This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3269323

Edge Selections in Bilinear Dynamic Networks

Arthur Castello B. de Oliveira[1], Milad Siami[1], and Eduardo D. Sontag [1,2]

***Abstract*—We develop some basic principles for the design and robustness analysis of a continuous-time bilinear dynamical network, where an attacker can manipulate the strength of the interconnections/edges between some of the agents/nodes. We formulate the edge protection optimization problem of picking a limited number of attack-free edges and minimizing the impact of the attack over the bilinear dynamical network. In particular, the $\mathcal{H}_2$-norm of bilinear systems is known to capture robustness and performance properties analogous to its linear counterpart and provides valuable insights for identifying which edges are most sensitive to attacks. The exact optimization problem is combinatorial in the number of edges, and brute-force approaches show poor scalability. However, we show that the $\mathcal{H}_2$-norm as a cost function is supermodular and, therefore, allows for efficient greedy approximations of the optimal solution. We illustrate and compare the effectiveness of our theoretical findings via numerical simulations.**

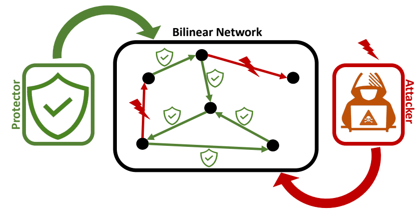

Fig. 1: This illustration shows a bilinear network with 6 nodes and 7 edges. The nodes represent "agents" and the edges show the "flow" between the agents. The presented network has an edge protection budget of 5. The designer has chosen to protect the 5 edges depicted in green, leaving the ones in red vulnerable to attacks.

I. Introduction

The robust design of control systems against adversarial attacks is crucial for sustainability, from engineering infrastructures to living cells. One way to think of this problem is to consider a set of interacting dynamic agents, whose behaviors are influenced by the flows between them. These flows can involve either information or physical objects, such as electrical currents in power grids, social mobility in networks of interconnected populations, and transmission of infections, among others [1]–[6]. In this context, we consider a scenario where each agent has a state or quantity that evolves over time based on its own previous values and on the flow of information from other nearby agents. These interactions are represented by a graph, where the agents are depicted as nodes and the flow of information is represented by edges, as shown in Fig. 1. To ensure the robust synthesis of systems, the design may need to be strengthened to mitigate the effects of interference by adversaries or unexpected failures that could disrupt progress towards the desired goal state. In this context, a relevant question is how to optimize the design to reduce the network's vulnerability and make the system safer.

In the classical linear dynamic network literature, the actions available to an adversary are restricted to directly affecting the dynamics of specific agents, in the hopes (of the adversary) that these disruptions will propagate to the rest of the network. This formulation is still very rich and versatile, even with such a limited assumption. Many works can be found in the literature, [7]–[18], where the authors investigate how disturbances on specific agents propagate and how to evaluate the weak spots of a network. Despite the extensive literature on this subject, the assumption that interference only impacts node dynamics restricts the analysis to local conclusions in situations where the adversary holds a greater level of power over the network. On the other hand, analyzing the network while allowing for arbitrary levels of control by the adversary is a difficult problem, so in this paper, we take a middle-ground approach. We examine the scenario where the adversary has the ability to not only interfere with individual nodes but also disturb the capacity of the connections between them in an "affine" manner, which means our network has bilinear terms in its dynamics.

While the tools and insights from the study of linear systems do not directly apply to bilinear systems, they are still a fascinating class of nonlinear systems that have been studied extensively in the literature (e.g. see references [1], [2], [19]–[22]). Bilinear systems have the ability to approximate a wide range of functions, and have been used to model problems in a diverse range of fields, including electrical and transportation networks, surface vehicles, and immunology. Moreover, bilinear systems have been applied in various ways in Artificial intelligence (AI), including in the modeling and analysis of neural networks, optimization and control of complex systems, natural language processing tasks, and image recognition [2], [4]–[6], [23]. These systems can improve the robustness and efficiency of machine learning algorithms, leading to enhanced performance in tasks such as pattern recognition, decision making, and control.

