FlipDyn in Graphs: Resource Takeover Games in Graphs

Sandeep Banik[1(✉)], Shaunak D. Bopardikar[2], and Naira Hovakimyan[1]

[1] University of Illinois Urbana-Champaign, Urbana, IL 61801-3633, USA
{baniksan,nhovakim}@illinois.edu
[2] Michigan State University, East Lansing, MI 48823-24, USA
shaunak@egr.msu.edu

Abstract. We present FlipDyn-G, a dynamic game model extending the FlipDyn framework to a graph-based setting, where each node represents a dynamical system. This model captures the interactions between a defender and an adversary who strategically take over nodes in a graph to minimize (resp. maximize) a finite horizon additive cost. At any time, the FlipDyn state is represented as the current node, and each player can transition the FlipDyn state to a node based on the connectivity from the current node. Such transitions are driven by the node dynamics, state, and node-dependent costs. This model results in a hybrid dynamical system where the discrete state (FlipDyn state) governs the continuous state evolution and the corresponding state cost. Our objective is to compute the Nash equilibrium of this finite horizon zero-sum game on a graph. Our contributions are two-fold. First, we model and characterize the FlipDyn-G game for general dynamical systems, along with the corresponding Nash equilibrium (NE) takeover strategies. Second, for scalar linear discrete-time dynamical systems with quadratic costs, we derive the NE takeover strategies and saddle-point values independent of the continuous state of the system. Additionally, for a finite state birth-death Markov chain (represented as a graph) under scalar linear dynamical systems, we derive analytical expressions for the NE takeover strategies and saddle-point values. We illustrate our findings through numerical studies involving epidemic models and linear dynamical systems with adversarial interactions.

Keywords: Game Theory · Graphs · Dynamical Systems

1 Introduction

Cyber-Physical Systems (CPS) are essential for integrating computational elements with physical processes, enabling advanced functionalities in various domains. Examples include smart grids for efficient energy distribution, autonomous vehicles for navigation and safety, and industrial automation systems for enhanced productivity through precise control and sensor feedback [1,2].

A. Sinha et al. (Eds.): GameSec 2024, LNCS 14908, pp. 220–239, 2025.
https://doi.org/10.1007/978-3-031-74835-6_11

In the context of CPS, each node in a graph can be represented as a dynamical process, such as the generation and consumption of electricity in smart grids, the motion dynamics of autonomous vehicles, or the operational processes in industrial automation. These dynamical processes are interconnected through edges that represent the interactions and dependencies between them. For instance, in a smart grid, nodes may represent dynamic processes of energy generation and consumption at different substations, while edges denote the power flow between these substations [3]. Similarly, in autonomous vehicle networks, nodes could represent the dynamic driving processes of individual vehicles, with edges capturing the communication and coordination required for safe and efficient traffic flow [4–6].

The use of graphs in modeling CPS is crucial for understanding the system's overall behavior and ensuring its robust operation. Graphs facilitate the visualization and analysis of how individual dynamic processes interconnect to form a larger, cohesive system. This interconnection highlights the importance of securing these nodes and their interactions to prevent disruptions that could compromise the entire system [7,8].

Securing CPS critically involves mitigating the risks of stealthy takeovers, where an adversary covertly gains control of system components. The `FlipIT` game [9] provides a framework for analyzing such scenarios, where both the attacker and defender can stealthily control a static resource without the other party's immediate knowledge. This model captures the continuous and covert nature of security threats in CPS, highlighting the need for persistent vigilance and strategic defense mechanisms.

The `FlipIT` framework was extended to dynamical systems in `FlipDyn` [10], where a defender and adversary aim to take over a common resource modeled as a discrete-time dynamical system over a finite horizon. Building on `FlipDyn`, this paper focuses on resource takeovers in graphs, where each node represents a resource with its own dynamics, and nodes are connected by edges reflecting CPS interactions. Two players, a defender and an adversary, seek to repeatedly take over the graph's resources. This setup captures strategic interactions in a dynamic, interconnected environment, generalizing the `FlipDyn` framework to multiple states.

Analyzing takeover games involves understanding optimal strategies for both the adversary and defender, considering various graph topologies and CPS characteristics. By leveraging game-theoretic models and topology structures, this paper proposes robust defense mechanisms to enhance CPS resilience against takeover attacks. This approach is crucial for ensuring the continued reliability and safety of essential infrastructures amidst emerging cyber threats.

1.1 Related Works

The seminal `FlipIT` [9] analyzes a two-player zero-sum game between a defender and an adversary attempting to take over a static resource, such as a computing device, virtual machine, or cloud service [11]. The work of `FlipIT` was generalized to the games of timing [12], where the actions of each player are

dependent on the available exploitable vulnerability, and extended to include time-based exponential discounting [13]. `FlipThem` [14] expanded the game to multiple resources with AND/OR models. The work in `FlipThem` was extended to i) a threshold-based version [15], which considered a finite number (threshold) of resources beyond which there exists no incentives to takeover, ii) multiple resource with constraints on the frequency of takeover actions [16], and ii) heterogeneous resource costs and a learning-based method to determine player strategies [17]. Similar extensions include, Cheat-FlipIt model [18], in which the opponent agent may feint to flip the resources first, and then control the resources after a finite delay. Such takeover strategies have also impacted the blockchain system [19], where arbitrage bots in decentralized exchanges engage in priority gas auctions to exploit against ordinary users. Beyond the domain of cybersecurity, the `FlipIT` model has been introduced in supervisory control and data acquisition (SCADA) to evaluate the impact of cyberattacks with insider assistance. The model of `FlipIT` has been extensively applied in system security [11]. These works primarily focused on resource takeovers within a static system, lacking consideration for the dynamic evolution of physical systems. In contrast, our work incorporates the dynamics of a physical system in the game of resource takeovers between an adversary and a defender, addressing the need for strategies that account for the continuous and evolving nature of CPS.

A finite-horizon zero-sum stochastic game has been used to analyze probabilistic reachable sets for discrete-time stochastic hybrid systems [20], where both players act simultaneously. Conversely, controllers have been synthesized [21] for intermittent switching between a defender and an adversary in discrete-time systems with multi-dimensional control inputs and constraints [22]. Such takeovers correspond to covert misappropriation of a plant [23], where an attacker controls the plant while remaining hidden from the supervisory system, extending these attacks to load frequency control (LFC) systems [24]. Unlike previous research, our paper provides a feedback signal to infer control and allows taking control of the plant at any instant, balancing operational cost and performance.

The `FlipNet` model [25] extends `FlipIT` to a graph, representing a networked system of multiple resources, where each player can take over nodes. Network security in graphs is also viewed as advanced persistent threats (APT), modeled as a zero-sum repeated game with states as compromised edges [26]. Similarly, APTs are modeled as multi-stage zero-sum network hardening games, where the adversary finds the shortest path and the defender allocates resources to block it. Recently, dynamic information flow tracking has been proposed to detect APTs via a multistage game [27]. A similar APT model is explored in Cut-the-Rope [28], where the defender cuts the backdoor access of an adversary, demonstrating efficacy on attack graphs in the robotics domain [29]. `FlipIT` has also been used to study malware diffusion in epidemic models [30]. This paper addresses `FlipIT` in a graph-based setting, where the defender and adversary repeatedly aim to take over nodes. Unlike previous works, this zero-sum game is played over a finite horizon with a discrete-time dynamical process on each node and time-varying costs.

Our prior work which extends the FlipIT model to incorporate dynamical systems, termed FlipDyn [10]. The model of FlipDyn was extended to jointly solve the takeover and control policy [31]. In this paper, we extend the FlipDyn model to a finite horizon zero-sum game over a graph, where each node represents a dynamical system and the edges correspond to the interaction between these systems. The contributions of this work are two-fold:

1. **Takeover strategies over a graph with discrete-time dynamical system on nodes**: We formulate a two-player zero-sum takeover game involving a defender and an adversary seeking to takeover the nodes of a graph, representing a discrete-time dynamical systems. The costs incurred by each player are contingent on the current node of the graph. Assuming knowledge of the discrete-time dynamics, we establish the Nash equilibrium (NE) takeover strategies and saddle-point values.
2. **State-independent takeover strategies and saddle-point values for scalar/1− dimensional systems**: For a linear discrete-time scalar dynamical system with quadratic takeover and state costs, we determine NE takeover policies independent of the continuous state of both players. Furthermore, for a topology representing a finite state birth-death process, termed *dual deter model*, we derive analytical expression of the NE takeover policies and saddle-point values.

We illustrate our results on an epidemic model with no node dynamics and on an example from finance.

This paper is structured as follows. Section 2 formally defines the FlipDyn problem in a graph setting with continuous state and node dependent costs. In Sect. 3, we outline a solution methodology applicable to general discrete-time dynamical systems on nodes. Section 4 presents a solution for takeover policies for linear scalar discrete-time dynamical systems featuring quadratic costs, along with a topology dependent analytical solution and numerical examples in Sect. 5. The paper concludes with a discussion on future directions in Sect. 6.

