# Robust Defense Strategy for Network Security Against Unknown Attack Models

Armita KazemiNajafabadi* and Mahdi Imani†
*Northeastern University, Boston, MA, 02115*

**Network security plays an increasingly vital role across various domains, particularly in sensitive areas such as aerospace systems. Examples include computer networks controlling flight systems and computers securely transmitting classified data. Several security approaches have been developed in recent years. This paper models network security as a Bayesian attack graph (BAG), a powerful model to capture the penetration and propagation of attacks in the network. Most existing defense policies for BAGs are designed for networks with known vulnerabilities and threats, denoted by a known BAG. However, in practice, the network vulnerabilities or threats could be presented by a set of BAGs. As attackers become more intelligent and dynamic, they utilize their resources to execute new or hard-to-detect attacks, posing uncertainty in network models. Given the uncertainty in the threat model, developing a robust defense strategy to ensure network security is crucial. This paper formulates an optimal robust defense policy that maximizes expected accumulated security reward under worst-case conditions (i.e., against the most aggressive threat model). We provide proof of convergence for the proposed policy, demonstrating that the optimal policy is computable for any network with any type of vulnerability. Furthermore, we introduce an efficient matrix-based computation of the optimal policy through an offline process, which enables real-time implementation during system operation. Numerical experiments demonstrate the robustness and accuracy of the proposed policy under various conditions.**

## I. Introduction

Network security in aerospace systems plays a critical role in safeguarding against cyber threats and ensuring operational integrity. As technology advances, the complexity and interconnectedness of these systems make them vulnerable to various types of attacks. For instance, malicious actors could exploit vulnerabilities in flight control systems or intercept classified communications, potentially leading to disastrous consequences such as unauthorized access, data breaches, or even physical damage. Given the high stakes involved, effective network security measures are imperative [1–7]. To protect these systems against cyberattacks, various techniques have been developed [8–10]. In particular, firewalls [11, 12] serve as the initial layer of network defense by blocking unwanted traffic but can be bypassed by advanced attacks. Intrusion detection systems (IDS) [6, 13, 14] help detect threats by monitoring traffic, though attackers can exploit their weaknesses to trigger false alarms, overwhelming the system. Encryption algorithms [15] secure data in transit, but determined attackers with sufficient resources may eventually break them.

More advanced security systems go beyond the automatic security checks on a single device (e.g., IDS and firewall) and consider the system as a network of interconnected components or defended by artificial intelligence (AI) and or human-AI agents [16–22]. These approaches take into account the connections between elements and with outside sources to assess the vulnerabilities and derive security solutions [23]. Figure 1 provides a simple illustration of attacks on air traffic control systems. The connectivity of devices and servers, which are responsible for critical decision-making in air traffic control, makes them vulnerable to various types of attacks. These attacks target devices connected to external sources. Such breaches can infiltrate and spread through the network, significantly disrupting air traffic control operations.

Attack graphs are a powerful class of models that consider the dependencies between the components of a network for security analyses. The components are modeled as nodes of the graph and the dependencies between them as edges [24]. In particular, Bayesian attack graphs (BAGs) [25, 26] are a variation of the attack graphs which allow for uncertain representation of attack propagation in computer networks. BAGs are particularly useful in situations where the system

*PhD Candidate, Department of Electrical and Computer Engineering, Northeastern University
†Assistant Professor, Department of Electrical and Computer Engineering, Northeastern University

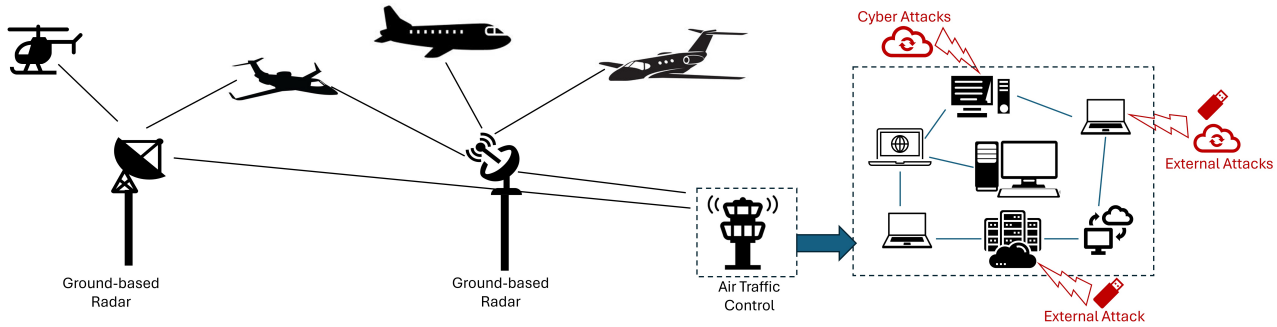



**Fig. 1 An illustration of potential vulnerabilities in air traffic control systems to cyber attacks, which can infiltrate the network and disrupt critical decision-making processes.**

being modeled is complex and has many interdependent components, allowing for a systematic and comprehensive analysis of network vulnerabilities and potential attack scenarios [27].

Several defense policies have been developed for the security of networks modeled by BAGs. These methods rely on full knowledge of the attack model, including network vulnerabilities and attackers' behavior [16–18]. However, in practice, the defender might have partial knowledge or uncertainty about the true attack model(s). This is due to the ever-growing connectivity and evolving attack landscapes, which allow attackers to develop new strategies to compromise networks. Particularly, the attackers can design sophisticated attacks, target new network components, or dynamically alter their behavior to maximize damage [28, 29]. Existing defense policies often struggle to adapt to such complexities, demanding the development of security solutions that yield robust performance given the uncertainty in the model. This is especially critical in sensitive aerospace systems, where intrusions can severely impact network security and overall system performance [30].

This paper models the uncertainty in network compromises using a set of Markov decision processes (MDPs). Each MDP represents network security given a specific threat model, capturing the stochastic nature of attack propagation during the defense process. These models reflect realistic scenarios where the defender has knowledge of the set of possible attack behaviors but does not know which model the attacker will follow. Given such uncertainty, the paper develops a robust defense policy that optimizes performance under the worst-case attack scenario. We provide the proof of convergence for the proposed policy for the general form of BAG with an arbitrary set of threat models, along with an efficient matrix-form implementation. This policy can be computed offline and deployed in real-time. The numerical experiments demonstrate the superiority of the proposed policy in terms of robustness and performance.

## II. Attack Penetrations and Propagation through Bayesian Attack Graph

The security of a network comprising $n$ components can be represented using the following graph:

$$\mathcal{G}^\theta = (\mathcal{N}, \mathcal{T}, \mathcal{E}^\theta, \mathcal{P}^\theta)$$

where $\mathcal{N} = \{1, \cdots, n\}$ represents $n$ components of the network and $\mathcal{T}$ denotes the type of components expressing their security levels. $\Theta$ represents the set of possible threat models, where $\theta \in \Theta$ indicates a specific attack model. $\mathcal{E}^\theta$ indicates the connection between the components through directed edges used by the attacker, and $\mathcal{P}^\theta$ is the set of exploit probabilities under the attack model $\theta$. The nodes are random variables taking in $\{0, 1\}$, where 0 indicates an uncompromised component and 1 indicates a compromised one. Each node belongs to one of two types, $\mathcal{T}_i \in \{\text{AND}, \text{OR}\}$, reflecting the vulnerability type of the $i$-th device, machine, or computer. An edge $(i, j) \in \mathcal{E}^\theta$ signifies that node $i$ can potentially be compromised via node $j$ in the attack model $\theta$. $\mathcal{P}^\theta$ contains the exploit probabilities for the edges, with $\rho_{ij}^\theta \in \mathcal{P}^\theta$ denoting the likelihood of node $j$ being compromised through node $i$ (assuming node $i$ is already compromised). These exploit probabilities are often derived from the NIST's Common Vulnerability Scoring System (CVSS) [31], which quantifies vulnerability severity using numerical scores.

The graph models the probabilistic spread of attacks, which can also be viewed as a Markov process with binary state variables. The state vector, $\mathbf{x}_k = [\mathbf{x}_k(1), ..., \mathbf{x}_k(n)]^T$, describes the compromise status of all $n$ nodes in the network at time step $k$, where $\mathbf{x}_k(i)$ is either 0 (uncompromised) or 1 (compromised). A fully uncompromised network is represented as $\mathbf{x}_k = [0, 0, ..., 0]^T$, while $\mathbf{x}_k = [1, 1, ..., 1]^T$ indicates all nodes are compromised. There are $2^n$ possible states for the vector, noted as $\mathbf{x}^1, ..., \mathbf{x}^{2^n}$. The propagation of attacks across the graph depends on several factors, such as

external attack probabilities, internal exploit probabilities between nodes, component types, and mitigation efforts. The set $\mathcal{N}_{ex}^{\theta}$ represents components connected to external sources exposed to direct attack in model $\theta$ with $\rho_j^{\theta}$ denoting the external exploit probability for arbitrary node $j \in \mathcal{N}_{ex}^{\theta}$. Each internal node ($j \notin \mathcal{N}_{ex}^{\theta}$) can be compromised through internal connections propagating attacks. For an arbitrary internal node $j$, the set $D_j^{\theta}$ consists of all its incoming edges that can be utilized to compromise the node. It can be formally defined as $D_j^{\theta} = \{i \in \mathcal{N} | (i, j) \in \mathcal{E}^{\theta}\}$. Internal components come in two forms: AND nodes are compromised only if all connected nodes are compromised, OR nodes can be compromised by any single connected compromised node.

