal eigenvector of $\mathbf{M}$, defined by columns of the matrix $\mathbf{U}$ in the equation

$$\mathbf{M} := \mathbf{U}\Lambda\mathbf{U}^{\mathrm{T}}.$$

Miller et al. [36–38] also consider thresholding arbitrary eigenvectors of the modularity matrix, which by the Courant minimax principle biases the Newman community-detection algorithm to smaller subgraphs, a desirable property for many applications. They also outline an approach for exploiting observations within the spectral framework [36].

The block diagram for community detection is illustrated in Figure 8. The graph is constructed from knowledge of new vertices and edges. The modularity matrix is formed, implicitly or explicitly, from the graph's adjacency matrix, and eigensolvers are used in conjunction with a subspace selection algorithm to compute a two-dimensional scatterplot. The resulting scatterplot is clustered by using a *k*-means clustering algorithm, and the smaller cluster is declared to be the detected network.

Network Models and Performance

Analyzing and predicting the performance of detection methods are essential for data collection and system design. Currently, only a few limited methods are used to assess network-detection performance, though several areas for future research are promising. There are two ways to demonstrate network-detection performance: empirical and theoretical, both of which depend on detailed knowledge of network behavior and dynamics. Figure 9 illustrates the challenges in assessing network-detection performance for covert networks—a notional representation of the performance fidelity is plotted against the number of real-world or simulation cases currently available. Ideally, we would like full knowledge of an entire covert network and its activities, including its connectivity to the benign or "gray" background network and multiple real-world examples from diverse scenarios. However, the authors are unaware of even a single example of such a dataset, not surprisingly because full knowledge of real-world covert network behavior is, by design, extraordinarily rare or nonexistent. In some cases, partial information about covert networks has been integrated over time [5]. Empirical detection performance is demonstrated with either a real-world or simulated dataset for which the truth is at least partially known, and theoretical performance predictions are derived from statistical assumptions about the foreground

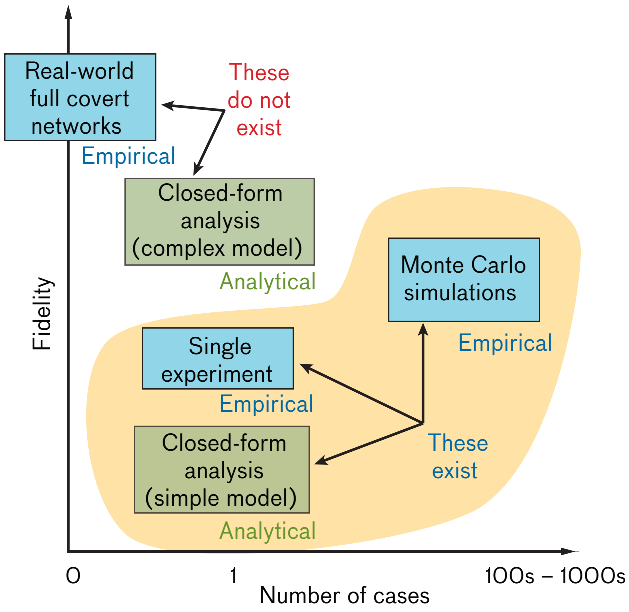


FIGURE 9. The challenges assessing network-detection performance depend upon the desired fidelity of the performance prediction and the number of available cases. Empirically, more cases provide greater statistical significance and therefore higher fidelity. Analytically, closed-form analysis of realistic network models also provides higher fidelity. At present, there are no full real-world covert networks known, and no closed-form results for complex network models, leaving only single experiments, Monte Carlo methods, and closed-form analysis using very simple models.

and background networks. To date, closed-form analytic performance predictions have been accomplished for very simple network models, e.g., cliques embedded within Erdös-Rényi backgrounds [39–42], and there are no theoretical results at all for space-time graphs or realistic models appropriate for covert networks. Therefore, realistic models are essential for performance analysis of network-detection algorithms. There are two basic approaches to modeling networks: stochastic models, which attempt to capture the aggregate statistical properties of networks, and agent-based models, which attempt to describe specific behaviors. In general, stochastic models have greater tractability because they do not rely on the detailed description of actions or objectives of a specific network.