2 Problem Formulation

Consider a directed multigraph $\mathcal{G} := \{V, E, \phi\}$, where V is the set of nodes with $|V| \in \mathbb{N}^+$, E is the set of edges (paired nodes), and $\phi : E \to \{\{\alpha, \beta\} | \alpha, \beta \in V^2\}$ is the incidence function mapping every edge to an ordered pair of nodes, defining the connectivity of the graph. The term $e_{\alpha,\beta} \in E$ represents the edges connecting the node $\alpha \in V$ with the node $\beta \in V$, such that when $\alpha = \beta$, it represents a self-loop. We consider a single adversary, originating from any node of the graph $\mathcal{G}$. The adversary's goal is to reach nodes within the graph which induces maximum cost, while a defender's mission is to hinder the adversary's advances.

We model the actions of the players and state evolution in discrete-time, with the variable k denoting the current time step, which takes on values from the set $\mathcal{K} := \{1, 2, \ldots, L, L+1\}$. We represent the current node at time k using

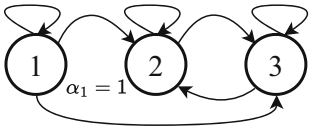


Fig. 1. A directed multigraph consisting of 3 nodes. At time $k = 1$, the `FlipDyn` state is $\alpha_1 = 1$. The actions of both players are $\{1, 2, 3\}$.

a variable $\alpha_k \in V$, referred to as the `FlipDyn` state. The adversary's action is denoted by the variable $\pi_k^{\mathrm{a}} \in \epsilon(\alpha_k)$, where the set $\epsilon(\alpha_k)$ is defined as:

$$\epsilon(\alpha_k) := \{j \in V | e_{\alpha_k, j} \in E\}.$$

Here, $\epsilon(\alpha_k)$ represents the nodes the adversary can potentially target from the current node α_k at time k, with $j = \alpha_k$ indicating the choice to remain idle or stay in the same node. Similarly, the defender's action is denoted by $\pi_k^{\mathrm{d}} \in \epsilon(\alpha_k)$. Notice that the defender's action set is identical to that of the adversary's, to deter or prevent further escalation. The `FlipDyn` state update is based on both the action of the defender and adversary, given by:

$$\alpha_{k+1} = \begin{cases} \pi_k^d, & \text{if } \pi_k^{\mathrm{d}} = \pi_k^{\mathrm{a}}, \\ \pi_k^{\mathrm{d}}, & \text{else if } \pi_k^{\mathrm{d}} \in \{\epsilon(\alpha_k) | \pi_k^{\mathrm{a}} = \alpha_k\}, \\ \pi_k^{\mathrm{a}}, & \text{else if } \pi_k^{\mathrm{a}} \in \{\epsilon(\alpha_k) | \pi_k^{\mathrm{d}} = \alpha_k\}, \\ \alpha_k, & \text{otherwise.} \end{cases} \tag{1}$$

The `FlipDyn` update (1) states that if the actions of both the defender and adversary are identical, then the `FlipDyn` state remains unchanged. However, if the defender opts to choose any node while the adversary remains idle, then the `FlipDyn` state transitions into the chosen node. Similarly, if the defender remains idle while the adversary chooses any node, then the `FlipDyn` transitions to the chosen node. The `FlipDyn` state transition can be compactly written as:

$$\alpha_{k+1} = -\alpha_k \mathbf{1}_{\alpha_k}(\pi_k^{\mathrm{a}}) \mathbf{1}_{\alpha_k}(\pi_k^{\mathrm{d}}) + \mathbf{1}_{\alpha_k}(\pi_k^{\mathrm{a}}) \pi_k^{\mathrm{d}} + \mathbf{1}_{\alpha_k}(\pi_k^{\mathrm{d}}) \pi_k^{\mathrm{a}} + \bar{\mathbf{1}}_{\alpha_k}(\pi_k^{\mathrm{a}}) \bar{\mathbf{1}}_{\alpha_k}(\pi_k^{\mathrm{d}}) \pi_k^d, \tag{2}$$

where $\mathbf{1}_{\alpha_k} : \epsilon(\alpha_k) \to \{0, 1\}$ is the indicator function, which maps to one if $\pi_k^{\mathrm{d}} = \alpha_k$ or $\pi_k^{\mathrm{a}} = \alpha_k$, and maps to zero, otherwise. The term $\bar{\mathbf{1}}_{\alpha_k}$ is the one's complement of $\mathbf{1}_{\alpha_k}$. For illustrative purpose, consider the graph shown in Fig. 1 with the `FlipDyn` state at time $k = 1$ as $\alpha_1 = 1$. The `FlipDyn` state can transition to the node 2, 3 or remain in node 1 based on the update equation (1).

In addition to the described graph environment, there is an underlying dynamical system whose continuous state at time k is indicated by $x_k \in \mathcal{X} \subseteq \mathbb{R}^n$, where $\mathcal{X}$ denotes the Euclidean state space. The state transition is dependent on the node α_{k+1} given by:

$$x_{k+1} = F_k^{\alpha_{k+1}}(x_k), \tag{3}$$

where $F_k^{\alpha_{k+1}} : \mathcal{X} \to \mathcal{X}$ is the transition function for each $k \in \mathcal{K}$ and $\alpha_{k+1} \in V$.

Our objective is to compute a strategy for both the players to transition the FlipDyn state to different nodes of the graph based on the dynamics (2), (3), takeover, and state costs. Given the initial state x_1 and node α_1, we pose the node takeover problem as a zero-sum dynamic game governed by the FlipDyn update (2) and state dynamics (3), over a finite-time L, where the defender aims to minimize an additive cost given by:

$$J(\alpha_1, x_1, \{\pi_{\mathbf{L}}^{\mathrm{a}}\}, \{\pi_{\mathbf{L}}^{\mathrm{d}}\}) = g_{L+1}^{\alpha_{L+1}}(x_{L+1}) + \sum_{t=1}^{L} g_t^{\alpha_t}(x_t) + \bar{\mathbf{1}}_{\alpha_t}(\pi_t^{\mathrm{d}}) d_t^{\pi_t^{\mathrm{d}}}(x_t) - \bar{\mathbf{1}}_{\alpha_t}(\pi_t^{\mathrm{a}}) a_t^{\pi_t^{\mathrm{a}}}(x_t), \tag{4}$$

where $g_t^{\alpha_t}(x_t) : \mathcal{X} \to \mathbb{R}$ represents the cost for every FlipDyn state $\alpha_t \in V$, continuous state x_t at time $t \in \mathcal{K}$, with $g_{L+1}^{\alpha_{L+1}}(x_{L+1}) : \mathcal{X} \to \mathbb{R}$ representing the terminal cost for each $\alpha_{L+1} \in V$. The terms $d_t^{\pi_t^{\mathrm{d}}}(x_t) : \mathcal{X} \to \mathbb{R}$ and $a_t^{\pi_t^{\mathrm{a}}}(x_t) : \mathcal{X} \to \mathbb{R}$ represent the instantaneous takeover costs of the defender and adversary, respectively, for each $t \in \mathcal{K}$ and action $\pi_t^{\mathrm{d}}, \pi_t^{\mathrm{a}} \in \epsilon(\alpha_t)$. The defender and adversary actions over the finite-horizon L is given by the notations $\{\pi_{\mathbf{L}}^{\mathrm{a}}\} := \{\pi_1^{\mathrm{a}}, \ldots, \pi_L^{\mathrm{a}}\}$, and $\{\pi_{\mathbf{L}}^{\mathrm{d}}\} := \{\pi_1^{\mathrm{d}}, \ldots, \pi_L^{\mathrm{d}}\}$, respectively. In contrast, the adversary aims to maximize the cost function (4) leading to a zero-sum dynamic game. This formulation characterizes the strategic interaction between the two players in the context of a node takeover problem in a graph environment, termed as FlipDyn-G game.

We seek to find Nash Equilibrium (NE) solutions of the game (4). To guarantee the existence of a pure or mixed NE takeover strategy, we expand the set of player policies to behavioral strategies – probability distributions over the space of discrete actions at each discrete time [32]. Specifically, let

$$\mathbf{y}_k^{\alpha_k} := \{y_{k,j}^{\alpha_k} | j \in \epsilon(\alpha_k)\}, \sum_{j \in \epsilon(\alpha_k)} y_{k,j}^{\alpha_k} = 1, y_{k,j}^{\alpha_k} \geq 0, \text{ and} \tag{5}$$

$$\mathbf{z}_k^{\alpha_k} := \{z_{k,j}^{\alpha_k} | j \in \epsilon(\alpha_k)\}, \sum_{j \in \epsilon(\alpha_k)} z_{k,j}^{\alpha_k} = 1, z_{k,j}^{\alpha_k} \geq 0 \tag{6}$$

be the behavioral strategies for the defender and adversary, respectively, at time instant k for the FlipDyn state α_k. The takeover actions are

$$\pi_k^{\mathrm{d}} \sim \mathbf{y}_k^{\alpha_k}, \quad \pi_k^{\mathrm{a}} \sim \mathbf{z}_k^{\alpha_k},$$