A common method to mitigate network compromises involves running advanced firewalls on selected computers or servers vulnerable to attack. Although this approach is convenient and can be executed remotely, it may not always effectively eliminate compromises, particularly as some threats can bypass firewalls. To complement this strategy, applying software patches or updates to vulnerable systems is another method to address network security issues. However, while patching can close known vulnerabilities, it may not fully resolve the problem, especially when dealing with persistent threats. Another widely used technique is reimaging, which involves reinstalling operating systems and software on compromised machines or servers. Although this method is potentially costly and disruptive to network operations, it has a high success rate in removing compromises. However, if attackers have obtained critical credentials, such as domain passwords, reimaging may not fully secure the system. In this context, for simplicity, any method of removing compromises will be referred to as reimaging.

At each time step, the defender selects a subset of nodes to be defended. In line with the modeling approach used in our previous work [32–34], let $\mathbf{a}_{k-1} \subset \mathcal{N}$ denote the subset of nodes chosen for the reimaging process at time step $k$. The probability of successfully removing a compromise at any selected node is given by $(1 - \alpha)$, where $0 \leq \alpha \leq 1$ represents the probability of an unsuccessful removal. The value of $\alpha$ is influenced by the complexity of the reimaging process; a more comprehensive reimaging approach results in a smaller $\alpha$ value. For the attack model $\theta$, the conditional probability that the $j$th node is compromised at time step $k$, given the nodes' state at time step $k-1$, denoted as $\mathbf{x}_{k-1}$, and the reimaged nodes $\mathbf{a}_{k-1} = \{i_1, ..., i_r\} \subset \mathcal{N}$, can be expressed for AND and OR nodes as:

- *AND Nodes:*

$$P(\mathbf{x}_k(j) = 1 \mid \mathbf{x}_{k-1}, \mathbf{a}_{k-1}, \theta) = \begin{cases} (1_{j \notin a_{k-1}} + \alpha 1_{j \in a_{k-1}}) \Big[ \rho_j^{\theta} + (1 - \rho_j^{\theta}) \\ \qquad \prod_{i \in D_j^{\theta}} \rho_{ij}^{\theta} 1_{x_{k-1}(i)=1} \Big] & \text{if } \mathbf{x}_{k-1}(j) = 0, \\ 1_{j \notin a_{k-1}} + \alpha 1_{j \in a_{k-1}} & \text{if } \mathbf{x}_{k-1}(j) = 1, \end{cases}$$

- *OR Nodes:*

$$P(\mathbf{x}_k(j) = 1 \mid \mathbf{x}_{k-1}, \mathbf{a}_{k-1}, \theta) = \begin{cases} (1_{j \notin a_{k-1}} + \alpha 1_{j \in a_{k-1}}) \Big[ \rho_j^{\theta} + (1 - \rho_j^{\theta}) \\ \qquad \Big[ 1 - \prod_{i \in D_j^{\theta}} \Big( 1 - \rho_{ij}^{\theta} 1_{\mathbf{x}_{k-1}(i)=1} \Big) \Big] \Big] & \text{if } \mathbf{x}_{k-1}(j) = 0, \\ 1_{j \notin a_{k-1}} + \alpha 1_{j \in a_{k-1}} & \text{if } \mathbf{x}_{k-1}(j) = 1, \end{cases}$$

Note that $P(\mathbf{x}_k(j) = 0 \mid \mathbf{x}_{k-1}, \mathbf{a}_{k-1}, \theta) = 1 - P(\mathbf{x}_k(j) = 1 \mid \mathbf{x}_{k-1}, \mathbf{a}_{k-1}, \theta)$.

## III. Proposed Robust Defense Policy

To effectively secure complex networks, it is essential to develop a strategy that can perform robustly given the unknown information of the true threat model. In the context of network security, the objective is to determine and implement the optimal sequence of defensive actions that safeguard the network from potential attacks. The proposed defense policy must not only meet the network's security requirements but also effectively respond to potential breaches that could lead to severe disruptions or catastrophic consequences for the entire system. An optimal defense policy can be obtained to address the specific characteristics and vulnerabilities of a known attack model. In our previous work [35], we developed and evaluated optimal policies under both full knowledge of network states and scenarios with partial observability of network compromises. Those policies, however, consider a known and fully known BAG model, which limits their application to domains with known and single threat models. Given a finite set of threat models, this paper focuses on developing robust defense policies that can yield robust security performance despite the uncertainty

of the true threat model. This is crucial because real-world network security involves uncertainty in both the attacker's behavior and the network model. Our goal is to create a strategy that maintains proper security, even when the true attack model is unknown, ensuring better resilience compared to approaches that assume a single, fixed model.

Let $\Theta$ represent the space of threat models, where each $\theta \in \Theta$ corresponds to a specific BAG that describes potential attack propagation within the network. If $\theta \neq \theta'$, the models differ in terms of vulnerabilities and attack methods, meaning the defense strategies for these models could also vary. To simplify, we assume the uncertainty in network vulnerabilities is captured by a finite set: $\Theta = \{\theta^1, \ldots, \theta^M\}$. The true threat model is unknown, among one of these possibilities at any given time. Changes in the model over time reflect the evolving nature of adversaries or shifting attack patterns, which are often difficult to detect. In the next section, we first outline the optimal defense policy for a single threat model and then introduce a strategy designed to provide robust protection under model uncertainty.

**A. Proposed Robust Defense Policy: Markov Decision Process**

The attack propagation and penetration in the network with varying or unknown underlying model(s) can be expressed using unknown MDP, represented by a 6-tuple $\langle \Theta, \mathcal{X}, \mathcal{A}, \mathcal{P}^\theta, R, \gamma \rangle$, where $\Theta$ represents the possible threat models, $\mathcal{X} = \{0, 1\}^n$ is the state space, $\mathcal{A}$ is the defense action space, $\mathcal{P}^\theta : \mathcal{X} \times \mathcal{A} \times \mathcal{X}$, denoted in (II), (II), is the state transition probability function such that $P(\mathbf{x}' \mid \mathbf{x}, \mathbf{a}, \theta)$ represents the probability of moving to state $\mathbf{x}'$ after taking defense action $\mathbf{a}$ in state $\mathbf{x}$ under model $\theta$. $R : \mathcal{X} \times \mathcal{A} \times \mathcal{X} \to \mathbb{R}$ is a bounded reward function such that $R(\mathbf{x}, \mathbf{a}, \mathbf{x}')$ encodes the reward earned when defense action $\mathbf{a}$ is taken in state $\mathbf{x}$ and the system moves to state $\mathbf{x}'$, and $0 < \gamma < 1$ is a discount factor. This paper interchangeably uses defense, and reimaging as forms of security actions to defend the network. The security action space $\mathcal{A}$ includes all possible network components that can be picked for defense/reimaging. The defender aims to maximally enhance network security; thus, the reward function $R(\mathbf{x}, \mathbf{a}, \mathbf{x}')$ measures the improvement in the network security upon taking action $\mathbf{a}$, transitioning the system from $\mathbf{x}$ to $\mathbf{x}'$, while also accounting for the potential cost of the defense process.

Let $\pi : \mathcal{X} \to \mathcal{A}$ be a deterministic policy, mapping an action to any given state. The optimal defense policy for the network model $\theta \in \Theta$ can be expressed as:

$$\pi^*_\theta(\mathbf{x}) = \max_{\pi \in \Pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(\mathbf{x}_t, \mathbf{a}_t, \mathbf{x}_{t+1}) \mid \mathbf{x}_0 = \mathbf{x}, \mathbf{a}_{0:\infty} \sim \pi, \theta\right], \tag{1}$$

for all $\mathbf{x} \in \mathcal{X}$; where the expectation is with respect to attack propagation and penetration under model $\theta$, and $\Pi$ is the space of all deterministic policies. Finding the defense policy using dynamic programming or reinforcement learning approaches leads to solutions that ensure achieving the highest average accumulated rewards under model $\theta$.

Despite the optimality of the policy in (1) for model $\theta$, uncertainty in the true threat model makes the use of a single policy unreliable. Additionally, the threat model may evolve over time. As a result, a policy optimized for one model may not perform well for another, and relying on a single model can leave the system, particularly its sensitive components, vulnerable to significant security risks. This paper derives an optimal robust defense policy that maintains proper security performance under model uncertainty. Let $\mu : \mathcal{X} \to \mathcal{A}$ be a robust policy, mapping the state to action space. The optimal robust policy can be formulated through the following optimization problem:

$$\mu^*(\mathbf{x}) = \max_{\mu \in \Pi} \min_{\theta \in \Theta} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(\mathbf{x}_t, \mathbf{a}_t, \mathbf{x}_{t+1}) \mid \mathbf{x}_0 = \mathbf{x}, \mathbf{a}_{0:\infty} \sim \mu, \theta\right], \tag{2}$$

where the minimization is with respect to the threat models, and the maximization is over the policy space. The minimization ensures that we consider the worst-case probable condition among models $\Theta$ to select a policy for those specific scenarios. The consideration of the minimum in (2) ensures the robustness of the policy, even if the worst-case security scenario is encountered in practice.