The empirical detection performance of the covert network-detection algorithms described earlier will be computed using a Monte Carlo analysis based upon a new stochastic block model. Empirical performance predictions may also be based on a single dataset, oftentimes a practical necessity for real-world measurements. Detection performance for specific, real-world single datasets is illustrated in an accompanying article entitled “Network Discovery with Multi-intelligence Sources” [43].

Empirical Data Performance

A simulated empirical dataset is used to evaluate space-time threat propagation performance [27]. These data are derived from a scripted scenario performed by the Institute for Defense Analyses (IDA) that contains a clandestine insurgent network, illustrated in Figure 10. The simulated data covers a 48-hour time period and consists of approximately 116,720 vehicle tracks between 4478 locations made by 4623 individual actors. Of these, 31 locations and 22 actors are part of the insurgent network. The simulated tracks in these data are perfectly accurate and unambiguous. The receiver operating characteristic (ROC) plot on the right-hand side of Figure 10 shows the performance of the space-time threat propagation algorithm by quantifying the fraction of the true covert network detected as a function of the number or rate of false alarms from the background. In this particular example, a 50% probability of detection is achievable with a 2% false-alarm number. This quantification of detection performance provides insight into examples in which these data are representative; however, a single example is not statistically significant and does not allow generalization to different scenarios.

Covert Network Stochastic Block Model

Real-world networks are sparse, have power-law degree distribution, and display community structure based on membership. Furthermore, activities on these networks tend to be coordinated (e.g., meetings and dispatches). To study detection performance adhering to real-world applications, we developed a generating model that captures these network phenomenologies. Our model is described in the plate diagram in Figure 11. We discuss the interaction model and the temporal model separately in the next sections. (Greater details and more results on detection performance using this model are available in [17].)

INTERACTION MODEL

The interaction model combines three well-known network models: (1) Erdös-Rényi for sparsity [44]; (2) Chung-Lu for power-law degree distribution [45]; (3) mixed-membership stochastic block model for com-

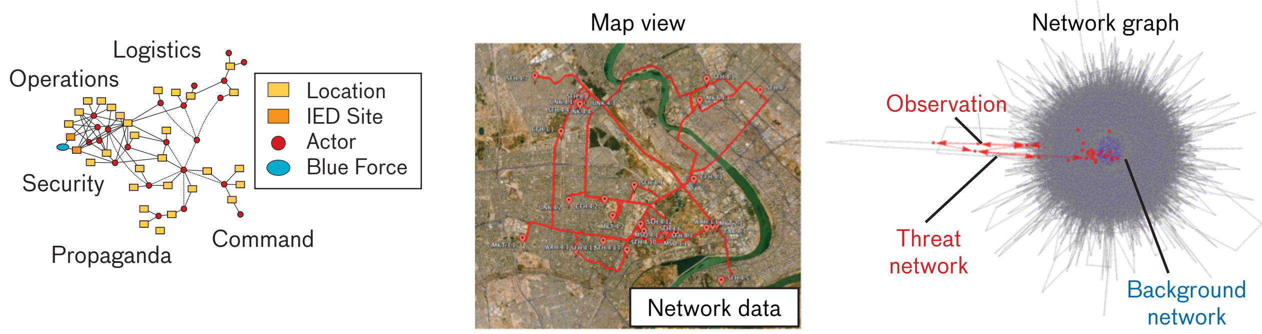


Image source material © DigitalGlobe

munity structure [12]. Specifically, the rate λ_{ij} of interactions between nodes i and j is given by the product

$$\lambda_{ij} = I^{\mathbf{S}}_{ij} \cdot \frac{\lambda_i \lambda_j}{\sum_k \lambda_k} \cdot \mathbf{z}^{\mathbf{T}}_{i\to j}\mathbf{B}\,\mathbf{z}_{i\to j},$$

where the term $\boldsymbol{I}^{\mathbf{S}}_{ij}$ represents the (modified) Erdös-Rényi model, the term $\lambda_i\lambda_j / \sum_k \lambda_k$ represents the Chung-Lu model, and the term $\mathbf{z}^{\mathbf{T}}_{i\to j}\mathbf{B}\,\mathbf{z}_{i\to j}$ represents the stochastic block model.