for the defender and adversary at any time k are sampled from the corresponding behavioral strategy. The behavioral strategies are $y_k^{\alpha_k}, z_k^{\alpha_k} \in \Delta_{|\epsilon(\alpha_k)|}$, where $\Delta_{|\epsilon(\alpha_k)|}$ is the probability simplex in $|\epsilon(\alpha_k)|$ dimensions. Over the finite horizon L, let $y_{\mathbf{L}} := \{\mathbf{y}_1^{\alpha_1}, \mathbf{y}_2^{\alpha_2}, \ldots, \mathbf{y}_L^{\alpha_L}\} \in \Delta_{|\epsilon(\alpha_1)|} \times \Delta_{|\epsilon(\alpha_2)|} \times \cdots \times \Delta_{|\epsilon(\alpha_L)|}$ and $z_{\mathbf{L}} := \{\mathbf{z}_1^{\alpha_1}, \mathbf{z}_2^{\alpha_2}, \ldots, \mathbf{z}_L^{\alpha_L}\} \in \Delta_{|\epsilon(\alpha_1)|}^L \times \Delta_{|\epsilon(\alpha_2)|}^L \times \cdots \times \Delta_{|\epsilon(\alpha_L)|}^L$ be the sequence of defender and adversary behavioral strategies. Thus, the expected outcome of the zero-sum game (4) is given by:

$$J_E(x_1, \alpha_1, y_{\mathbf{L}}, z_{\mathbf{L}}) := \mathbb{E}[J(x_1, \alpha_1, \{\pi_L^{\mathrm{a}}\}, \{\pi_L^{\mathrm{d}}\})], \tag{7}$$

where the expectation is computed with respect to the distributions $y_{\mathbf{L}}$ and $z_{\mathbf{L}}$. Specifically, we seek a saddle-point solution $(y^*_{\mathbf{L}}, z^*_{\mathbf{L}})$ in the space of behavioral strategies such that for any non-zero initial state $x_1 \in \mathcal{X}, \alpha_1 \in V$, we have:

$$J_E(x_1, \alpha_1, y^*_{\mathbf{L}}, z_{\mathbf{L}}) \leq J_E(x_1, \alpha_1, y^*_{\mathbf{L}}, z^*_{\mathbf{L}}) \leq J_E(x_1, \alpha_1, y_{\mathbf{L}}.z^*_{\mathbf{L}}).$$

The FlipDyn game over a graph is completely defined by the expected cost (7) and the space of player takeover strategies subject to the dynamics in (2) and (3). In the next section, we derive the outcome of the FlipDyn game for each node in the graph for general systems.

3 FlipDyn-G for General Problem

3.1 Saddle-Point Value of Any Node

At time instant $k \in \mathcal{K}$, given a FlipDyn state α_k, the saddle-point value comprises the instantaneous state cost and an additive cost-to-go based on the players takeover actions. The cost-to-go is determined via a cost-to-go matrix in each of the FlipDyn state α_k, represented by $\Xi^{\alpha_k}_{k+1} \in \mathbb{R}^{|\epsilon(\alpha_k)|\times|\epsilon(\alpha_k)|}$. Let $V^{\alpha_k}_k(x, \Xi^{\alpha_k}_{k+1})$ be the saddle-point value at time instant k with the continuous state x and cost-to-go matrix, corresponding to the FlipDyn state of α_k. Let us define a set of nodes connected to α_k as, $\{\alpha_k, j_2, j_3, \ldots, j_{m(\alpha_k)}\} \in \epsilon(\alpha_k)$, where $m(\alpha_k) = |\epsilon(\alpha_k)|$. Such a set of nodes will help us define the cost-to-go matrix. The entries of the cost-to-go matrix $\Xi^{\alpha_k}_{k+1}$ corresponding to each pair of takeover actions are:

$$\begin{array}{c} \\ \alpha_k \\ j_2 \\ \ldots \\ j_{m(\alpha_k)} \end{array} \underbrace{\begin{array}{c} \begin{array}{cccc} \alpha_k & j_2 & \ldots & j_{m(\alpha_k)} \end{array} \\ \left[\begin{array}{cccc} v^{\alpha_k}_{k+1}(\alpha_k, \alpha_k) & \ldots & \ldots & v^{j_{m(\alpha_k)}}_{k+1}(\alpha_k, j_{m(\alpha_k)}) \\ v^{j_2}_{k+1}(j_2, \alpha_k) & v^{j_2}_{k+1}(j_2, j_2) & \ldots & v^{\alpha_k}_{k+1}(j_2, j_{m(\alpha_k)}) \\ \ldots & \ldots & \ldots & \ldots \\ v^{j_{m(\alpha_k)}}_{k+1}(j_{m(\alpha_k)}, \alpha_k) & v^{\alpha_k}_{k+1}(j_{m(\alpha_k)}, j_2) & \ldots & v^{j_{m(\alpha_k)}}_{k+1}(j_{m(\alpha_k)}, j_{m(\alpha_k)}) \end{array}\right] \end{array}}_{\Xi^{\alpha_k}_{k+1}}, \quad (8)$$

where $v^{\alpha_{k+1}}_{k+1}(\pi^{\mathrm{d}}_k, \pi^{\mathrm{a}}_k)$ corresponds to the cost-to-go value of a FlipDyn state $\alpha_{k+1} \in V$, defined as:

$$v^{\alpha_{k+1}}_{k+1}(\pi^{\mathrm{d}}_k, \pi^{\mathrm{a}}_k) := V^{\alpha_{k+1}}_{k+1}(F^{\alpha_{k+1}}_k(x), \Xi^{\alpha_{k+2}}_{k+2}) + \bar{\mathbf{1}}_{\alpha_k}(\pi^{\mathrm{d}}_k) d^{\pi^{\mathrm{d}}_k}_k(x_k) - \bar{\mathbf{1}}_{\alpha_k}(\pi^{\mathrm{a}}_k) a^{\pi^{\mathrm{a}}_k}_k(x_k).$$

The diagonal terms in (8) correspond to the saddle-point value of the FlipDyn states under identical defender and adversary actions. Notice, only under the action of $\pi^{\mathrm{d}}_k = \pi^{\mathrm{a}}_k = \alpha_k$ the takeover costs for both players are zero. The first row of $\Xi^{\alpha_k}_{k+1}$ corresponds to the saddle-point values of FlipDyn states chosen by the adversary, when the defender remains idle. Similarly, the first column corresponds to the saddle-point value of the FlipDyn states chosen by the defender under an idle adversary action. The remaining entries of $\Xi^{\alpha_k}_{k+1}$ correspond to

the saddle-point value of the FlipDyn state α_k with the corresponding takeover costs. The entries of the cost-to-go matrix are constructed using the FlipDyn dynamics (2) and continuous state dynamics (3). Thus, at time k for a given state x and α_k, the saddle-point value satisfies

$$V_k^{\alpha_k}(x, \Xi_{k+1}^{\alpha_k}) = g_k^{\alpha_k}(x) + \mathrm{Val}(\Xi_{k+1}^{\alpha_k}), \tag{9}$$

where $\mathrm{Val}(X_{k+1}^{\alpha_k}) := \min_{y_k^{\alpha_k}} \max_{z_k^{\alpha_k}} {y_k^{\alpha_k}}^{\mathrm{T}} X_{k+1} z_k^{\alpha_k}$, represents the (mixed) saddle-point value of the zero-sum matrix X_{k+1} for the FlipDyn state α_k, and $\Xi_{k+1}^{\alpha_k} \in \mathbb{R}^{|\epsilon(\alpha_k)| \times |\epsilon(\alpha_k)|}$ is the cost-to-go zero-sum matrix. The defender's and adversary's action results in either an entry within $\Xi_{k+1}^{\alpha_k}$ (if the matrix has a saddle point in pure strategies) or in the expected sense, resulting in a cost-to-go from state x at time k.

With the saddle-point values established in each of the FlipDyn states $\alpha_k \in V$, next, we will characterize the NE takeover strategies and the saddle-point values for the entire time horizon L.

3.2 NE Takeover Strategies of the FlipDyn-G game

To characterize the saddle-point value of the game, we restrict the state and takeover costs to a particular domain, stated in the following mild assumption.

Assumption 1. *[Non-negative costs] For any time instant $k \in \mathcal{K}$, the state and takeover costs $g_k^{\alpha}(x), d_k^{\alpha}(x), a_k^{\alpha}(x)$, for all $x \in \mathcal{X}$, and $\alpha \in V$ are non-negative ($\mathbb{R}_{\geq 0}$).*

Assumption 1 enables us to compare the entries of the cost-to-go matrix without changes in the sign of the costs, thereby, characterizing the strategies of the players (pure or mixed strategies). Under Assumption 1, we derive the following result to compute a recursive saddle-point value for the horizon length L and the corresponding NE takeover strategies for both the players in every node of the graph environment.

Lemma 1. *Under Assumption 1, the saddle-point value of the FlipDyn-G game (7) at any time $k \in \mathcal{K}$, subject to the FlipDyn dynamics (2) and continuous state dynamics (3) is given by:*

$$V_k^{\alpha_k *}(x, \Xi_{k+1}^{\alpha_k}) = g_k^{\alpha_k} + {y_k^{\alpha_k *}}^{\mathrm{T}} \Xi_{k+1}^{\alpha_k} z_k^{\alpha_k *}, \tag{10}$$

where $y_k^{\alpha_k *}$ and $z_k^{\alpha_k *}$ correspond to NE takeover policies obtained upon solving the zero-sum matrix defined by $\Xi_{k+1}^{\alpha_k}$ (cost-constructed backward in time using the saddle-point values at $k+1$) as a linear program [32]. The boundary condition of the saddle-point value recursion (10) at $k = L$ is given by:

$$\Xi_{L+2}^{\alpha_{L+1}} := \mathbf{0}_{m(\alpha_{L+1}) \times m(\alpha_{L+1})}, \forall \alpha_{L+1} \in V. \tag{11}$$

We skip the proof of the Lemma 1 as it involves simple substitutions and the use of recursive optimality. For a finite cardinality of the state space $\mathcal{X}$, `FlipDyn` states V, and a finite horizon L, Lemma 1 yields an exact (behavioral) saddle-point value of the `FlipDyn-G` game (7). However, the computational and storage complexities scale undesirably with the cardinality of $\mathcal{X}$, especially in continuous state spaces. For this purpose, in the next section, we will provide a parametric form of the saddle-point value especially in the case of scalar linear dynamics with quadratic costs.