We define the *robust state value function* corresponding to a given policy $\mu$ as:

$$V^\mu(\mathbf{x}) = \min_{\theta \in \Theta} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(\mathbf{x}_t, \mathbf{a}_t, \mathbf{x}_{t+1}) \mid \mathbf{x}_0 = \mathbf{x}, \mathbf{a}_{0:\infty} \sim \mu, \theta\right], \tag{3}$$

for all $\mathbf{x} \in \mathcal{X}$. For the optimal robust policy $\mu^*$, we define the corresponding state-value function as $V^{\mu^*} := V^*$. Similar to the Bellman operator for average accumulated rewards [36], the robust state-value function for the optimal robust

policy $\mu^*$ holds:

$$\begin{aligned} V^*(\mathbf{x}) &= \max_{\mathbf{a}\in\mathcal{A}} \min_{\theta\in\Theta} \mathbb{E}_{\mathbf{x}'|\mathbf{x},\mathbf{a},\theta} \left[R(\mathbf{x},\mathbf{a},\mathbf{x}') + \gamma V^*(\mathbf{x}')\right] \\ &= \max_{\mathbf{a}\in\mathcal{A}} \min_{\theta\in\Theta} \sum_{\mathbf{x}'\in\mathcal{X}} P(\mathbf{x}' \mid \mathbf{x},\mathbf{a},\theta) \left[R(\mathbf{x},\mathbf{a},\mathbf{x}') + \gamma V^*(\mathbf{x}')\right], \end{aligned} \tag{4}$$

for all $\mathbf{x} \in \mathcal{X}$; where the maximum is over the action space.

The optimal robust policy can also be expressed using the optimal robust state value function $V^*$ as:

$$\mu^*(\mathbf{x}) = \operatorname*{argmax}_{\mathbf{a}\in\mathcal{A}} \min_{\theta\in\Theta} \sum_{\mathbf{x}'\in\mathcal{X}} P(\mathbf{x}'|\mathbf{x},\mathbf{a},\theta) \left[R(\mathbf{x},\mathbf{a},\mathbf{x}') + \gamma V^*(\mathbf{x}')\right], \tag{5}$$

for all $\mathbf{x} \in \mathcal{X}$. Note that any $V^*$ holding equality expression in (4) for all $\mathbf{x} \in \mathcal{X}$, corresponds to an optimal robust policy in (5). In the following sections, the proof of the existence of such a robust policy and a detailed matrix-form computation of it are provided.

### B. Proposed Robust Defense Policy: Notation

In this section, we extend the definitions introduced earlier to a vector and matrix format. Consider the elements of $\mathcal{X}$ arranged in an arbitrary fixed order and labeled as $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{2^n}$. This set includes all possible distinct states of the network, where the $i$th state is represented by $\mathbf{x}^i$. The transition matrix under the defense action $\mathbf{a} \in \mathcal{A}$ and the threat model $\theta$ is defined as:

$$\left(M^\theta(\mathbf{a})\right)_{ij} = P(\mathbf{x}^j \mid \mathbf{x}^i, \mathbf{a}, \theta), \text{ for } i, j = 1, \ldots, 2^n, \tag{6}$$

where $\left(M^\theta(\mathbf{a})\right)_{ij}$ indicates the element in the $i$the row and $j$th column of the transition matrix. The transition probabilities are determined by the parameters of the BAG model $\theta$ and can be incorporated into the matrix using expressions (II) and (II).

We define the vector-from representation of the expected reward function as:

$$\left(\mathbf{R}_\mathbf{a}^\theta\right)_i := R^\theta(\mathbf{x}^i, \mathbf{a}) = \sum_{j=1}^{2^n} P(\mathbf{x}^j \mid \mathbf{x}^i, \mathbf{a}, \theta) R\left(\mathbf{x}^i, \mathbf{a}, \mathbf{x}^j\right), \text{ for } i = 1, \ldots, 2^n, \tag{7}$$

where the expectation over the next state is taken with respect to transition probabilities to indicate the expected reward before observing the next state. Finally, we define the vector-form expression for the robust state-value function under the policy $\mu$:

$$\mathbf{V}^\mu = [V^\mu(\mathbf{x}^1), V^\mu(\mathbf{x}^2), \ldots, V^\mu(\mathbf{x}^{2^n})]^T \tag{8}$$

which is a vector of size $2^n$ with the $i$th element representing $V^\mu(\mathbf{x}^i)$. We represent the robust state value vector under the optimal robust policy $\mu^*$ as $\mathbf{V}^*$, which is equivalent to $\mathbf{V}^{\mu^*}$.

### C. Proposed Robust Defense Policy: Mathematical Foundation

Let $\mathcal{V} = [\mathcal{V}(1), ..., \mathcal{V}(2^n)]^T \in \mathbb{R}^{2^n}$ be a real-valued vector. We define the robust operator over any given $\mathcal{V}$ as:

$$\mathcal{T}^{\text{Ro}}[\mathcal{V}] = \max_{\mathbf{a}\in\mathcal{A}} \min_{\theta\in\Theta} \left[\mathbf{R}_\mathbf{a}^\theta + \gamma M^\theta(\mathbf{a})\mathcal{V}\right], \tag{9}$$

where the "max" and "min" operators are applied element-wise, $\mathbf{R}_\mathbf{a}^\theta$ is the expected reward introduced in (7) and $M^\theta(\mathbf{a})$ is the transition matrix defined in (6).

**Theorem III.1** *Iteratively applying the robust operator in (9) on a given $\mathcal{V} \in \mathbb{R}^{2^n}$, leads to the optimal robust state value vector (i.e. $\mathbf{V}^*$), which is a fixed-point solution of $\mathcal{T}^{Ro}$ (i.e. $\mathbf{V}^* = \mathcal{T}^{Ro}[\mathbf{V}^*]$).*

To establish the proof, we first state the following Banach fixed-point theorem [37], also known as contraction mapping theorem, which is later used for the argument.

**Theorem III.2** ***Banach Fixed-Point (Contraction Mapping) Theorem:*** *Let $(X, d)$ be a complete metric space, and let $\Phi : X \to X$ be a contraction mapping on X, i.e., there exists a constant $0 \le \kappa < 1$ such that for all $x, y \in X$, we have $d(\Phi(x), \Phi(y)) \le \kappa \cdot d(x, y)$. Then, $\Phi$ has a unique fixed-point $x^* \in X$.*

***Validating the Robust Operator as a Contraction Mapping***: Let $\mathbb{R}^{2^n}$ be the space of all real-valued vectors of dimension $2^n$. Consider the distance metric $d : \mathbb{R}^{2^n} \times \mathbb{R}^{2^n} \to \mathbb{R}$ as the maximum absolute difference between the corresponding components of two vectors $\mathcal{V}_1, \mathcal{V}_2 \in \mathbb{R}^{2^n}$ (i.e. $L_\infty$-norm) as:

$$d(\mathcal{V}_1, \mathcal{V}_2) = \max_{i \in \{1,\dots,2^n\}} |\mathcal{V}_1(i) - \mathcal{V}_2(i)|. \tag{10}$$

It is well-known that $(\mathbb{R}^{2^n}, d)$, is a complete metric space. To show that $\mathcal{T}^{\text{Ro}}$ in (9) is a contraction mapping, let $\mathcal{V}_1, \mathcal{V}_2$ be arbitrary vectors in $\mathbb{R}^{2^n}$. We want to show that:

$$d\left(\mathcal{T}^{\text{Ro}}[\mathcal{V}_1], \mathcal{T}^{\text{Ro}}[\mathcal{V}_2]\right) \leq \kappa \cdot d(\mathcal{V}_1, \mathcal{V}_2) \tag{11}$$

for some $0 \leq \kappa < 1$.

Let $\mathcal{W}_1 = \mathcal{T}^{\text{Ro}}[\mathcal{V}_1]$ and $\mathcal{W}_2 = \mathcal{T}^{\text{Ro}}[\mathcal{V}_2]$. Then:

$$\begin{aligned} d(\mathcal{W}_1, \mathcal{W}_2) &= \max_{i=1}^{2^n} |\mathcal{W}_1(i) - \mathcal{W}_2(i)| \\ &= \max_{i=1}^{2^n} \left| \max_{\mathbf{a} \in \mathcal{A}} \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right] - \max_{\mathbf{a} \in \mathcal{A}} \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right] \right|, \end{aligned} \tag{12}$$

where $\mathbf{R}_{\mathbf{a}}^{\theta}(i)$ is another representation of $\left(\mathbf{R}_{\mathbf{a}}^{\theta}\right)_i$.