The indicator function $\boldsymbol{I}^{\mathbf{S}}_{ij}$ turns an edge on with probability p on the basis of the memberships of node i and j; p is typically small to achieve sparsity. The Chung-Lu term $\lambda_i\lambda_j / \sum_k \lambda_k$ is determined by the per-node expected degrees λ_i, $i = 1, \ldots, N$, which are themselves drawn from a power-law distribution of parameter $\alpha \in \mathbb{R}$. This term makes highly active nodes interact more. The block model term $\mathbf{z}^{\mathbf{T}}_{i\to j}\mathbf{B}\,\mathbf{z}_{i\to j}$ introduces community structure between nodes with mixed membership. This value is high if the community of i interacts frequently with the community of j. Specifically, it is determined by $\mathbf{B}$, a K-by-K matrix of the rate of interaction between communities $\mathbf{z}_{i\to j} \in \mathbb{R}^K$ and $\mathbf{z}_{j\to i} \in \mathbb{R}^K$. $\mathbf{z}_{i\to j} \in \mathbb{R}^K$, an indicator (01)-vector, is the community to which node i belongs when interacting with node j. Similarly, $\mathbf{z}_{j\to i} \in \mathbb{R}^K$ is the community to which node j belongs when interacting with node i. $\mathbf{z}_{i\to j}$ and $\mathbf{z}_{j\to i}$ are drawn from a multinomial over $\pi_i \in \mathbb{R}^K$ and $\pi_j \in \mathbb{R}^K$, the i and j node distribution over communities. Finally, the distribution of π is drawn from a Dirichlet random variable, with concentration parameter $\mathbf{I}_i^{\mathrm{T}}\mathbf{X}$. Node i's lifestyle, $\mathbf{I}_i$, is a multinomial drawn with the lifestyle probability $\phi \in \mathbb{R}^L$. This way, each node may belong to a number of communities (i.e., mixed membership) with its proportion of membership π_i depending on its lifestyle $\mathbf{I}_i$. However, when two nodes interact, they each assume a certain community membership $\mathbf{z}_{i\to j}$ and $\mathbf{z}_{j\to i}$.

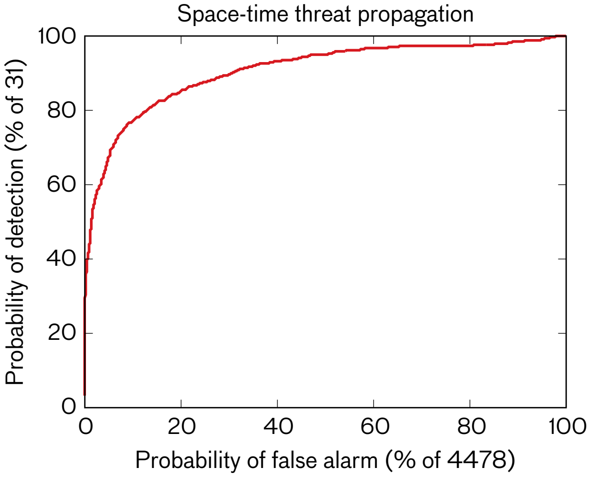


FIGURE 10. A single simulated experiment, shown on the left and embedded in the center map, is generated to assess the receiver operating characteristic performance of various network detection methods in a wide-area motion imagery example [27]. The simulated insurgent network graph is composed of 4478 locations and 116,720 tracks. The foreground subgraph is shown using red nodes and edges, and the background graph is shown using blue nodes and gray edges. For clarity, only a partial graph is shown here.