4 `FlipDyn-G` for scalar LQ Problems

To render a tractable solution for continuous state of the `FlipDyn-G` game, we restrict ourselves to scalar linear discrete-time dynamical system with quadratic costs (LQ problem). The discrete-time dynamics of a linear system at time instant $k \in \mathcal{K}$ in the `FlipDyn` state α_{k+1} is given by:

$$x_{k+1} = F_k^{\alpha_{k+1}}(x_k) := f_k^{\alpha_{k+1}} x_k, \tag{12}$$

where $f_k^{\alpha_{k+1}} \in \mathbb{R}$ denotes the state transition scalar coefficient. The stage and takeover costs are assumed to be quadratic for each player and given by:

$$g_k^{\alpha_k}(x) = x^2 \mathbf{g}_k^{\alpha_k}, \quad d_k^{\alpha_k}(x) = x^2 \mathbf{d}_k^{\alpha_k}, \quad a_k^{\alpha_k}(x) = x^2 \mathbf{a}_k^{\alpha_k}, \tag{13}$$

where $\mathbf{g}_k^{\alpha_k} \in \mathbb{R}, \mathbf{a}_k^{\alpha_k} \in \mathbb{R}, \mathbf{d}_k^{\alpha_k} \in \mathbb{R}$ are non-negative ($\mathbb{R}_{\geq 0}$) under Assumption 1.

Under Assumption 1 for scalar dynamical systems of the form (12), we postulate a parametric form for the saddle-point value for each `FlipDyn` state $\alpha \in V$ of the form:

$$V_k^{\alpha_k}(x, \Xi_{k+1}^{\alpha_k}) \Rightarrow V_k^{\alpha_k}(x) := \mathbf{p}_k^{\alpha_k} x^2, \ \forall \alpha_k \in V, \ k \in \mathcal{K}, \tag{14}$$

where $\mathbf{p}_k^{\alpha_k} \in \mathbb{R}_{\geq 0}$ corresponds to a non-negative coefficient for each of the `FlipDyn` states. Under the scalar linear dynamical system (12), takeover costs (13) and the parameteric form (14), the cost-to-go matrix $\hat{\Xi}_{k+1}^{\alpha_k}$ can be re-expressed as:

$$\begin{array}{c} \\ \alpha_k \\ j_2 \\ \dots \\ j_{m(\alpha_k)} \end{array} \underbrace{\begin{array}{c} \begin{array}{cccc} \alpha_k & j_2 & \dots & j_{m(\alpha_k)} \end{array} \\ \begin{bmatrix} \mathbf{v}_{k+1}^{\alpha_k}(\alpha_k, \alpha_k) & \dots & \dots & \mathbf{v}_{k+1}^{j_{m(\alpha_k)}}(\alpha_k, j_{m(\alpha_k)}) \\ \mathbf{v}_{k+1}^{j_2}(j_2, \alpha_k) & \mathbf{v}_{k+1}^{j_2}(j_2, j_2) & \dots & \mathbf{v}_{k+1}^{\alpha_k}(j_2, j_{m(\alpha_k)}) \\ \dots & \dots & \dots & \dots \\ \mathbf{v}_{k+1}^{j_{m(\alpha_k)}}(j_{m(\alpha_k)}, \alpha_k) & \mathbf{v}_{k+1}^{\alpha_k}(j_{m(\alpha_k)}, j_2) & \dots & \mathbf{v}_{k+1}^{j_{m(\alpha_k)}}(j_{m(\alpha_k)}, j_{m(\alpha_k)}) \end{bmatrix} \end{array}}_{\hat{\Xi}_{k+1}^{\alpha_k}}, \tag{15}$$

where $\mathbf{v}_{k+1}^{\alpha_k}(u, w)$ corresponds to the cost-to-go term of a `FlipDyn` state independent of the term x^2, defined as:

$$\mathbf{v}_{k+1}^{\alpha_{k+1}}(\pi_k^{\mathrm{d}}, \pi_k^{\mathrm{a}}) := (f_k^{\alpha_{k+1}})^2 \mathbf{p}_{k+1}^{\alpha_{k+1}} + \bar{\mathbf{1}}_{\alpha_k}(\pi_k^{\mathrm{d}}) \mathbf{d}_k^{\pi_k^{\mathrm{d}}} - \bar{\mathbf{1}}_{\alpha_k}(\pi_k^{\mathrm{a}}) \mathbf{a}_k^{\pi_k^{\mathrm{a}}}.$$

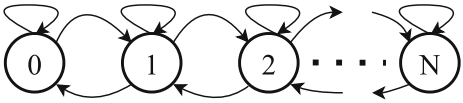


Fig. 2. A graph consisting of N nodes .

Notice the cost-to-go entries consists of the system transition coefficients and takeover costs, while factoring out the term x^2. Building on Lemma 1, we present the following result, which provides the NE takeover policies of both players, and outlines the saddle-point value update of $\mathbf{p}_k^{\alpha_k}$ for any FlipDyn state.

Lemma 2. *Under Assumption 1, at any time $k \in \mathcal{K}$, the saddle-point value parameter of the FlipDyn-G game (7) for quadratic state and takeover costs (13), subject to the FlipDyn dynamics (2) and scalar state dynamics (12), is given by:*

$$\mathbf{p}_k^{\alpha_k *} = g_k^{\alpha_k} + {y_k^{\alpha_k *}}^{\mathrm{T}} \hat{\Xi}_{k+1}^{\alpha_k} z_k^{\alpha_k *}, \tag{16}$$

where $y_k^{\alpha_k *}$ and $z_k^{\alpha_k *}$ correspond to NE takeover policies obtained upon solving the zero-sum matrix $\hat{\Xi}_{k+1}^{\alpha_k}$ as a linear program [32]. The boundary condition of the saddle-point value recursion (10) at $k = L$ is given by:

$$\hat{\Xi}_{L+2}^{\alpha_{L+1}} := \mathbf{0}_{m(\alpha_{L+1}) \times m(\alpha_{L+1})}, \forall \alpha_{L+1} \in V. \tag{17}$$

Substituting the scalar state dynamics (12) along with state and takeover costs (13) yields the NE strategies and saddle-point value parameters (16). We skip the proof of Lemma 2 for brevity. Lemma 2 presents a complete solution for the FlipDyn-G (7) game with NE takeover strategies independent of state of the scalar dynamical system. In the following subsection, we will derive closed-form expressions of the FlipDyn-G game for a special graph structure and show how the structure represents the original FlipDyn game [10].

4.1 Dual Deter FlipDyn-G game

We examine a special case of the graph environment, termed the *dual deter* model, which consists of a start and end node each connecting to only one other node, while the remaining nodes connect to two different nodes. This model can be viewed as a finite state Markov chain birth-death process [33]. We assume the dual deter model has an ordered set of nodes from node 0 to N, resulting in a total of $|V| = N + 1$ nodes, as illustrated in Fig. 2.

A key difference compared to the general graph model lies in the action space of the defender and adversary. At any node $\alpha_k \in \{1, 2, \ldots, N-1\}$, the action space of the adversary is $\pi_k^{\mathrm{a}} := \{\alpha_k, \overline{\alpha}\}$, $\overline{\alpha} \in \{V | \overline{\alpha} > \alpha_k\}$, and of the defender is $\pi_k^{\mathrm{d}} := \{\alpha_k, \underline{\alpha}\}$, $\underline{\alpha} \in \{V | \underline{\alpha} < \alpha_k\}$. The action space of both the defender and adversary in the start and end node $\alpha_k = \{0, N\}$ are given as $\pi_k^{\mathrm{d}} := \{\alpha_k, \tau\}$ and $\pi_k^{\mathrm{a}} := \{\alpha_k, \tau\}$, where τ represents a takeover action in the node α_k, preventing transition to other nodes. Such an action space and model represents the defender

deterring an adversary from escalating through the graph. The `FlipDyn` state updates in such a dual deter model as follows:

$$\alpha_{k+1} = \begin{cases} \alpha_k, & \text{if } \pi_k^{\mathrm{d}} = \pi_k^{\mathrm{a}} | \alpha_k = \{0, N\}, \\ 1, & \text{else if } \pi_k^{\mathrm{a}} = \tau | \pi_k^{\mathrm{d}} = 0, \alpha_k = 0, \\ N-1, & \text{else if } \pi_k^{\mathrm{d}} = \tau | \pi_k^{\mathrm{a}} = N, \alpha_k = N, \\ \pi_k^{d}, & \text{else if } \pi_k^{\mathrm{d}} = \pi_k^{\mathrm{a}}, \\ \underline{\alpha}, & \text{else if } \pi_k^{\mathrm{d}} = \underline{\alpha} | \pi_k^{\mathrm{a}} = \alpha_k, \\ \overline{\alpha}, & \text{else if} \pi_k^{\mathrm{a}} = \overline{\alpha} | \pi_k^{\mathrm{d}} = \alpha_k \}, \\ \alpha_k, & \text{otherwise.} \end{cases} \tag{18}$$