The following holds for an arbitrary choice of functions; however, for specificity, consider the functions $f_1, f_2, f$, defined over $\mathcal{A}$ as:

$$\begin{aligned} f_1(\mathbf{a}) &= \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right], \text{ for } \mathbf{a} \in \mathcal{A}, \\ f_2(\mathbf{a}) &= \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right], \text{ for } \mathbf{a} \in \mathcal{A}, \\ f(\mathbf{a}) &= f_1(\mathbf{a}) - f_2(\mathbf{a}), \text{ for } \mathbf{a} \in \mathcal{A}. \end{aligned} \tag{13}$$

The following inequalities derived from the properties of the maximum operator, hold for the general functions with the same domain $f_1, f_2, f = (f_1 - f_2)$ where the maximum operator is defined over their domains (in this case $\mathcal{A}$):

$$\begin{aligned} &\max_{\mathbf{a} \in \mathcal{A}} \{f(\mathbf{a}) + f_2(\mathbf{a})\} \leq \max_{\mathbf{a} \in \mathcal{A}} \{f(\mathbf{a})\} + \max_{\mathbf{a} \in \mathcal{A}} \{f_2(\mathbf{a})\} \quad \Rightarrow \quad \max_{\mathbf{a} \in \mathcal{A}} \{f(\mathbf{a}) + f_2(\mathbf{a})\} - \max_{\mathbf{a} \in \mathcal{A}} \{f_2(\mathbf{a})\} \leq \max_{\mathbf{a} \in \mathcal{A}} \{f(\mathbf{a})\} \\ &\underset{f = f_1 - f_2}{\Rightarrow} \quad \max_{\mathbf{a} \in \mathcal{A}} \{f_1(\mathbf{a})\} - \max_{\mathbf{a} \in \mathcal{A}} \{f_2(\mathbf{a})\} \leq \max_{\mathbf{a} \in \mathcal{A}} \{f_1(\mathbf{a}) - f_2(\mathbf{a})\} \leq \max_{\mathbf{a} \in \mathcal{A}} |\{f_1(\mathbf{a}) - f_2(\mathbf{a})\}|. \end{aligned} \tag{14}$$

Without loss of generality, $f_1$ and $f_2$ can be swapped, redefining $f = (f_2 - f_1)$ and updating the inequalities accordingly:

$$\underset{\text{similar to (14)}}{\Rightarrow} \max_{\mathbf{a} \in \mathcal{A}} \{f_2(\mathbf{a})\} - \max_{\mathbf{a} \in \mathcal{A}} \{f_1(\mathbf{a})\} \leq \max_{\mathbf{a} \in \mathcal{A}} \{f_2(\mathbf{a}) - f_1(\mathbf{a})\} \leq \max_{\mathbf{a} \in \mathcal{A}} |\{f_2(\mathbf{a}) - f_1(\mathbf{a})\}|. \tag{15}$$

Combining the results of (14) and (15) one can prove that:

$$|\max_{\mathbf{a} \in \mathcal{A}} \{f_2(\mathbf{a})\} - \max_{\mathbf{a} \in \mathcal{A}} \{f_1(\mathbf{a})\}| \leq \max_{\mathbf{a} \in \mathcal{A}} |\{f_1(\mathbf{a}) - f_2(\mathbf{a})\}|. \tag{16}$$

Substituting the functions $f_1, f_2$ from (13) into the last expression (i.e. 16), we can conclude that:

$$\begin{aligned} &\left| \max_{\mathbf{a} \in \mathcal{A}} \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right] - \max_{\mathbf{a} \in \mathcal{A}} \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right] \right| \\ &\leq \max_{\mathbf{a} \in \mathcal{A}} \left| \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right] - \min_{\theta \in \Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right] \right|. \end{aligned} \tag{17}$$

Given an arbitrary $\mathbf{a} \in \mathcal{A}$, we now define the functions $g_1, g_2, g$ over $\Theta$ as:

$$\begin{aligned} g_1(\theta) &= \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right], \text{ for } \theta \in \Theta, \\ g_2(\theta) &= \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right], \text{ for } \theta \in \Theta, \\ g(\theta) &= g_2(\theta) - g_1(\theta), \text{ for } \theta \in \Theta. \end{aligned} \tag{18}$$

The following inequalities hold for arbitrary functions defined over the same domain. Here, we focus on inequalities arising from the properties of the minimum operator, that are applied generally to arbitrary functions $g_1$, $g_2$, and $g = g_2 - g_1$, with the minimum taken over their shared domain (in this case $\Theta$):

$$\begin{aligned} &\min_{\theta\in\Theta}\{g(\theta)\} + \min_{\theta\in\Theta}\{g_1(\theta)\} \leq \min_{\theta\in\Theta}\{g(\theta) + g_1(\theta)\} \Rightarrow \min_{\theta\in\Theta}\{g_1(\theta)\} - \min_{\theta\in\Theta}\{g(\theta) + g_1(\theta)\} \leq -\min_{\theta\in\Theta}\{g(\theta)\} \\ &\underset{g=(g_2-g_1)}{\Rightarrow} \min_{\theta\in\Theta}\{g_1(\theta)\} - \min_{\theta\in\Theta}\{g_2(\theta)\} \leq -\min_{\theta\in\Theta}\{g_2(\theta) - g_1(\theta)\} = \max_{\theta\in\Theta}\{g_1(\theta) - g_2(\theta)\} \\ &\Rightarrow \quad \min_{\theta\in\Theta}\{g_1(\theta)\} - \min_{\theta\in\Theta}\{g_2(\theta)\} \leq \max_{\theta\in\Theta}\{g_1(\theta) - g_2(\theta)\} \leq \max_{\theta\in\Theta} |\{g_1(\theta) - g_2(\theta)\}|. \end{aligned} \tag{19}$$

In the second line, the minimum operator is replaced with the negative maximum operator, while the third line applies the maximum operator alongside the properties of the absolute value. Without loss of generality, $g_1$ and $g_2$ can be swapped, redefining $g = (g_1 - g_2)$ and updating the inequalities in (19) accordingly:

$$\underset{\text{similar to (19)}}{\Rightarrow} \min_{\theta\in\Theta}\{g_2(\theta)\} - \min_{\theta\in\Theta}\{g_1(\theta)\} \leq \max_{\theta\in\Theta}\{g_2(\theta) - g_1(\theta)\} \leq \max_{\theta\in\Theta} |\{g_2(\theta) - g_1(\theta)\}|. \tag{20}$$

By combining the results of (19) and (20), it can be shown that:

$$|\min_{\theta\in\Theta}\{g_1(\theta)\} - \min_{\theta\in\Theta}\{g_2(\theta)\}| \leq \max_{\theta\in\Theta} |\{g_1(\theta) - g_2(\theta)\}|. \tag{21}$$

Substituting the functions $g_1, g_2$ with the given $\mathbf{a} \in \mathcal{A}$ from (18) into the last expression (21), the following result holds:

$$\begin{aligned} &\left| \min_{\theta\in\Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right] - \min_{\theta\in\Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right] \right| \\ &\leq \max_{\theta\in\Theta} \left| \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right] - \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right] \right| \end{aligned} \tag{22}$$

By combining the inequalities in (17), (22) with the notation introduced in (12), we obtain:

$$\begin{aligned} d(\mathcal{W}_1, \mathcal{W}_2) &= \max_{i=1}^{2^n} \left| \max_{\mathbf{a}\in\mathcal{A}} \min_{\theta\in\Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) \right] - \max_{\mathbf{a}\in\mathcal{A}} \min_{\theta\in\Theta} \left[ \mathbf{R}_{\mathbf{a}}^{\theta}(i) + \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right] \right| \\ &\leq \max_{i=1}^{2^n} \left( \max_{\mathbf{a}\in\mathcal{A}} \max_{\theta\in\Theta} \left| \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_1(j) - \gamma \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \mathcal{V}_2(j) \right| \right) \\ &= \max_{i=1}^{2^n} \left( \gamma \max_{\mathbf{a}\in\mathcal{A}} \max_{\theta\in\Theta} \left\{ \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} |\mathcal{V}_1(j) - \mathcal{V}_2(j)| \right\} \right) \leq \max_{i=1}^{2^n} \left( \gamma \max_{\mathbf{a}\in\mathcal{A}} \max_{\theta\in\Theta} \left\{ \sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} \max_{j=1}^{2^n} |\mathcal{V}_1(j) - \mathcal{V}_2(j)| \right\} \right) \\ &= \max_{i=1}^{2^n} \left( \gamma \max_{\mathbf{a}\in\mathcal{A}} \max_{\theta\in\Theta} \max_{j=1}^{2^n} |\mathcal{V}_1(j) - \mathcal{V}_2(j)| \right) \leq \gamma \max_{i=1}^{2^n} |\mathcal{V}_1(i) - \mathcal{V}_2(i)|, \end{aligned} \tag{23}$$

where the last line holds because the rows of the transition matrix consist of non-negative values that sum to one for each row $i$ (i.e., $\sum_{j=1}^{2^n} \left( M^{\theta}(\mathbf{a}) \right)_{ij} = 1$). For $\kappa = \gamma$, the inequality in (23) shows that $d(\mathcal{T}^{\text{Ro}}(\mathcal{V}_1), \mathcal{T}^{\text{Ro}}(\mathcal{V}_2)) \leq \kappa \cdot d(\mathcal{V}_1, \mathcal{V}_2)$.