In this paper, we describe an optimization problem for safeguarding a network with vulnerable nodes and edges. To reduce the impact of attacks on the network, the system designer tries to determine the best combination of edges to safeguard, based on the nodes being targeted and the available resources for protecting a certain number of edges. We show that, when it is well-defined, our proposed $\mathcal{H}_2$-based performance metric is supermodular on the power set of vulnerable edges, enabling the use of approximation algorithms with guaranteed performance. The first contribution of this paper is to clearly formulate the optimization problem. We then proceed to examine the problem using algorithmic approaches.

II. Preliminary Definitions

A. Notations and Assumptions

Throughout this paper, the set of real numbers and non-negative real numbers are represented by $\mathbb{R}$ and $\mathbb{R}_+$, respectively. Similarly, the set of the strictly positive integers is denoted by $\mathbb{N}$ and the set of strictly positive integers up to m by $\mathbb{N}_{\leq m}$. For any finite set of elements $\mathcal{V}$, let $|\mathcal{V}| \in \mathbb{N} \cup \{0\}$ be the number of elements in the set, with $|\mathcal{V}| = 0 \iff \mathcal{V} = \emptyset$, and let $2^{\mathcal{V}}$ be the power set of $\mathcal{V}$. Furthermore, for any function f with domain in $\mathcal{V}$, for each subset $\bar{\mathcal{V}} \subset \mathcal{V}$ define $f(\bar{\mathcal{V}}) = \{f(v) \mid v \in \bar{\mathcal{V}}\}$. The elementary vector of index i is denoted e_i and is a vector of all elements zero except for the i-th element, which is 1. Similarly, an elementary matrix

This work was partially supported by AFOSR Grant FA9550-21-1-0289 (EDS), NSF 2208182 (MS), NSF 2121121 (ACBO, MS), and ONR Grant N00014-21-1-2431 (ACBO, MS, EDS).

[1]Department of Electrical & Computer Engineering, Northeastern University, Boston, MA 02115 USA (e-mails: {castello.a, m.siami, e.sontag}@northeastern.edu).

[2]Departments of Bioengineering, Northeastern University, Boston, MA 02115 USA.

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3269323

$E_{ij} = e_i e_j^\top$ has all its elements zero except for the one in position (i, j), which is one. In some of the derivations in this paper, we use matrix algebra techniques involving vectorization and the Kronecker product (denoted by $\otimes$), which can be found in [24].

B. Bilinear Dynamical Networks

Consider a linear (di)graph given by the triplet $\mathcal{G} = (\mathcal{V},\ \mathcal{E},\ w)$, where $\mathcal{V} \subset \mathbb{N}$, $\mathcal{E} \subseteq \mathcal{V}\times\mathcal{V}$, and $w : \mathcal{E} \to \mathbb{R}$, together with sets $\mathcal{E}_a \subseteq \mathcal{E}$ and $\mathcal{V}_a \subset \mathcal{V}$ defined as the set of edges and nodes (respectively) under the effect of external disturbances. The set $\mathcal{V}$ is composed of agents with linear dynamics given by

$$\Sigma_i : \dot{x}_i(t) = -d_i x_i(t) + \sum_{(i,j)\in\mathcal{E}} w(j, i)x_j(t) + \sum_{(i,j)\in\mathcal{E}_a} \eta_{ij}(t)x_j + \delta_i v_i(t), \tag{1}$$

where, for every $i \in \mathcal{V}$, $d_i \in \mathbb{R}_+$ and $\delta_i = 1$ if $i \in \mathcal{V}_a$ and zero otherwise.

Let $n{=}|\mathcal{V}|$, $m_v{=}|\mathcal{V}_a|$, and $m_e{=}|\mathcal{E}_a|$ be the number of nodes, nodes under attack, and edges under attack in the network respectively. Then we can write the dynamics of the entire network Σ as follows

$$\Sigma : \dot{x}(t) = \left(N_0 + \sum_{k=1}^{m_\mathrm{e}} \eta_k(t)N_k\right) x(t) + Bv(t) \tag{2}$$