We characterize the NE strategies and saddle-point values of the dual deter model under the assumption of a scalar linear dynamical system (12) and quadratic costs (13) with a parameterized saddle-point value (14). Such an action space leads to a reduced dimension of the cost-to-go matrix independent of the state term x^2 at any node $\alpha_k \in \{1, 2, \ldots, N-1\}$, given by:

$$\begin{array}{c} \\ \alpha_k \\ \underline{\alpha} \end{array} \begin{array}{c} \begin{array}{cc} \alpha_k & \overline{\alpha} \end{array} \\ \begin{bmatrix} \left(f_k^{\alpha_k}\right)^2 \mathbf{p}_{k+1}^{\alpha_k} & \left(f_k^{\overline{\alpha}}\right)^2 \mathbf{p}_{k+1}^{\overline{\alpha}} - \mathbf{a}_k^{\alpha_k} \\ \left(f_k^{\underline{\alpha}}\right)^2 \mathbf{p}_{k+1}^{\underline{\alpha}} + \mathbf{d}_k^{\alpha_k} & \left(f_k^{\alpha_k}\right)^2 \mathbf{p}_{k+1}^{\alpha_k} + \mathbf{d}_k^{\alpha_k} - \mathbf{a}_k^{\alpha_k} \end{bmatrix} \end{array}. \tag{19}$$

Similarly, the cost-to-go matrix for the start node $\alpha_k = 0$ independent of the state term x^2 is given by:

$$\begin{array}{c} \\ 0 \\ \tau \end{array} \begin{array}{c} \begin{array}{cc} 0 & \tau \end{array} \\ \begin{bmatrix} \left(f_k^{0}\right)^2 \mathbf{p}_{k+1}^{0} & \left(f_k^{1}\right)^2 \mathbf{p}_{k+1}^{1} - \mathbf{a}_k^{0} \\ \left(f_k^{0}\right)^2 \mathbf{p}_{k+1}^{0} + \mathbf{d}_k^{0} & \left(f_k^{0}\right)^2 \mathbf{p}_{k+1}^{0} + \mathbf{d}_k^{0} - \mathbf{a}_k^{0} \end{bmatrix} \end{array}, \tag{20}$$

whereas for the end node $\alpha_k = N$, we have:

$$\begin{array}{c} \\ N \\ \tau \end{array} \begin{array}{c} \begin{array}{cc} N & \tau \end{array} \\ \begin{bmatrix} \left(f_k^{N}\right)^2 \mathbf{p}_{k+1}^{N} & \left(f_k^{N}\right)^2 \mathbf{p}_{k+1}^{N} - \mathbf{a}_k^{N} \\ \left(f_k^{N-1}\right)^2 \mathbf{p}_{k+1}^{N-1} + \mathbf{d}_k^{N} & \left(f_k^{N}\right)^2 \mathbf{p}_{k+1}^{N} + \mathbf{d}_k^{N} - \mathbf{a}_k^{N} \end{bmatrix} \end{array}. \tag{21}$$

The transition of the nodes in (19) follows from the `FlipDyn` dynamics (2). Next, we present the NE takeover in both pure and mixed strategies of both the players along with the saddle-point value parameter $\mathbf{p}_k^{\alpha_k}$ for every node in the dual deter model.

Theorem 1. *The unique NE takeover strategies of the* ***FlipDyn-G*** *game* (7) *at any time* $k \in \mathcal{K}$ *for quadratic state and takeover costs* (13)*, subject to the* ***FlipDyn*** *dynamics* (18) *and scalar state dynamics* (12) *are given by:*

Case i) - $\alpha_k = 0$

$$y_k^{0*} = \begin{cases} \left[\dfrac{\mathbf{a}_k^0}{\hat{\mathbf{p}}_{k+1}} \; 1 - \dfrac{\mathbf{a}_k^0}{\hat{\mathbf{p}}_{k+1}}\right]^{\mathrm{T}}, & \text{if} \quad \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^0, \hat{\mathbf{p}}_{k+1} > \mathbf{d}_k^0, \\ \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathrm{T}}, & \text{otherwise,} \end{cases} \tag{22}$$

$$z_k^{0*} = \begin{cases} \left[1 - \dfrac{\mathbf{d}_k^0}{\hat{\mathbf{p}}_{k+1}} \; \dfrac{\mathbf{d}_k^0}{\hat{\mathbf{p}}_{k+1}}\right]^{\mathrm{T}}, & \text{if} \quad \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^0, \hat{\mathbf{p}}_{k+1} > \mathbf{d}_k^0, \\ \begin{bmatrix} 0 & 1 \end{bmatrix}^{\mathrm{T}}, & \text{if} \quad \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^0, \hat{\mathbf{p}}_{k+1} \leq \mathbf{d}_k^0, \\ \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathrm{T}}, & \text{otherwise,} \end{cases} \tag{23}$$

and the saddle-point value parameter satisfies:

$$\mathbf{p}_k^0 = \begin{cases} \mathbf{g}_k^0 + (f_k^0)^2 \mathbf{p}_{k+1}^0 + \mathbf{d}_k^0 - \dfrac{\mathbf{a}_k^0 \mathbf{d}_k^0}{\hat{\mathbf{p}}_{k+1}}, & \text{if} \quad \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^0, \hat{\mathbf{p}}_{k+1} > \mathbf{d}_k^0, \\ \mathbf{g}_k^0 + (f_k^1)^2 \mathbf{p}_{k+1}^1 - \mathbf{a}_k^0, & \text{if} \quad \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^0, \hat{\mathbf{p}}_{k+1} \leq \mathbf{d}_k^0, \\ \mathbf{g}_k^0 + (f_k^0)^2 \mathbf{p}_{k+1}^0, & \text{otherwise,} \end{cases} \tag{24}$$

where $\hat{\mathbf{p}}_{k+1} := (f_k^1)^2 \mathbf{p}_{k+1}^1 - (f_k^0)^2 \mathbf{p}_{k+1}^0$.

Case ii) - $\alpha_k = \{1, 2, \ldots, N-1\}$

$$y_k^{\alpha_k *} = \begin{cases} \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathrm{T}}, & \text{if} \quad \tilde{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{d}_k^{\alpha_k}, -\check{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{d}_k^{\alpha_k}, \\ \begin{bmatrix} 0 & 1 \end{bmatrix}^{\mathrm{T}}, & \text{else if } \tilde{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{d}_k^{\alpha_k}, -\check{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{d}_k^{\alpha_k}, \\ \left[\dfrac{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} - \mathbf{a}_k^{\alpha_k}}{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \check{\mathbf{p}}_{k+1}^{\alpha_k}} \; \dfrac{\check{\mathbf{p}}_{k+1}^{\alpha_k} + \mathbf{a}_k^{\alpha_k}}{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \check{\mathbf{p}}_{k+1}^{\alpha_k}}\right]^{\mathrm{T}}, & \text{otherwise} \end{cases} \tag{25}$$

$$z_k^{0*} = \begin{cases} \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathrm{T}}, & \text{if} \quad -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{a}_k^{\alpha_k}, \check{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{a}_k^{\alpha_k}, \\ \begin{bmatrix} 0 & 1 \end{bmatrix}^{\mathrm{T}}, & \text{if} \quad -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{a}_k^{\alpha_k}, \check{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{a}_k^{\alpha_k}, \\ \left[\dfrac{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \mathbf{d}_{k+1}^{\alpha_k}}{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \check{\mathbf{p}}_{k+1}^{\alpha_k}} \; \dfrac{\check{\mathbf{p}}_{k+1}^{\alpha_k} - \mathbf{d}_{k+1}^{\alpha_k}}{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \check{\mathbf{p}}_{k+1}^{\alpha_k}}\right]^{\mathrm{T}}, & \text{otherwise,} \end{cases} \tag{26}$$

and the saddle-point value parameter satisfies:

$$\mathbf{p}_k^{\alpha_k} = \begin{cases} \mathbf{g}_k^{\alpha_k} + (f_k^{\alpha_k})^2 \mathbf{p}_{k+1}^{\alpha_K}, & \text{if } \begin{array}{l} -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{a}_k^{\alpha_k}, \check{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{a}_{k+1}^{\alpha_k}, \\ \check{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{d}_{k+1}^{\alpha_k}, \end{array} \\ \mathbf{g}_k^{\alpha_k} + (f_k^{\overline{\alpha}})^2 \mathbf{p}_{k+1}^{\overline{\alpha}} + \mathbf{d}_{k+1}^{\alpha_k}, & \text{if } \begin{array}{l} -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{a}_k^{\alpha_k}, \check{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{a}_{k+1}^{\alpha_k}, \\ \check{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{d}_{k+1}^{\alpha_k}, \end{array} \\ \mathbf{g}_k^{\alpha_k} + (f_k^{\overline{\alpha}})^2 \mathbf{p}_{k+1}^{\overline{\alpha}} - \mathbf{a}_{k+1}^{\alpha_k}, & \text{if } \begin{array}{l} -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{a}_k^{\alpha_k}, \check{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{a}_{k+1}^{\alpha_k}, \\ -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} < \mathbf{d}_{k+1}^{\alpha_k}, \end{array} \\ \mathbf{g}_k^{\alpha_k} + (f_k^{\alpha_k})^2 \mathbf{p}_{k+1}^{\alpha_k} - \mathbf{a}_{k+1}^{\alpha_k} + \mathbf{d}_{k+1}^{\alpha_k}, & \text{if } \begin{array}{l} -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{a}_k^{\alpha_k}, \check{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{a}_{k+1}^{\alpha_k}, \\ -\tilde{\mathbf{p}}_{k+1}^{\alpha_k} > \mathbf{d}_{k+1}^{\alpha_k}, \end{array} \\ \mathbf{g}_k^0 + \dfrac{(f_k^{\alpha_k})^4 (\mathbf{p}_{k+1}^{\alpha_k})^2 + \mathbf{a}_k^{\alpha_k} \mathbf{d}_k^{\alpha_k}}{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \check{\mathbf{p}}_{k+1}^{\alpha_k}} & \\ + \dfrac{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} \mathbf{d}_k^{\alpha_k} - \check{\mathbf{p}}_{k+1}^{\alpha_k} \mathbf{a}_k^{\alpha_k}}{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \check{\mathbf{p}}_{k+1}^{\alpha_k}} & \text{otherwise,} \\ - \dfrac{(f_k^{\overline{\alpha}})^2 \mathbf{p}_{k+1}^{\overline{\alpha}} (f_k^{\overline{\alpha}})^2 \mathbf{p}_{k+1}^{\overline{\alpha}}}{\tilde{\mathbf{p}}_{k+1}^{\alpha_k} + \check{\mathbf{p}}_{k+1}^{\alpha_k}}, & \end{cases} \tag{27}$$

where

$$\tilde{\mathbf{p}}_{k+1}^{\alpha_k} := (f_k^{\alpha_k})^2 \mathbf{p}_{k+1}^{\alpha_k} - (f_k^{\overline{\alpha}})^2 \mathbf{p}_{k+1}^{\overline{\alpha}}, \check{\mathbf{p}}_{k+1}^{\alpha_k} := (f_k^{\alpha_k})^2 \mathbf{p}_{k+1}^{\alpha_k} - (f_k^{\overline{\alpha}})^2 \mathbf{p}_{k+1}^{\overline{\alpha}}.$$

Case iii) - $\alpha_k = N$

$$y_k^{N*} = \begin{cases} \begin{bmatrix} 1 - \dfrac{\mathbf{a}_k^N}{\bar{\mathbf{p}}_{k+1}} & \dfrac{\mathbf{a}_k^N}{\bar{\mathbf{p}}_{k+1}} \end{bmatrix}^{\mathrm{T}}, & \text{if } \bar{\mathbf{p}}_{k+1} > \mathbf{a}_k^N, \bar{\mathbf{p}}_{k+1} > \mathbf{d}_k^N, \\ \begin{bmatrix} 0 & 1 \end{bmatrix}^{\mathrm{T}}, & \text{if } \bar{\mathbf{p}}_{k+1} \le \mathbf{a}_k^N, \bar{\mathbf{p}}_{k+1} > \mathbf{d}_k^N, \\ \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathrm{T}}, & \text{otherwise,} \end{cases} \tag{28}$$

$$z_k^{N*} = \begin{cases} \begin{bmatrix} \dfrac{\mathbf{d}_k^N}{\bar{\mathbf{p}}_{k+1}} & 1 - \dfrac{\mathbf{d}_k^N}{\bar{\mathbf{p}}_{k+1}} \end{bmatrix}^{\mathrm{T}}, & \text{if } \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^N, \hat{\mathbf{p}}_{k+1} > \mathbf{d}_k^N, \\ \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathrm{T}}, & \text{otherwise,} \end{cases} \tag{29}$$

and the saddle-point value parameter is given by:

$$\mathbf{p}_k^N = \begin{cases} \mathbf{g}_k^N + (f_k^N)^2 \mathbf{p}_{k+1}^N - \mathbf{d}_k^N + \dfrac{\mathbf{a}_k^0 \mathbf{d}_k^0}{\hat{\mathbf{p}}_{k+1}}, & \text{if } \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^N, \hat{\mathbf{p}}_{k+1} > \mathbf{d}_k^N, \\ \mathbf{g}_k^N + (f_k^{N-1})^2 \mathbf{p}_{k+1}^{N-1} + \mathbf{d}_k^N, & \text{if } \hat{\mathbf{p}}_{k+1} > \mathbf{a}_k^N, \hat{\mathbf{p}}_{k+1} \le \mathbf{d}_k^N, \\ \mathbf{g}_k^N + (f_k^N)^2 \mathbf{p}_{k+1}^N, & \text{otherwise,} \end{cases} \tag{30}$$

where $\bar{\mathbf{p}}_{k+1} := (f_k^N)^2 \mathbf{p}_{k+1}^N - (f_k^{N-1})^2 \mathbf{p}_{k+1}^{N-1}$.

The boundary condition of the saddle-point value recursion (24), (27), (30) *at* $k = L+1$ *is given by:*

$$\mathbf{p}_{L+1}^{\alpha_{L+1}} := \mathbf{g}_{L+1}^{\alpha_{L+1}}, \forall \alpha_{L+1} \in V. \tag{31}$$

□

The derivation of the NE takeover policies and saddle-point value parameters in Theorem 1 closely follows the procedure outlined in [31]. Therefore, we omit the proofs for the sake of brevity. Theorem 1 presents a closed-form solution for the `FlipDyn-G` (7) game with NE takeover strategies independent of state for scalar linear dynamical systems. The dual deter model captures a specific structure of the general `FlipDyn-G` game. This structure enables us to complete the NE strategies and saddle-point value of the game in closed-form. The following remark indicates when the dual deter model maps to the `FlipDyn` model [10].

Remark 1. When the dual deter model consists of only two nodes, $\alpha = \{0, 1\}$, the `FlipDyn-G` game reduces to a `FlipDyn` [10] model with a full state feedback control, with NE strategy and saddle-point value parameter as described in (22), (23), (24), (28), (29), and (30).

Next, we illustrate the results of Lemma 2 through two numerical examples.

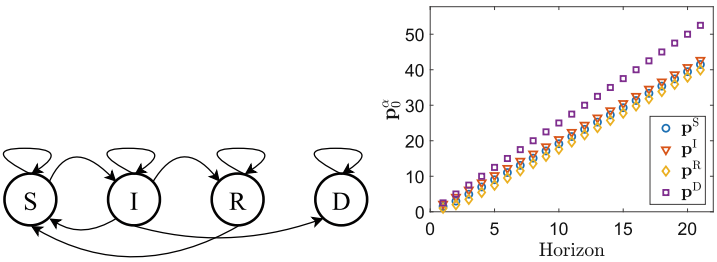
5 Numerical Examples

5.1 Numerical Example I

We evaluate the NE takeover strategy and saddle-point value of the `FlipDyn-G` game on an epidemic dynamic model, which is a discrete-time linear model capturing the dynamics of infection. This model can be mapped to a graph environment with four nodes: susceptible, infected, recovered, and deceased, termed as the SIRD model. The adversary is assumed to be the source of infection causing transitions between nodes, while a government organization represents the defender preventing transitions that can lead to significant losses. Typically, epidemic models have fixed transition probabilities between nodes; however, in this setup, transitions are governed by NE takeover policies. The SIRD model is shown in Fig. 3a, with four `FlipDyn` states: susceptible (S), infected (I), recovered (R), and deceased (D). Therefore, the `FlipDyn` state can take on the value $\alpha_k \in \{\mathrm{S}, \mathrm{I}, \mathrm{R}, \mathrm{D}\}$ for all $k \in \mathcal{K}$.

This example presents only a `FlipDyn` dynamics, as the nodes do not have an underlying continuous state dynamics. In this example, we will consider the costs to be time-invariant, i.e., $\mathbf{g}_k^\alpha = \mathbf{g}^\alpha, \mathbf{d}_k^\alpha = \mathbf{d}^\alpha$, and $\mathbf{a}_k^\alpha = \mathbf{a}^\alpha, \forall k \in \mathcal{K}$ and $\alpha \in \{\mathrm{S,I,R,D}\}$. The state costs follow the order given by:

$$\mathbf{g}^{\mathrm{D}} > \mathbf{g}^{\mathrm{I}} > \mathbf{g}^{\mathrm{S}} > \mathbf{g}^{\mathrm{R}}. \tag{32}$$



(a) (b)

Fig. 3. (a) An epidemic model represented as a graph with four nodes. The `FlipDyn` states of the graph are susceptible (S), Infected (I), Recovered (R), and Deceased (D). (b) Saddle-point parameters for each node $\alpha = \{\text{S,I,R,D}\}$, over time k, with horizon length $L = 20$.