TEMPORAL MODEL

After the number of interactions, m_{ij}, between nodes are drawn, the time for each interaction is generated. Real-world interactions are often coordinated, with many individuals arriving at or leaving from a location at a set of predefined times. We generate this behavior as a Poisson process, parameterized by an average number of meeting times $\Psi \in \mathbb{R}^K$ for each community. Smaller ψ_k yields a community whose activities are tightly coordinated because there are only a couple of times for the members to meet. On the other hand, a higher expected number of meeting times (e.g., $\psi_k = 20$) yields a community whose activities are

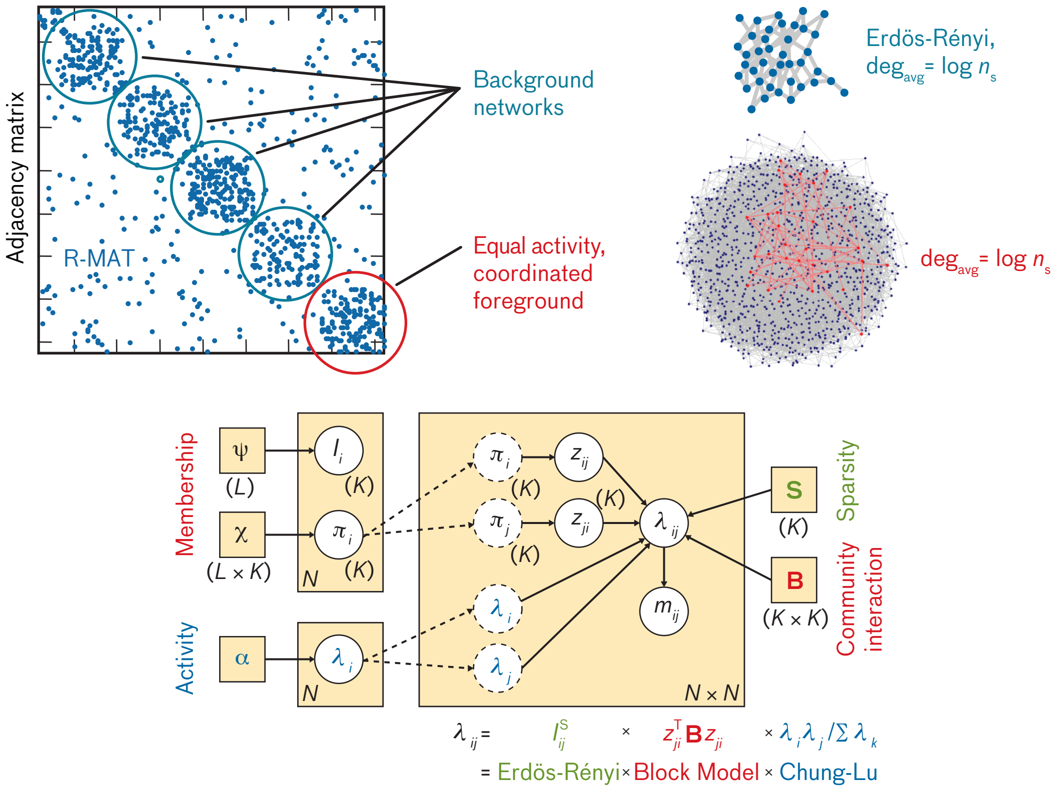


FIGURE 11. Stochastic block models of varying complexity are used to simulate realistic network-detection problems. The model on the top uses connected foreground networks embedded within a much larger connected graph. The generalized model on the bottom uses a mixed-membership model along with standard background models to control connectivity, scale-free degree distribution properties, community interactions, and interaction dynamics [17].

loosely coordinated because interactions may occur at any one of a large number of meeting times. The meetings times themselves are chosen uniformly over time, and each node arrives at the meeting time perturbed by a zero-mean Gaussian random variable with a parameterized variance.