This establishes that the robust operator $\mathcal{T}^{\text{Ro}}$ is a contraction mapping. According to Theorem $III.2$, there exists a unique fixed point $\mathcal{V}^*$ for the robust operator $\mathcal{T}^{\text{Ro}}$ in the space $\mathbb{R}^{2^n}$. This fixed point $\mathcal{V}^*$ is the optimal robust state value vector $\mathbf{V}^*$. We now show that starting from any initial vector $\mathcal{V}_0 \in \mathbb{R}^{2^n}$, the sequence $\mathcal{V}_{k+1} = \mathcal{T}^{\text{Ro}}[\mathcal{V}_k]$, generated by iteratively applying the robust operator, converges to $\mathbf{V}^*$. Since $\mathcal{T}^{\text{Ro}}$ is a contraction mapping with constant $\kappa \in [0, 1)$, for any two consecutive iterates $\mathcal{V}_{k+1}$ and $\mathcal{V}_k$, we have:

$$\max_{i\in\{1,\dots,2^n\}} |\mathcal{V}_{k+1}(i) - \mathbf{V}^*(i)| = \max_{i\in\{1,\dots,2^n\}} |\mathcal{T}^{\text{Ro}}[\mathcal{V}_k](i) - \mathcal{T}^{\text{Ro}}[\mathbf{V}^*](i)| \leq \kappa \max_{i\in\{1,\dots,2^n\}} |\mathcal{V}_k(i) - \mathbf{V}^*(i)|$$

By applying this inequality recursively, we obtain:

$$\max_{i\in\{1,\dots,2^n\}} |\mathcal{V}_{k+1}(i) - \mathbf{V}^*(i)| \leq \kappa^{k+1} \max_{i\in\{1,\dots,2^n\}} |\mathcal{V}_0(i) - \mathbf{V}^*(i)|$$

Since $\kappa \in [0, 1)$, as $k \to \infty$, $\kappa^{k+1} \to 0$. This implies that the sequence $\mathcal{V}_k$ converges to $\mathbf{V}^*$:

$$\lim_{k\to\infty} \max_{i\in\{1,\dots,2^n\}} |\mathcal{V}_k(i) - \mathbf{V}^*(i)| = 0$$

Thus, the iteratively computed values converge to the optimal robust state value vector $\mathbf{V}^*$, thereby proving Theorem III.1.

### D. Proposed Robust Defense Policy: Implementation

For a specific network, let $\Theta$ denote the set of threat models. At each step, an attack occurs using a fixed or varying threat model(s) that is unknown to the defender. Following the attack, the defender takes defensive actions based on the system states and compromises, aiming to maintain the long-term security of the system. The defense policy specifies which parts of the network require reimaging.

The proposed robust policy consists of both offline and real-time components. The offline component involves computing the proposed robust policy for the network, which can be achieved using the iterative process established in Theorem III.1. We can start with an initial vector $\mathcal{V}_0 \in \mathbb{R}^{2^n}$ (e.g., $\mathcal{V}_0 = [0, \cdots, 0]^T$) and recursively apply the robust operator $\mathcal{T}^{\text{Ro}}$. This process is defined as $\mathcal{V}_{k+1} = \mathcal{T}^{\text{Ro}}[\mathcal{V}_k]$, continuing until the maximum difference between the value vectors in two consecutive iterations falls below a small pre-specified threshold: $\max_i^{2^n} |\mathcal{V}_{k+1}(i) - \mathcal{V}_k(i)| < \epsilon$. The optimal robust policy $\mu^*$ can then be derived from the state value function $\mathcal{V}_{k+1}$, which is sufficiently close to the optimal robust state value function.

The time complexity of the value iteration method (offline step) is of order $O(|\mathcal{A}| \times |\Theta| \times 2^{2n} \times L)$, where $2^{2n}$ arises from the transition matrices involved in (9), and $L$ represents the number of iterations required before termination. In contrast, real-time action selection occurs in $O(1)$ time, as it involves a single lookup in the optimal robust policy vector to determine the action corresponding to the current state of the network.

## IV. Numerical Experiments

This section presents numerical experiments to evaluate the performance of the proposed defense policy. All results are averaged over 100 independent runs. The experiments are conducted on a network comprising 10 components, under various threat models. In the first set of experiments, the threat models differ based on the external attacks applied to the network. The performance of the proposed defense policy is compared against methods that assume a single threat model or employ random action selection. In the second set of experiments, the threat models involve similar external attacks but vary in their internal attack paths and corresponding propagation probabilities. Evaluations across both scenarios demonstrate the robustness of the proposed method in maintaining network security under diverse attack conditions.

**First Set of Threat Models: Various External Attacks**

The first set of experiments considers three different threat models, as illustrated in Figure 2. The network may be subjected to one of three types of external attacks, which are unknown to the defender in advance. These correspond to three distinct models: Model 1 ($\theta^1$), Model 2 ($\theta^2$), and Model 3 ($\theta^3$). Consequently, the space of threat models is defined as $\Theta = \{\theta^1, \theta^2, \theta^3\}$. The security reward is designed to reduce the number of compromised components. Thus, the negative of the number of network compromises in the next time step is considered as the reward, defined as: $R(\mathbf{x}, \mathbf{a}, \mathbf{x}') = -\sum_{j=1}^{n} 1_{\mathbf{x}'(j)=1}$. The reward ranges from 0 to $-10$, where 0 indicates no component is compromised, and $-10$ indicates that all nodes are compromised.

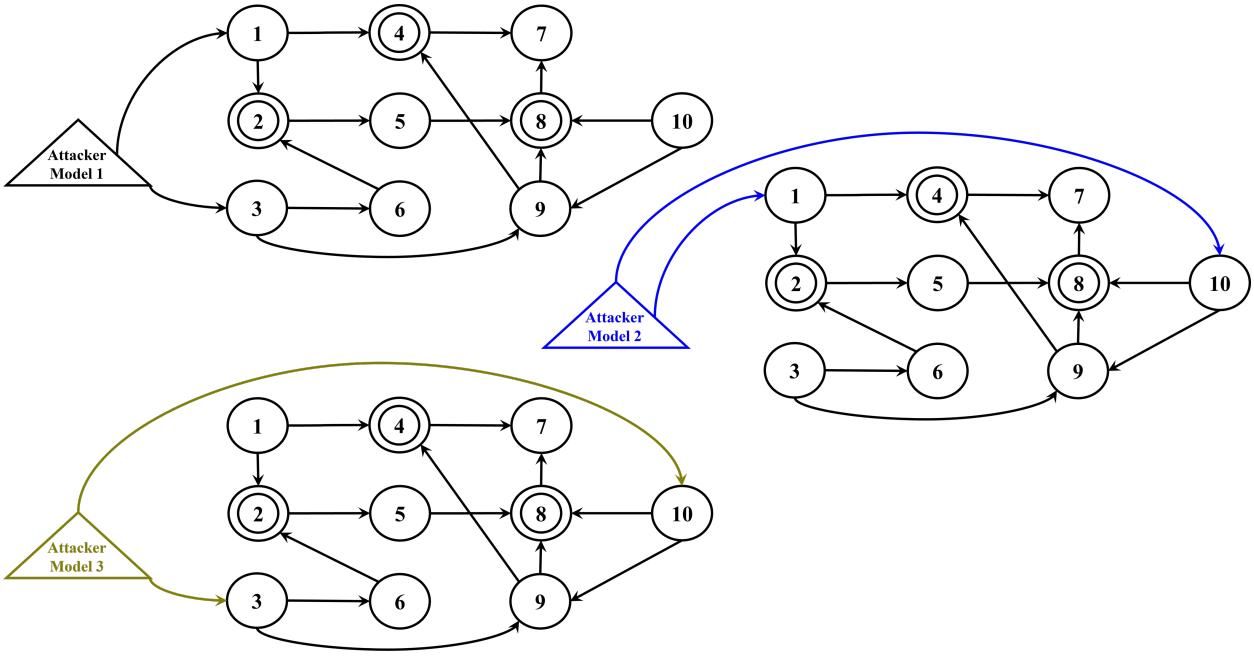
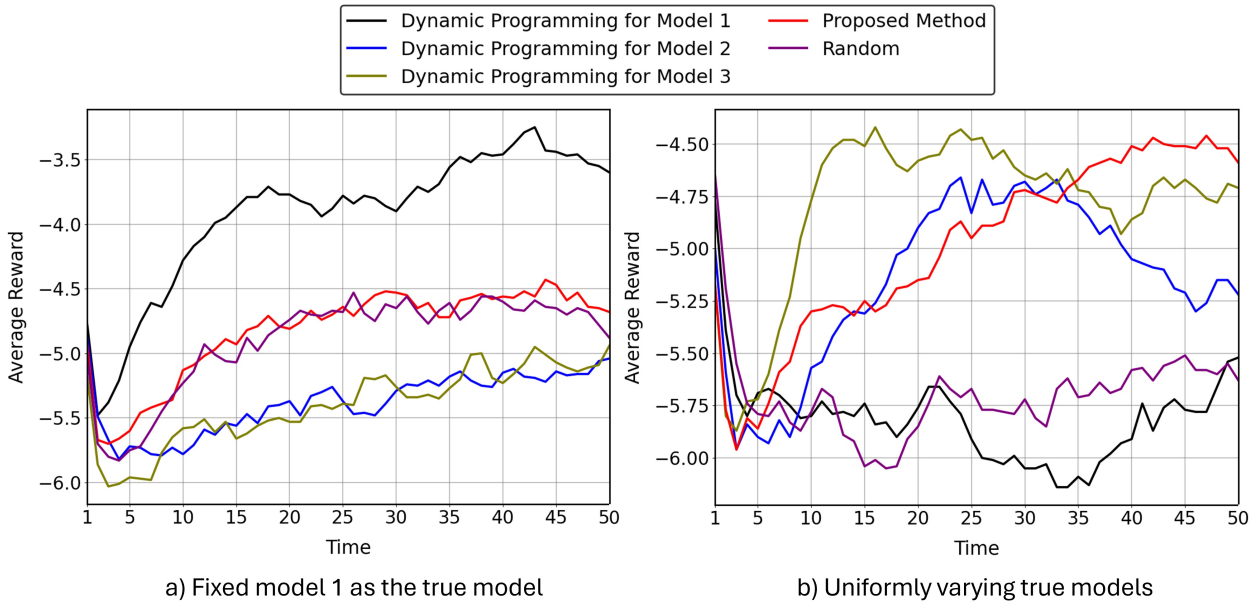