The state costs (32) imply that the `FlipDyn` state of death ($\alpha = \text{D}$) has the highest cost, while the least is for the recovered ($\alpha = \text{R}$). Similarly, the defender and adversary takeover costs follow the order given by:

$$\mathbf{d}^{\text{R}} > \mathbf{d}^{\text{S}} > \mathbf{d}^{\text{I}} > \mathbf{d}^{\text{D}}, \qquad \mathbf{a}^{\text{D}} > \mathbf{a}^{\text{I}} > \mathbf{a}^{\text{S}} > \mathbf{a}^{\text{R}}. \tag{33}$$

The costs used in this numerical example are:

$$\begin{aligned} &\mathbf{g}^{\text{S}} = 1.5,\ \mathbf{g}^{\text{I}} = 2.2,\ \mathbf{g}^{\text{R}} = 1.0,\ \mathbf{g}^{\text{D}} = 2.5,\\ &\mathbf{d}^{\text{S}} = 0.7,\ \mathbf{d}^{\text{I}} = 0.5,\ \mathbf{d}^{\text{R}} = 0.8,\ \mathbf{d}^{\text{D}} = 0.2,\\ &\mathbf{a}^{\text{S}} = 0.5,\ \mathbf{a}^{\text{I}} = 0.7,\ \mathbf{a}^{\text{R}} = 0.1,\ \mathbf{a}^{\text{D}} = 0.9. \end{aligned}$$

We solve for the NE takeover strategies and saddle-point value using Lemma 2. Figure 3b shows the saddle-point value parameters $\mathbf{p}^{\alpha}_{k}, \alpha = \{\text{S,I,R,D}\}$ for a horizon length of $L = 20$. The saddle-point values corresponding to each of the nodes follow the order described in (32) indicating the cost in transitioning to the state $\alpha = D$ is the highest. We also observe that the value of the node $\alpha = I$ remains close to the other node states $\alpha = \{\text{R,D}\}$ reflective of the defender policy to prevent transition to $\alpha = \text{D}$.

The defender and adversary policies for the state $\alpha = \text{I}$ are shown in Figs. 4a and 4b. The state $\alpha = \text{D}$ is a sink state, meaning once you transition to it, you cannot transition to other states. We illustrate the policy for the state $\alpha = \text{I}$ as it allows both players to transition to any state. The defender's policy involves transitioning only to the susceptible and recovered states, avoiding the death state or remaining in the infected state. In contrast, the adversary has a high probability of transitioning to the death state and a low probability of transitioning to the recovered state, with zero probability of transitioning to the susceptible and infected states.

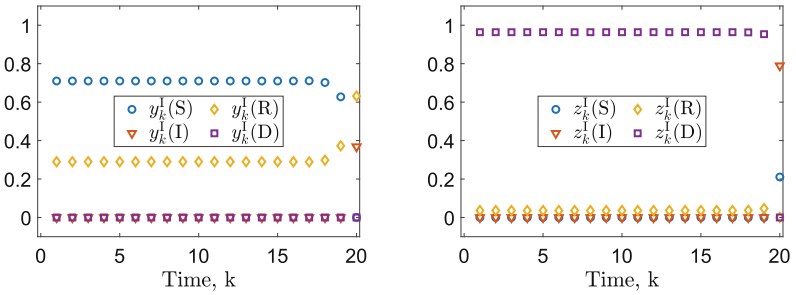


Fig. 4. For the node $\alpha = \mathrm{I}$, the NE policy of the (a) defender and (b) adversary, where $y_k^{\mathrm{I}}(\alpha), z_k^{\mathrm{I}}(\alpha), \alpha = \{\mathrm{S,I,R,D}\}$ corresponds to the probability of selecting the takeover node α, given $\alpha_k = \mathrm{I}$.

5.2 Numerical Example II

We evaluate the NE takeover strategy and saddle-point value of the `FlipDyn-G` game on a stock market Markov chain [34], with node dynamics. This model consists of three nodes: bull market, bear market, and stagnant market. An investor is represented as an adversary attempting to capitalize on the market, while the defender represents the rest of the players in the market. A bull, bear, and stagnant market represent an increase, decrease, and steady market growth, respectively. A graphical representation of this stock market model is shown in Fig. 5a, with three `FlipDyn` states: bull (Bu), bear (Br), and stagnant (St). Therefore, the `FlipDyn` state can take on the value $\alpha_k \in \{\mathrm{Bu}, \mathrm{Br}, \mathrm{St}\}$ for all $k \in \mathcal{K}$. For this example, we will assume the costs and dynamics are time-invariant, i.e., $\mathbf{g}_k^\alpha = \mathbf{g}^\alpha, \mathbf{d}_k^\alpha = \mathbf{d}^\alpha$, and $\mathbf{a}_k^\alpha = \mathbf{a}^\alpha, f_k^{\alpha_k} = f^{\alpha_k}, \forall k \in \mathcal{K}$ and $\alpha \in \{\mathrm{Bu,Br,St}\}$. The state costs and node dynamics follow the order given by:

$$\mathbf{g}^{\mathrm{Bu}} > \mathbf{g}^{\mathrm{Br}} > \mathbf{g}^{\mathrm{St}}, \qquad f^{\mathrm{Bu}} > f^{\mathrm{Br}} > f^{\mathrm{St}}. \tag{34}$$

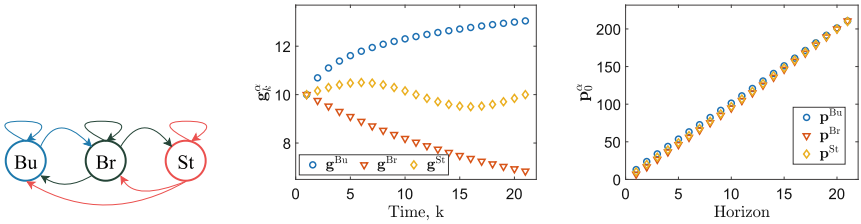
The state costs and dynamics (34) indicate the `FlipDyn` state of the bull market ($\alpha = \mathrm{Bu}$) has the highest value with the least being the stagnant market ($\alpha = \mathrm{St}$). Similarly, the defender and adversary takeover costs follow the order:

$$\mathbf{d}^{\mathrm{Bu}} > \mathbf{d}^{\mathrm{St}} > \mathbf{d}^{\mathrm{Br}}, \qquad \mathbf{a}^{\mathrm{Br}} > \mathbf{a}^{\mathrm{St}} > \mathbf{a}^{\mathrm{Bu}}. \tag{35}$$

The dynamics and takeover costs used in this numerical example are:

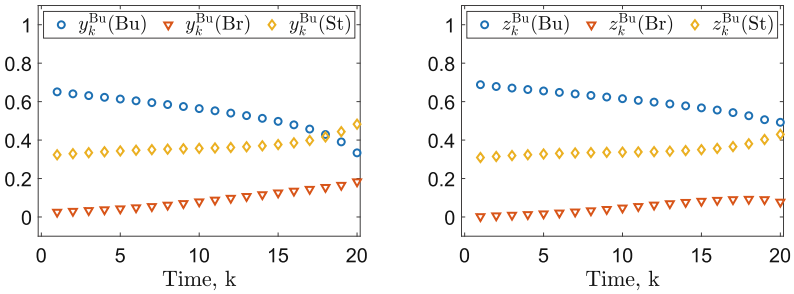
$$\mathbf{f}^{\mathrm{Bu}} = 1.1,\ \mathbf{f}^{\mathrm{Br}} = 0.95,\ \mathbf{g}^{\mathrm{St}} = 1.0,$$
$$\mathbf{d}^{\mathrm{Bu}} = \mathbf{a}^{\mathrm{Br}} = 0.90,\ \mathbf{d}^{\mathrm{Br}} = \mathbf{a}^{\mathrm{Bu}} = 0.50,\ \mathbf{d}^{\mathrm{St}} = \mathbf{a}^{\mathrm{St}} = 0.75.$$

The `FlipDyn` state costs are time-varying and indicate in Fig. 5b. We solve for the NE takeover strategies and saddle-point value using Lemma 2. Figure 5c shows the saddle-point value parameters $\mathbf{p}_k^\alpha, \alpha = \{\mathrm{Bu,Br,St}\}$ for a horizon length of



(a) (b) (c)

Fig. 5. (a) A stock market Markov chain model represented as a graph with three nodes. The `FlipDyn` states of the graph are Bull (Bu), Bear (Br), and Stagnant (St). (b) The state costs $\mathbf{g}_k^{\alpha}, \alpha \in \{\mathrm{Bu}, \mathrm{Br}, \mathrm{St}\}$. (c) Saddle-point parameters for each node $\alpha = \{\mathrm{Bu}, \mathrm{Br}, \mathrm{St}\}$, over time k, with horizon length $L = 20$.



(a) (b)

Fig. 6. For the node $\alpha = \mathrm{Br}$, the NE policy of the (c) defender and (d) adversary, where $y_k^{\mathrm{Bu}}(\alpha), z_k^{\mathrm{Bu}}(\alpha), \alpha = \{\mathrm{Bu}, \mathrm{Br}, \mathrm{St}\}$ corresponds to the probability of selecting the takeover node α, given $\alpha_k = \mathrm{Bu}$.

$L = 20$. At the start of the horizon, the difference between the saddle-point values follows the order (34). However, as the horizon increases, the differences between saddle-point values of the `FlipDyn` states become indistinguishable.

We only illustrate the defender and adversary policy for the state $\alpha = \mathrm{Bu}$ shown in Figs. 6a and 6b, respectively. The policy trends of both players are quite similar, with a high probability of being in the bull market, followed by the stagnant market and bear market. The investor (adversary) indicates a higher probability of being in the bull market and maintains this probability throughout the time horizon. In contrast, the defender exhibits a relatively lower probability of being in the bull state, with the highest probability gradually shifting to transitioning to the stagnant state over time.