Performance

To demonstrate the effect of network composition on detectability, we use the previously mentioned R-MAT model to generate graphs of different sizes with different embedded foregrounds. We provide STTP with a single random observation into a foreground subgraph of 30 nodes. We create networks of 100, 1000, and 10,000 nodes and embed either a minimally connected Erdös-Rényi random covert network or a tree covert network. The results for the 1000-trial Monte Carlo experiment are shown in Figure 12. As predicted, detection becomes harder as the background grows larger compared to the foreground, and the more realistic tree network is harder to detect because of the smaller connectivity.

Summary

The problem of covert network detection is analyzed from the perspectives of graph partitioning and algebraic graph theory. Network detection is addressed as a special case of graph partitioning in which membership in a small subgraph of interest must be determined, and a common framework is developed to analyze and compare different network-detection methods. A new Bayesian network-detection framework called space-time threat propagation is introduced that partitions the graph on the basis of prior information and direct observations. Space-time threat propagation is shown to be optimum in the Neyman-Pearson sense, subject to the assumption that threat networks are connected by edges temporally correlated to

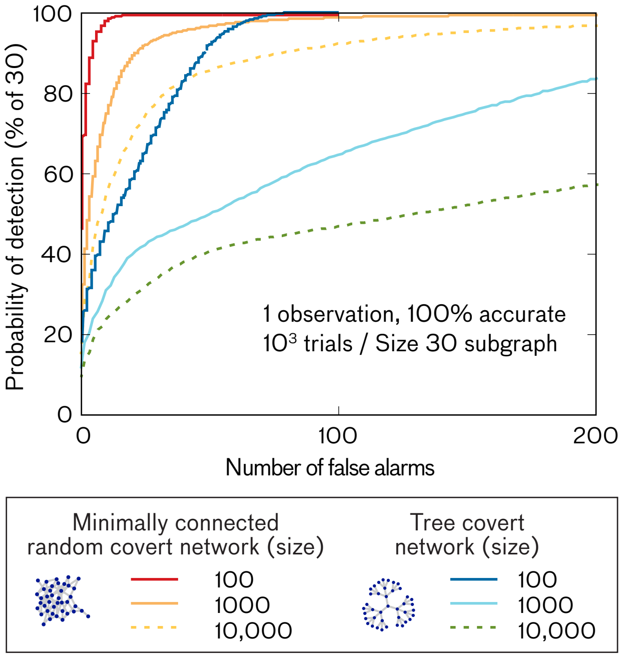


FIGURE 12. Receiver operating characteristics plot the probability of detections (PD) versus the number of false alarms (NFA). Better performance is measured with a higher PD at a given NFA. This plot illustrates the relative difficulty of detecting covert tree networks compared to minimally connected random networks, as predicted. The dependence on the background size is also shown and, also as expected, seen to be relatively weak—an order of magnitude increase in size only results in twice the number of false alarms.

a cue or observation. Bayesian space-time threat propagation is interpreted as the solution to a harmonic boundary value problem on the graph, in which a linear approximation to Bayes' rule determines the unknown probability of threat on the uncued nodes (the interior) on the basis of threat observations at cue nodes (the boundary). This new method is compared to well-known spectral methods by examining competing notions of network detection optimality. Finally, a new generative mixed-membership stochastic block model is introduced for performance-prediction network-detection algorithms. The parameterized model combines key real-world aspects of several random graph models: Erdös-Rényi for sparsity and connectivity, Chung-Lu for power-law degree distributions, and a mixed-membership stochastic block model for distinctive community-based interaction and dynamics. This model is used to compute empirical detection performance results for the detection algorithms described in the article as both foreground coordination and activity levels are varied. Though the results here are empirical, it is our hope that both the analytic results and performance modeling will be useful in future closed-form analysis of real-world covert network detection problems. ■