**Fig. 2 Network represented by 10-node BAG. Three possible attacker models are presented, leading to three possible network models $\Theta = \{\theta^1, \theta^2, \theta^3\}$.**

The network vulnerabilities across the models are distinguished by their superscript indices. In this set of experiments, for any $i, j \in \{1, \ldots, n\}$, the internal network vulnerabilities are identical across the three models. In other words, $\rho_{ij}^{\theta^1} = \rho_{ij}^{\theta^2} = \rho_{ij}^{\theta^3}$, so we use $\rho_{ij}$ to represent these common values: $\rho_{12} = 0.7, \rho_{14} = 0.6, \rho_{25} = 0.6, \rho_{36} = 0.55, \rho_{39} = 0.7, \rho_{47} = 0.7, \rho_{58} = 0.7, \rho_{62} = 0.7, \rho_{87} = 0.7, \rho_{94} = 0.6, \rho_{98} = 0.7, \rho_{108} = 0.7, \rho_{109} = 0.4$. The external exploit probabilities for each attack type are as follows: for Attack Type 1, $\rho_1^{\theta^1} = 0.65$ and $\rho_3^{\theta^1} = 0.6$; for Attack Type 2, $\rho_1^{\theta^2} = 0.65$ and $\rho_{10}^{\theta^2} = 0.55$; and for Attack Type 3, $\rho_3^{\theta^3} = 0.6$ and $\rho_{10}^{\theta^3} = 0.55$.




a) Fixed model 1 as the true model

b) Uniformly varying true models

**Fig. 3 Average security reward under different defense policies for BAGs with: a) the true network represented by model 1, b) the true network model is uniformly selected at each time among models 1 to 3.**

In this paper, we compare the performance of the proposed policy with dynamic programming policies for all three possible network models. Clearly, these models are not known in practice, so if we aim to select the policy corresponding to one of these three models, there is always a possibility that the policy for the non-true model is being implemented. By contrast, the proposed policy does not require any knowledge about the true model of the network. Moreover, the proposed policy can be utilized in cases where the network model might vary over time, reflecting shifts in vulnerabilities or attacker behavior. This is particularly common in adversarial conditions, where attackers aim to dynamically target nodes to maximize damage to the network.

Figure 3(a) expresses a case where model 1 is the true threat model. Since this is unknown to the defender, we perform the optimal dynamic programming policies for all models in defending model 1. Fig. 3(a) shows that the DP method for the true model (i.e., model 1) yields the highest average reward, i.e., the best security performance. This is due to the fact that this policy is derived to yield the best average performance for this threat model. However, the DP policies for model 2 and model 3 perform significantly poorly once applied to model 1 since these policies are not derived to maximize security performance for model 1. Therefore, given no knowledge of the true model, there is a huge risk of choosing among the security policies in practice. By contrast, the proposed robust defense policy has performed better than the DP policies for models 2 and 3. As expected, the robust defense policy performs worse than the DP policy for model 1, as actions are taken with respect to the worse case performance of the network. Therefore, the robustness of the proposed policy comes from the fact that no matter what the true network model is, the proposed policy prevents extreme damage to the network security, which might be possible once using the DP policies for an untrue network model. Finally, one can see that the random policy outperforms the DP policies for the wrong model, which again demonstrates the risk of performing the wrong policy in network security.

Figure 3(b) demonstrates a case with complex and time-varying behavior of the attackers, where the true network model is randomly selected at each time step. This expresses the case where the attacker might switch its behavior at any given time. Since no single model represents the network model at all times, we can see that the DP policies corresponding to each of the three models perform poorly. In particular, the DP for model 1 performs similarly to a random policy, and the other two DP policies also fluctuate and show unstable performance. Again, it should be emphasized that given the lack of knowledge about the true model, any of the three performances shown in Fig. 3 is possible to achieve if the defender aims to pick one of the three DP policies. This could cause a significant risk in network security (e.g. if the DP for model 1 is performed). However, one can see a more stable performance of the proposed robust defense policy, indicated by the red curve. This policy holds relatively good security results, and it becomes better than another method after 30 steps while not requiring any knowledge of the network model.

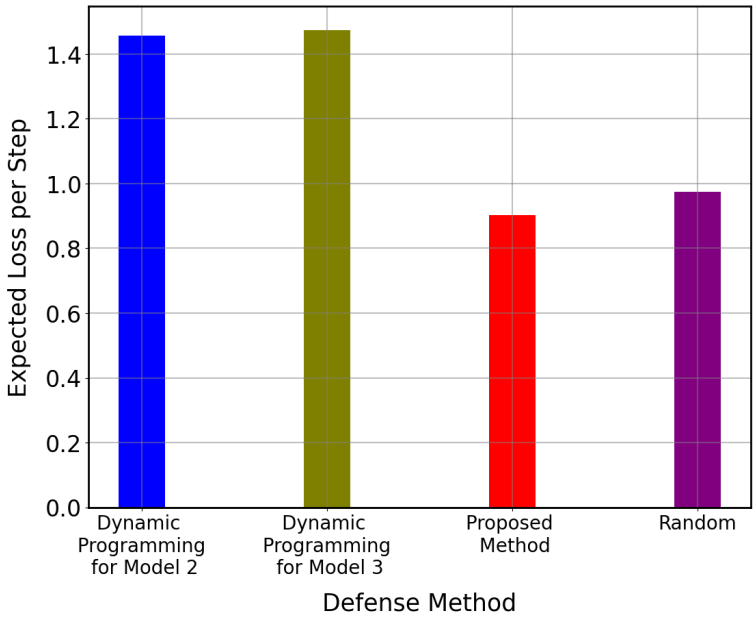



**Fig. 4 Expected loss in security reward per step under different defense policies when model 1 is the true network model.**

In the third experiment represented in Figure 4, we show the expected loss of different policies compared to the best achievable performance. Model 1 is considered as the true network model. The best achievable performance in this case corresponds to the DP policy for model 1, which is referred to as baseline. The increase in the number of compromises per step by different policies with respect to the baseline policy is shown in Figure 4. It can be seen that the proposed policy yields the lowest expected loss in security reward, compared to DP for model 2 and model 3, as well as random policy. This indicates the robustness of the proposed policy, which prevents achieving the worst-case performance due to the lack of knowledge about the true model.

**Second Set of Threat Models: Various Internal Attacks**

In the second set of experiments, a new threat model space is explored. Here, we assume a fixed external vulnerability at the edge nodes $\mathcal{N}_{ex}^{\theta}$ (those exposed to external resources), while the attacker can adopt different propagation strategies within the network, as shown in Figure 5. The figure illustrates a scenario where the network is exposed to one of four potential threat models, though the defender does not know which one will be active during execution. These four distinct models are: Model 1 ($\theta^1$), Model 2 ($\theta^2$), Model 3 ($\theta^3$), and Model 4 ($\theta^4$), forming the threat model space as $\Theta = \{\theta^1, \theta^2, \theta^3, \theta^4\}$. The security reward remains consistent with that of the previous experiments.