This numerical example illustrates the use of the `FlipDyn` model in graphs to determine node takeover strategies for each player. It provides insights into system behavior and stability, which are useful for designing costs that impact takeover policies.

6 Conclusion

In this paper, we have introduced the FlipDyn-G framework, extending the FlipDyn model to a graph-based setting where each node represents a dynamical system. Our model captures the strategic interactions between a defender and an adversary who aim to control node state in a graph to minimize and maximize a finite horizon sum cost, respectively.

Our contributions include modeling and characterizing the FlipDyn-G game for general dynamical systems and deriving the corresponding Nash Equilibrium (NE) takeover strategies. Additionally, for scalar linear discrete-time dynamical systems with quadratic costs, we derived NE takeover strategies and saddle-point values that are independent of the continuous state of the system. For a finite state birth-death Markov chain, we derived analytical expressions for these NE strategies and values. Through numerical studies involving epidemic models and linear dynamical systems with adversarial interactions, we have illustrated the applicability and effectiveness of our proposed methods. The results demonstrate that our approach can robustly determine optimal strategies for both players, enhancing the resilience and security of cyber-physical systems (CPS).

Future work will focus on extending this framework to more complex topologies and multi-agent systems.

Acknowledgments. This research was supported in part by i) the NSF Award CNS-2134076 under the Secure and Trustworthy Cyberspace (SaTC) program, ii) the NSF CAREER Award ECCS-2236537, and iii) AFOSR FA9550-21-1-0411.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Lee, E.A.: Cyber physical systems: design challenges. In: 11th IEEE international Symposium on Object and Component-oriented Real-time Distributed Computing (ISORC). IEEE 2008, pp. 363–369 (2008)
2. Shi, J., Wan, J., Yan, H., Suo, H.: A survey of cyber-physical systems. In: International Conference on Wireless Communications and Signal Processing (WCSP). IEEE 2011, pp. 1–6 (2011)
3. Han, Y., Zhang, K., Li, H., Coelho, E.A.A., Guerrero, J.M.: Mas-based distributed coordinated control and optimization in microgrid and microgrid clusters: a comprehensive overview. IEEE Trans. Power Electron. **33**(8), 6488–6508 (2017)
4. Zheng, Y., Li, S.E., Li, K., Wang, L.-Y.: Stability margin improvement of vehicular platoon considering undirected topology and asymmetric control. IEEE Trans. Control Syst. Technol. **24**(4), 1253–1265 (2015)
5. Zheng, Y., Li, S.E., Wang, J., Cao, D., Li, K.: Stability and scalability of homogeneous vehicular platoon: study on the influence of information flow topologies. IEEE Trans. Intell. Transp. Syst. **17**(1), 14–26 (2015)
6. Li, S.E., et al.: Dynamical modeling and distributed control of connected and automated vehicles: challenges and opportunities. IEEE Intell. Transp. Syst. Mag. **9**(3), 46–58 (2017)

7. Bullo, F., Cortés, J., Martinez, S.: Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms, vol. 27. Princeton University Press, Princeton (2009)
8. Olfati-Saber, R., Fax, J.A., Murray, R.M.: Consensus and cooperation in networked multi-agent systems. Proc. IEEE **95**(1), 215–233 (2007)
9. Van Dijk, M., Juels, A., Oprea, A., Rivest, R.L.: Flipit: the game of "stealthy takeover". J. Cryptol. **26**, 655–713 (2013)
10. Banik, S., Bopardikar, S.D.: Flipdyn: a game of resource takeovers in dynamical systems. In: 2022 IEEE 61st Conference on Decision and Control (CDC), pp. 2506–2511 (2022)
11. Bowers, K.D., et al.: Defending against the unknown enemy: applying FlipIt to system security. In: Grossklags, J., Walrand, J. (eds.) Decision and Game Theory for Security. GameSec 2012. LNCS, vol. 7638, pp. 248–263. Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34266-0_15
12. Johnson, B., Laszka, A., Grossklags, J.: Games of timing for security in dynamic environments. In: Khouzani, M.H.R., Panaousis, E., Theodorakopoulos, G. (eds.) GameSec 2015. LNCS, vol. 9406, pp. 57–73. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25594-1_4
13. Merlevede, J., Johnson, B., Grossklags, J., Holvoet, T.: Time-dependent strategies in games of timing. In: Alpcan, T., Vorobeychik, Y., Baras, J.S., Dán, G. (eds.) GameSec 2019. LNCS, vol. 11836, pp. 310–330. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32430-8_19
14. Laszka, A., Horvath, G., Felegyhazi, M., Buttyan, L.: FlipThem: modeling targeted attacks with flipit for multiple resources. In: Poovendran, R., Saad, W. (eds.) Decision and Game Theory for Security. GameSec 2014. LNCS, vol. 8840, pp. 175–194. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12601-2_10
15. Leslie, D., Sherfield, C., Smart, N.P.: Threshold FlipThem: when the winner does not need to take all. In: Khouzani, M.H.R., Panaousis, E., Theodorakopoulos, G. (eds.) GameSec 2015. LNCS, vol. 9406, pp. 74–92. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25594-1_5
16. Zhang, M., Zheng, Z., Shroff, N.B.: Defending against stealthy attacks on multiple nodes with limited resources: a game-theoretic analysis. IEEE Trans. Control Netw. Syst. **7**(4), 1665–1677 (2020)
17. Leslie, D., Sherfield, C., Smart, N.P.: Multi-rate threshold flipthem. In: Foley, S.N., Gollmann, D., Snekkenes, E. (eds.) ESORICS 2017. LNCS, vol. 10493, pp. 174–190. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66399-9_10
18. Yao, Q., Xiong, X., Wang, Y.: Cheat-FlipIt: an approach to modeling and perception of a deceptive opponent. In: Hermanns, H., Sun, J., Bu, L. (eds.) Dependable Software Engineering. Theories, Tools, and Applications. SETTA 2023. LNCS, vol. 14464, pp. 368–384. Springer, Singapore (2024). https://doi.org/10.1007/978-981-99-8664-4_21
19. Daian, P., et al.: Flash boys 2.0: frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In: IEEE Symposium on Security and Privacy (SP). IEEE 2020, pp. 910–927 (2020)
20. Ding, J., Kamgarpour, M., Summers, S., Abate, A., Lygeros, J., Tomlin, C.: A stochastic games framework for verification and control of discrete time stochastic hybrid systems. Automatica **49**(9), 2665–2674 (2013)
21. Kontouras, E., Tzes, A., Dritsas, L.: Adversary control strategies for discrete-time systems. In: European Control Conference (ECC). IEEE 2014, pp. 2508–2513 (2014)

22. Kontouras, E., Tzes, A., Dritsas, L.: Covert attack on a discrete-time system with limited use of the available disruption resources. In: European Control Conference (ECC). IEEE 2015, pp. 812–817 (2015)
23. Smith, R.S.: Covert misappropriation of networked control systems: presenting a feedback structure. IEEE Control Syst. Mag. **35**(1), 82–92 (2015)
24. Mohan, A.M., Meskin, N., Mehrjerdi, H.: Covert attack in load frequency control of power systems. In: 6th IEEE International Energy Conference (ENERGYCon). IEEE 2020, pp. 802–807 (2020)
25. Saha, S., Vullikanti, A., Halappanavar, M.: Flipnet: modeling covert and persistent attacks on networked resources. In: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2017, pp. 2444–2451 (2017)
26. Acquaviva, J., Mahon, M., Einfalt, B., LaPorta, T.: Optimal cyber-defense strategies for advanced persistent threats: a game theoretical analysis. In: IEEE 36th Symposium on Reliable Distributed Systems (SRDS). IEEE 2017, pp. 204–213 (2017)
27. Moothedath, S., et al.: A game-theoretic approach for dynamic information flow tracking to detect multistage advanced persistent threats. IEEE Trans. Autom. Control **65**(12), 5248–5263 (2020)
28. Rass, S., König, S., Panaousis, E.: Cut-the-rope: a game of stealthy intrusion. In: Alpcan, T., Vorobeychik, Y., Baras, J.S., Dán, G. (eds.) GameSec 2019. LNCS, vol. 11836, pp. 404–416. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32430-8_24
29. Rass, S., König, S., Wachter, J., Mayoral-Vilches, V., Panaousis, E.: Game-theoretic apt defense: an experimental study on robotics. Comput. Secur. **132**, 103328 (2023)
30. Miura, H., Kimura, T., Hirata, K.: Modeling of malware diffusion with the flipit game. In: 2020 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan). IEEE, 2020, pp. 1–2 (2020)
31. Banik, S., Bopardikar, S.D.: Flipdyn with control: resource takeover games with dynamics, arXiv preprint arXiv:2310.14484, 2023
32. Hespanha, J.P.: Noncooperative Game Theory: An Introduction for Engineers and Computer Scientists. Princeton University Press, Princeton (2017)
33. Li, S.-Q.: Overload control in a finite message storage buffer. IEEE Trans. Commun. **37**(12), 1330–1338 (1989)
34. Peovski, F., Cvetkoska, V., Trpeski, P., Ivanovski, I.: Monitoring stock market returns: a stochastic approach. Croa. Oper. Res. Rev. **13**(1), 65–76 (2022)