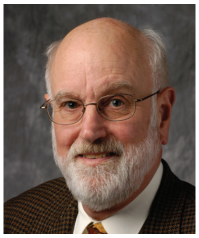
References

1. United States Army. Counterinsurgency: Field Manual 3-24, Appendix B. Washington: Government Printing Office, 2006.
2. K. Carley, "Estimating Vulnerabilities in Large Covert Networks," *Proceedings of the 16th International Command and Control Research and Technology Symposium.* (ICCRTS), 2004.
3. M. Sageman, *Understanding Terror Networks*. Philadelphia: University of Pennsylvania Press, 2004.
4. J. Jonas and J. Harper, "Effective Counterterrorism and the Limited Role of Predictive Data Mining," *Policy Analysis 584*, Cato Institute, 2006.
5. J. Xu and H. Chen, "The Topology of Dark Networks," *Communications of the ACM*, vol. 51, no. 10, 2008, pp. 58–65.
6. R. Trinquier, *Modern Warfare: A French View of Counterinsurgency*. Westport, Conn.: Praeger Security International, 2006.
7. V.E. Krebs, "Uncloaking Terrorist Networks," *First Monday*, vol. 7, no. 4, 2002.
8. Doctor Web, "Doctor Web Exposes 550,000 Strong Mac Botnet," 4 April 2012, http://news.drweb.com/show/?i=2341, accessed 3 September 2012.
9. D.J. Watts, "Networks, Dynamics, and the Small-World Phenomenon," *American Journal of Sociology*, vol. 13, no. 2, 1999, pp. 493–527.
10. J. W. Perry et al., *Protecting Individual Privacy in the Struggle Against Terrorists: A Framework for Program Assessment.* Washington, D.C., The National Academies Press, 2008.
11. G. Pontecorvo, dir., *The Battle of Algiers.* Rialto Pictures. 1966.
12. E.M. Airoldi, D.M. Blei, S.E. Fienberg, and E.P. Xing, "Mixed-Membership Stochastic Blockmodels," *Journal of Machine Learning Research*, vol. 9, 2008, pp. 1981–2014.
13. M. Jacomy and M. Roussel, "Mapping Science in Society Actors in Europe," Ecsite Annual Conference 2009, Milan, Italy, 2009.
14. J. Leskovec, K.J. Lang, and M. Mahoney, "Empirical Comparison of Algorithms for Network Community Detection," *Proceedings of the 19th International Conference on World Wide Web (WWW10)*, 2010, pp. 631–640.
15. H. Wolkowicz and Q. Zhao, "Semidefinite Programming Relaxations for the Graph Partitioning Problem," *Discrete Applied Mathematics*, vol. 96–97, 1999, pp. 461–479.
16. S. Philips, E.K. Kao, M. Yee, and C. Anderson, "Detecting Activity-Based Communities Using Dynamic Membership Propagation," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 2085–2088.