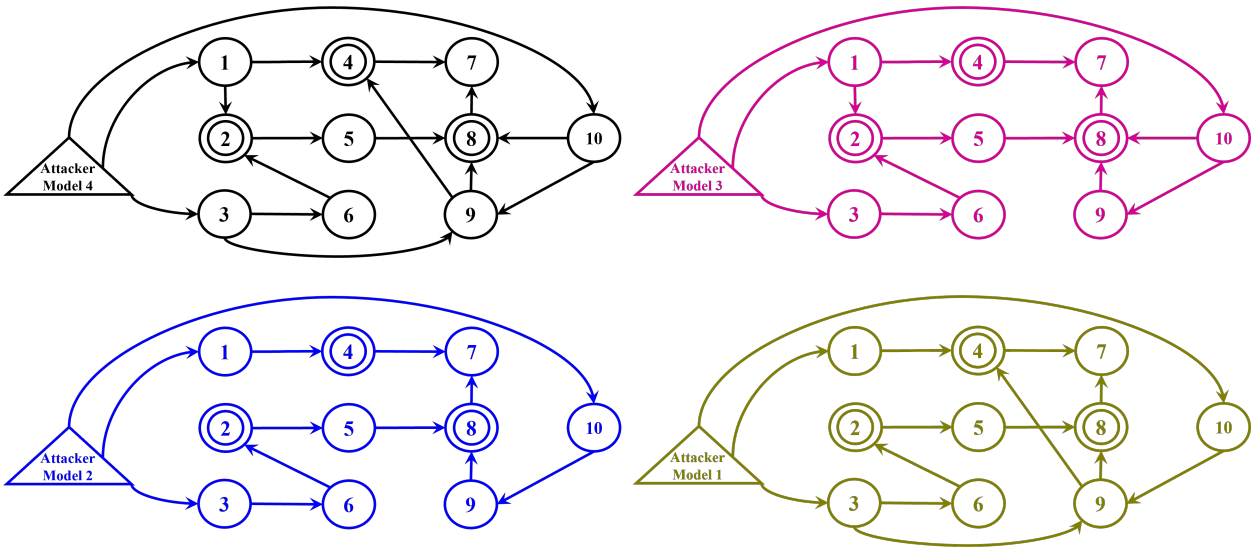



**Fig. 5 A 10-node BAG network with four possible threat models, represented as $\Theta = \{\theta^1, \theta^2, \theta^3, \theta^4\}$.**

In this set of experiments, the external network vulnerabilities are identical in all four models. Specifically, for any $i \in \mathcal{N}_{ex}^{\theta}$, the vulnerabilities are equal across the models, so $\rho_i^{\theta^1} = \rho_i^{\theta^2} = \rho_i^{\theta^3}$, and we denote this common value (without a superscript) as: $\rho_1 = 0.65$, $\rho_3 = 0.6$, and $\rho_{10} = 0.55$. The internal exploit probabilities exhibit both similarities and differences across the models. In scenarios where a node has two outgoing edges and the attacker selects one, the exploit probability on that chosen edge increases. In particular, we assume that the attacker's success rate becomes 1 if it sacrifices the other attack paths and focuses on exploiting a single edge. We use $\rho_{ij}$ to denote the common internal exploit probability, distinguishing any differing values across models with appropriate superscripts. Common internal network vulnerabilities, represented by $\rho_{ij}$, are specified as follows: $\rho_{25}=0.6, \rho_{47}=0.7, \rho_{58}=0.7, \rho_{62}=0.7, \rho_{87}=0.7$. The varying vulnerabilities within the network are outlined as follows:
$\rho_{12}^{\theta^4} = \rho_{12}^{\theta^3} = 0.7, \rho_{14}^{\theta^4} = \rho_{14}^{\theta^3} = 0.6, \rho_{14}^{\theta^1} = \rho_{14}^{\theta^2} = 1, \rho_{36}^{\theta^4} = \rho_{36}^{\theta^1} = 0.55, \rho_{36}^{\theta^2} = \rho_{36}^{\theta^3} = 1, \rho_{39}^{\theta^4} = \rho_{39}^{\theta^1} = 0.7, \rho_{94}^{\theta^4} = \rho_{94}^{\theta^1} = 0.6, \rho_{98}^{\theta^4} = \rho_{98}^{\theta^1} = 0.7, \rho_{98}^{\theta^2} = \rho_{98}^{\theta^3} = 1, \rho_{108}^{\theta^4} = \rho_{108}^{\theta^3} = 0.7, \rho_{109}^{\theta^4} = \rho_{109}^{\theta^3} = 0.4, \rho_{109}^{\theta^1} = \rho_{109}^{\theta^2} = 1.$

Table 1 presents the results when the underlying model is one of the four models in $\Theta$ and remains fixed throughout the defense process, although the defender is unaware of the true model. It can be seen that the dynamic programming policies associated with different threat models behave differently. In particular, performing dynamic programming policy associated with a model that does not match the true threat model can lead to significantly poor defense performance, as indicated by the number of compromises. It can be seen that the proposed policy consistently performed well compared to other model-specific policies. This indicates the importance and applicability of the proposed policy, without the need for assuming a single model, while the model-specific models can have poor performance if applied to

a wrong model.

**Table 1 Comparison of Proposed Method and Dynamic Programming Methods for Different Underlying Models with respect to the average number of compromises.**

| Underlying Model | Proposed Policy | DP Model 1 | DP Model 2 | DP Model 3 | DP Model 4 |
|---|---|---|---|---|---|
| **Model 1** | 2.8468 ± 0.7142 | | 2.7936 ± 0.7398 | 5.288 ± 0.2648 | 3.755 ± 0.5124 |
| **Model 2** | 2.9532 ± 0.7159 | 3.5760 ± 0.5965 | | 5.7232 ± 0.1562 | 5.1722 ± 0.1717 |
| **Model 3** | 2.8654 ± 0.7663 | 3.1924 ± 0.6606 | 2.8694 ± 0.6004 | | 3.1888 ± 0.5888 |
| **Model 4** | 2.7078 ± 0.6931 | 2.2758 ± 0.8233 | 2.7976 ± 0.7350 | 2.91 ± 0.6461 | |

Figure 6 represents the final analysis, which examines the impact of varying the underlying threat models during the defense process. At each step, there is a probability $p$ that the attacker switchs to a new random threat model in the subsequent state. This process can be characterized as a Bernoulli random variable with parameter $p$. When $p = 0$, the underlying model is selected randomly at the beginning of a trajectory. As $p$ increases, the frequency of model switching also rises, reaching a scenario where $p = 1$ signifies that the threat model is chosen randomly at each step. The comparison encompasses our proposed robust method, the average results of the dynamic programming method across all models, and the performance of random action selection at each step. The findings indicate that relying on a single threat model and adapting actions accordingly is generally not effective, despite some BAGs being more or less representative of the overall attacks. In this setting, random action selection yields slightly better average outcomes. However, our proposed robust methods consistently outperform both the single-model approach and random selection.

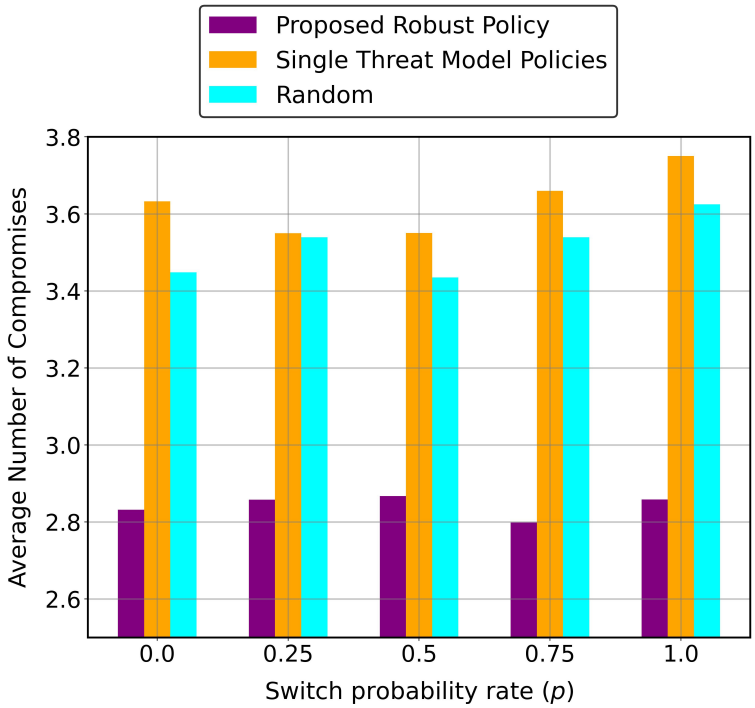



**Fig. 6 Average number of compromises across varying threat model switching rates ($p$), comparing the proposed robust defense, single threat model DP approach, and random defense policies for BAGs.**

## V. Conclusion

In conclusion, this paper develops a robust network security policy for networks with unknown threat models characterized by Bayesian Attack Graphs (BAGs). We introduce an optimal robust defense policy that maximizes expected accumulated rewards under worst-case scenarios, taking into account the uncertainty of network models and dynamic attacker behavior. The proof of the convergence of the proposed policy under any arbitrary network

and threat models is provided. Numerical experiments confirm the effectiveness of the proposed policy, showing its robustness across different threat models. The results demonstrate that our approach outperforms both single-threat model strategies and random defense policies. Additionally, we present a matrix-form implementation of the proposed policy, allowing for efficient offline computation and real-time implementation.