17. S.T. Smith, K.D. Senne, S. Philips, E.K. Kao, and G.B. Bernstein, “Network Detection Theory and Performance,” submitted to *IEEE Transactions on Signal Processing*, arXiv:1303.5613 [cs.SI], http://arxiv.org/abs/1303.5613i, accessed 1 May 2013.
18. S.T. Smith, S. Philips, and E.K. Kao, “Harmonic Space-Time Threat Propagation for Graph Detection,” *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 3933–3936.
19. Google, “The Technology Behind Google’s Great Results,” http://www.google.com/ onceuponatime/technology/ pigeonrank.html, accessed 3 September 2012.
20. A. Shamir, “A Survey on Mesh Segmentation Techniques,” *Computer Graphics Forum*, vol. 27, no. 6, 2008, pp. 1539–1556.
21. L. Yang, “Data Embedding Research,” Western Michigan University, http://www.cs.wmich.edu/yang/research/ dembed/, accessed 3 September 2012.
22. A. Brun, H. Knutsson, H.J. Park, M.E. Shenton, and C.-F. Westin, “Clustering Fiber Tracts Using Normalized Cuts,” *Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI 04)*, 2004, pp. 368–375.
23. TeleGeography. “Global Traffic Map 2010,” PriMetrica, http://www.telegeography.com/telecom-maps/ global-tracmap/index.html, accessed 3 September 2012.
24. K.G. Guruharsha et al., “A Protein Complex Network of *Drosophila melanogaster*,” *Cell*, vol. 147, no. 3, 2011, pp. 690–703.
25. R. Diestel, *Graph Theory.* New York: Springer-Verlag, 2000.
26. F.R.K. Chung, “Spectral Graph Theory,” *Regional Conference Series in Mathematics 92*, American Mathematical Society, 1994.
27. S.T. Smith, A. Silberfarb, S. Philips, E.K. Kao, and C. Anderson, “Network Discovery Using Wide-Area Surveillance Data,” *Proceedings of the 14th International Conference on Information Fusion*, 2011.
28. H.L. Van Trees, *Detection, Estimation, and Modulation Theory, Part 1.* New York: John Wiley and Sons, 1968.
29. C. Godsil and G. Royle, *Algebraic Graph Theory.* New York: Springer-Verlag, Inc., 2001.
30. J.M. Keynes, *A Treatise on Probability.* London: Macmillan, 1921.
31. W.E. Donath and A.J. Hoffman, “Lower Bounds for the Partitioning of Graphs,” *IBM Journal of Research and Development*, vol. 17, 1973, pp. 420–425.
32. M. Fiedler, “Algebraic Connectivity of Graphs,” *Czechoslovak Mathematical Journal*, vol. 23, no. 2, 1973, pp. 298–305.
33. M. Fiedler, “A Property of Eigenvectors of Non-negative Symmetric Matrices and Its Application to Graph Theory,” *Czechoslovak Mathematical Journal*, vol. 25, 1975, pp. 619–633.
34. A. Pothen, H. Simon, and K.-P. Liou, “Partitioning Sparse Matrices with Eigenvectors of Graphs,” *SIAM Journal on Matrix Analysis and Applications*, vol. 11, 1990, pp. 430–450.
35. M.E. J. Newman, “Finding Community Structure in Networks Using the Eigenvectors of Matrices,” *Physical Review E*, vol. 74, no. 3, 2006.
36. B.A. Miller, M.S. Beard, and N.T. Bliss, “Eigenspace Analysis for Threat Detection in Social Networks,” *Proceedings of the 14th International Conference on Information Fusion*, 2011.
37. B.A. Miller, N.T. Bliss, and P.J. Wolfe, “Toward Signal Processing Theory for Graphs and Other Non-Euclidean Data,” *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2010, pp. 5414–5417.
38. B.A. Miller, N.T. Bliss, and P.J. Wolfe, “Subgraph Detection Using Eigenvector L_1 Norms,” *Proceedings of the 2010 Conference on Neural Information Processing Systems*, 2010, pp. 1633–1641.
39. S. Fortunato and M. Barthélemy, “Resolution Limit in Community Detection,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 104, no. 1, 2007, pp. 36–41.
40. S. Fortunato, “Community Detection in Graphs,” *Physics Reports*, vol. 486, 2010, pp. 75–174.
41. J.M. Kumpula, J. Saramäki, K. Kaski, and J. Kertész, “Limited Resolution in Complex Network Community Detection with Potts Model Approach,” *The European Physical Journal B*, vol. 56, no. 1, 2007, pp. 41–45.
42. R.R. Nadakuditi and M.E.J. Newman, “Graph Spectra and the Detectability of Community Structure in Networks,” *Physical Review Letters*, vol. 108, 2012, 188701.
43. M.J. Yee, S. Philips, G.R. Condon, P.B. Jones, E.K. Kao, S.T. Smith, C.C. Anderson, and F.R. Waugh, “Network Discovery with Multi-intelligence Sources,” *Lincoln Laboratory Journal*, vol. 20, no. 1, 2013, pp. 31–46.
44. P. Erdös and A. Rényi, “On the Evolution of Random Graphs,” *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, vol. 5, 1960, pp. 17–61.
45. W. Aiello, F. Chung, and L. Lu, “A Random Graph Model for Power Law Graphs,” *Experimental Mathematics*, vol. 10, no. 1, 2001, pp. 53–66.

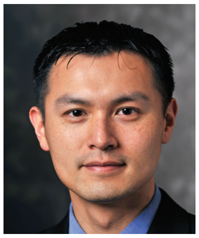
About the Authors

Steven T. Smith is a senior staff member of the Intelligence and Decision Technologies Group with many distinguished accomplishments in radar, sonar, and systems concept development. His contributions span all aspects of signal processing, from data modeling and measurement, to novel signal detection and estimation algorithms, to target tracking. Professional and career highlights involve novel solutions and analysis in new problems, such as geometric optimization for signal processing, statistical resolution limits, bounds for nonlinear parameter estimation, and optimum network detection. He is the recipient of outstanding paper awards from the IEEE and SIAM. He received his bachelor's degree in electrical engineering and mathematics from the University of British Columbia, Vancouver, and his doctoral degree in applied mathematics from Harvard University.



Kenneth D. Senne is a principal staff member in the Intelligence, Surveillance, and Reconnaissance and Tactical Systems Division. His research examines the application of large data analytics to decision support problems. He joined the Laboratory in 1972 to work on the design and collision avoidance application of the Mode-S beacon system for the Federal Aviation Administration. From 1977 to 1986, he contributed to the development of antijam airborne communication systems and superresolution direction finding with adaptive antennas. In 1986, he was asked to set up an array signal processing group as part of a large air defense airborne electronics program. This effort resulted in the pioneering demonstration of a large-scale, real-time embedded adaptive signal processor. In 1998, he was promoted to head of the Air Defense Technology Division. In 2002, he established the Laboratory's Technology Office, with responsibility for managing technology investments, including the internal innovative research program. Prior to joining Lincoln Laboratory, he earned a doctorate in electrical engineering from Stanford University with foundational research on digital adaptive signal processing. He also served as Captain in the U.S. Air Force with the Frank J. Seiler Research Laboratory at the Air Force Academy. He was elected Fellow of the IEEE in 2002 "for contributions to the development of the real-time adaptive signal processing systems for defense applications."

Scott Philips is currently a data scientist at Palantir Technologies. Before joining Palantir, Scott spent five years as a member of the technical staff in the Intelligence and Decision Technologies Group. While he was at Lincoln Laboratory, his research focused on developing statistical algorithms for the exploitation of data from intelligence, surveillance, and reconnaissance sensors. Scott received his doctoral degree in electrical engineering in 2007 from the University of Washington, where his research focused on signal processing and machine learning algorithms for the detection and classification of sonar signals.



Edward K. Kao is a Lincoln Scholar in the Intelligence and Decision Technologies Group. Since joining the Laboratory in 2008, he has been working on graph-based intelligence, in which actionable intelligence is inferred from interactions and relationships between entities. Applications include wide-area surveillance, threat network detection, homeland security, and cyber warfare. In 2011, he entered the doctoral program at Harvard University in the statistics department. Current research topics include causal inference on peer influence effects, statistical models for community membership estimation, information content in network inference, and optimal sampling and experimental design for network inference.

Garrett Bernstein is a member of the technical staff in the Intelligence and Decision Technologies Group. He previously interned in the Space Control Systems and Surveillance Systems Groups. His research focuses on statistical inference and machine learning applied to diverse problems, such as graph detection algorithms, model simulation, semantic analysis, and military operational effectiveness. Prior to joining the Laboratory, he received a bachelor's degree in applied and engineering physics and an engineering master's degree in computer science, both from Cornell University.