## Acknowledgments

Research was sponsored by the Army Research Office and was accomplished under Cooperative Agreement Number W911NF-24-2-0166. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. This work was also supported by the National Science Foundation award IIS-2311969, ARMY Research Laboratory award W911NF-23-2-0207, ARMY Research Office award W911NF2110299, and Office of Naval Research award N00014-23-1-2850.

## References

[1] Andrews, A., Elakeili, S., Gario, A., and Hagerman, S., "Testing proper mitigation in safety-critical systems: An aerospace Launch application," *IEEE Aerospace Conference Proceedings*, Vol. 2015, 2015.

[2] Elmarady, A. A., and Rahouma, K., "Studying Cybersecurity in Civil Aviation, Including Developing and Applying Aviation Cybersecurity Risk Assessment," *IEEE Access*, Vol. 9, 2021, pp. 143997–144016.

[3] Riahi Manesh, M., and Kaabouch, N., "Analysis of vulnerabilities, attacks, countermeasures and overall risk of the Automatic Dependent Surveillance-Broadcast (ADS-B) system," *International Journal of Critical Infrastructure Protection*, Vol. 19, 2017, pp. 16–31.

[4] Falco, G., and Boschetti, N., *A Security Risk Taxonomy for Commercial Space Missions*, 2021, p. 4241.

[5] Ukwandu, E., Ben-Farah, M. A., Hindy, H., Bures, M., Atkinson, R., Tachtatzis, C., Andonovic, I., and Bellekens, X., "Cyber-security challenges in aviation industry: A review of current and future trends," *Information*, Vol. 13, No. 3, 2022, p. 146.

[6] Maleh, Y., "Machine learning techniques for IoT intrusions detection in aerospace cyber-physical systems," *Machine Learning and Data Mining in Aerospace Technology*, 2020, pp. 205–232.

[7] Manulis, M., Bridges, C. P., Harrison, R., Sekar, V., and Davis, A., "Cyber security in new space: Analysis of threats, key enabling technologies and challenges," *International Journal of Information Security*, Vol. 20, 2021, pp. 287–311.

[8] Ravari, A., Jiang, G., Zhang, Z., Imani, M., Thomson, R. H., Pyke, A. A., Bastian, N. D., and Lan, T., "Adversarial Inverse Learning of Defense Policies Conditioned on Human Factor Models," *2024 58th Asilomar Conference on Signals, Systems, and Computers*, IEEE, 2024.

[9] Lin, Y., Ghoreishi, S. F., Lan, T., and Imani, M., "High-level human intention learning for cooperative decision-making," *2024 IEEE Conference on Control Technology and Applications (CCTA)*, IEEE, 2024, pp. 209–216.

[10] Asadi, N., Hosseini, S. H., Imani, M., Aldrich, D. P., and Ghoreishi, S. F., "Privacy-Preserved Federated Reinforcement Learning for Autonomy in Signalized Intersections," *ASCE International Conference on Transportation and Development (ICTD)*, 2024.

[11] Dhillon, G. S., Kumar, N., and Arora, A., "Firewalls: A Comprehensive Review and Taxonomy," *IEEE Communications Surveys & Tutorials*, Vol. 22, No. 1, 2020, pp. 621–661.

[12] Wang, J., Wang, Y., Hu, H., Sun, Q., Shi, H., and Zeng, L., "Towards a Security-Enhanced Firewall Application for OpenFlow Networks," *Cyberspace Safety and Security*, edited by G. Wang, I. Ray, D. Feng, and M. Rajarajan, Springer International Publishing, 2013, pp. 92–103.

[13] Mukherjee, S., and Sharma, A., "Intrusion detection techniques," *International Journal of Computer Applications*, Vol. 55, No. 2, 2012, pp. 17–22.

[14] Eldefrawy, M. A., Elhoseny, M., and Ramzy, H. M., "A survey on intrusion detection systems: Concepts, types and techniques," *Journal of Network and Computer Applications*, Vol. 131, 2019, pp. 1–23.

[15] Spinsante, S., Chiaraluce, F., and Gambi, E., *Evaluation of AES-Based Authentication and Encryption Schemes for Telecommand and Telemetry in Satellite Applications*, 2006, p. 5558.

[16] Wohlfart, E., Schauer, S., and Holz, T., "Bayesian attack graphs: Security risk assessment via probabilistic modeling," *ACM Transactions on Information and System Security (TISSEC)*, Vol. 20, No. 4, 2017, p. 18.

[17] Wohlfart, E., Schauer, S., and Holz, T., "Bayesian attack graphs: An advanced probabilistic model for security risk assessments," *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, ACM, 2015, pp. 783–794.

[18] Modelo-Howard, G., Bagchi, S., and Lebanon, G., "Determining Placement of Intrusion Detectors for a Distributed Application through Bayesian Network Modeling," *Recent Advances in Intrusion Detection*, edited by R. Lippmann, E. Kirda, and A. Trachtenberg, Springer Berlin Heidelberg, 2008, pp. 271–290.

[19] Zhang, Z., Imani, M., and Lan, T., "Modeling other players with Bayesian beliefs for games with incomplete information," *arXiv preprint arXiv:2405.14122*, 2024.

[20] Asadi, N., and Ghoreishi, S. F., "Active Learning for Efficient Data Acquiring in Coupled Multidisciplinary Systems," 2024.

[21] Zhang, Z., Zhou, H., Imani, M., Lee, T., and Lan, T., "Collaborative AI teaming in unknown environments via active goal deduction," *arXiv preprint arXiv:2403.15341*, 2024.

[22] Ni, Y., Abraham, D., Issa, M., Hernández-Cano, A., Imani, M., Mercati, P., and Imani, M., "Dynamic MAC Protocol for Wireless Spectrum Sharing via Hyperdimensional Self-Learning," *IEEE Access*, 2024.

[23] Lykou, G., Iakovakis, G., and Gritzalis, D., *Aviation Cybersecurity and Cyber-Resilience: Assessing Risk in Air Traffic Management: Theories, Methods, Tools and Technologies*, 2019, pp. 245–260.

[24] Wang, X., Cheng, M., Eaton, J., Hsieh, C.-J., and Wu, F., "Attack graph convolutional networks by adding fake nodes," *arXiv preprint arXiv:1810.10751*, 2018.

[25] Wohlfart, E., Schauer, S., and Holz, T., "Bayesian attack graphs: Security risk assessment and probabilistic graphical models," *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, ACM, 2013, pp. 621–632.

[26] Li, K., and Koutsoukos, X., "Bayesian Attack Graphs: A New Approach for Modeling Security Risks in Computer Networks," *IEEE Transactions on Dependable and Secure Computing*, Vol. 16, No. 3, 2019, pp. 386–399.

[27] Munoz Gonzalez, L., and Lupu, E., "Bayesian attack graphs for security risk assessment," *IST-153 Workshop on Cyber Resilience*, 2016.

[28] Maple, C., Bradbury, M., Le, A., and Ghirardello, K., "A Connected and Autonomous Vehicle Reference Architecture for Attack Surface Analysis," *Applied Sciences*, Vol. 9, 2019, p. 5101.

[29] Sheik, A. T., Atmaca, U., Maple, C., and Epiphaniou, G., "Challenges in Threat Modelling of New Space Systems: A Teleoperation Use-Case," *Advances in Space Research*, Vol. 70, 2022.

[30] Bradbury, M., Maple, C., Yuan, H., Atmaca, U. I., and Cannizzaro, S., "Identifying Attack Surfaces in the Evolving Space Industry Using Reference Architectures," *2020 IEEE Aerospace Conference*, 2020, pp. 1–20.

[31] Mell, P., Scarfone, K., and Romanosky, S., "Common vulnerability scoring system," *IEEE Security & Privacy*, Vol. 4, No. 6, 2006, pp. 85–89.

[32] Kazeminajafabadi, A., and Imani, M., "Optimal monitoring and attack detection of networks modeled by Bayesian attack graphs," *Cybersecurity*, Vol. 6, No. 1, 2023, p. 22.

[33] Kazeminajafabadi, A., Ghoreishi, S. F., and Imani, M., "Optimal Detection for Bayesian Attack Graphs under Uncertainty in Monitoring and Reimaging," *2023 American Control Conference (ACC)*, IEEE, 2024.

[34] KazemiNajafabadi, A., Aksaray, D., and Imani, M., "Defense Policy Optimization with Linear Temporal Logic Specifications for Interconnected Networks," *AIAA SciTech 2025 Forum*, 2025.

[35] Kazeminajafabadi, A., and Imani, M., "Optimal Joint Defense and Monitoring for Networks Security under Uncertainty: A POMDP-Based Approach," *IET Information Security*, Vol. 2024, No. 1, 2024, p. 7966713.

[36] Bellman, R., *Dynamic Programming*, Princeton University Press, 1957.

[37] Banach, S., "Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales," *Fundamenta mathematicae*, Vol. 3, No. 1, 1922, pp. 133–181.