where $x \in \mathbb{R}^n$ is the state vector, $N_0 = -D + A$ is assumed to be Hurwitz, $D = \mathrm{diag}([d_1, \ldots , d_n])$ and A is the adjacency matrix of $\mathcal{G}$. For every $i \in \mathcal{V}_a$, vector v is the concatenation of the additive (node) disturbances v_i, and $B \in \mathbb{R}^{n\times m_\mathrm{v}}$ is the column composition of elementary vectors $e_i \in \mathbb{R}^n$, both in the same order. If the graph is directed $N_k{=}E_{j_k i_k}{=}e_{j_k} e_{i_k}^\top$, and $\eta_k{=}\eta_{j_k i_k}$ is called a multiplicative (edge) disturbance, with (i_k, j_k) being the k-th edge in $\mathcal{E}_a$, according to some arbitrary ordering. If the graph is undirected then all the above holds except that $N_k = E_{j_k i_k} + E_{i_k j_k} = e_{j_k} e_{i_k}^\top + e_{i_k} e_{j_k}^\top$. The set $\mathcal{G}_b = \{\mathcal{V}, \mathcal{E}, w, \mathcal{V}_a, \mathcal{E}_a\}$ and its associated dynamics (2) are called a bilinear (di)graph or network.

Notice that the dynamics of a bilinear dynamical network is a particular case of the generic bilinear system given by

$$\begin{cases} \dot{x}(t) = \left(N_0 + \sum\limits_{k=1}^{m} u_k(t)\bar{N}_k\right) x(t) + \bar{B}u(t) \\ y(t) = Cx(t) \end{cases} \tag{3}$$

where $m = m_\mathrm{v} + m_\mathrm{e}$, $u = [\eta^\top, v^\top]^\top$, $\bar{N}_k = N_k$ for $1 \le k \le m_\mathrm{e}$ and $\bar{N}_k = 0_{n\times n}$ for $k > m_\mathrm{e}$, $\bar{B} = [0_{n\times m_\mathrm{e}}, B]$, and $C = I_{n\times n}$. As a particular case, any results from the bilinear systems literature are immediately applicable to bilinear dynamical networks.

III. $\mathcal{H}_2$-Based Performance Measure and its Properties

A. $\mathcal{H}_2$-norm of bilinear systems

The Volterra series is routinely used to obtain solutions for bilinear systems, with many results associating N_0 being Hurwitz with the convergence of the series and the stability of the system [3], [25]–[28]. The i-th order Volterra kernel of a bilinear system is given by

$$\begin{aligned} h_i(t, \tau_1, \ldots \tau_i) = &\sum_{k_2,\ldots,k_i=1}^{m} \mathrm{e}^{N_0(t-\tau_i)} N_{k_i} \mathrm{e}^{N_0(\tau_i-\tau_{i-1})} \\ &\times N_{k_{i-1}} \mathrm{e}^{N_0(\tau_{i-1}-\tau_{i-2})} \cdots N_{k_2} \mathrm{e}^{N_0(\tau_2-\tau_1)} B \end{aligned} \tag{4}$$

and the $\mathcal{H}_2$-norm is defined as a function of the multivariable Laplace transform of the volterra kernels as below.

Definition 1. *Assuming zero initial condition, that N_0 is Hurwitz, and letting the i-th order transfer function $H_i(s_1, s_2, \ldots , s_i)$ be the multivariable Laplace transform of the i-th Volterra kernel $h_i(t_1, t_2, \ldots , t_i)$, the $\mathcal{H}_2$-norm of a bilinear system Σ is defined as*

$$\begin{aligned} \|\Sigma\|_{\mathcal{H}_2} = &\Bigg(\sum_{i=1}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \mathrm{trace}(H_i^\top(jw_1, \ldots , jw_i) \\ &\times H_i(jw_1, \ldots , jw_i))dw_1 \ldots dw_i \Bigg)^{1/2}. \end{aligned} \tag{5}$$

With this definition, the $\mathcal{H}_2$-norm of bilinear systems is known to satisfy some of the same properties as its linear counterpart. For example, from [29] we can write the value of the $\mathcal{H}_2$-norm of (3) as a function of the reachability Gramian as follows:

$$\|\Sigma\|_2 = \sqrt{\mathrm{trace}(CPC^\top)}, \tag{6}$$

where P is the reachability Gramian of the bilinear system defined as

$$P = \sum_{q=1}^{\infty} \int_0^{\infty} \cdots \int_0^{\infty} \ P_q P_q^\top \ dt_1 \cdots dt_q, \tag{7}$$

where

$$P_q P_q^\top = \mathrm{e}^{N_0 t_q} \sum_{k\in\mathcal{E}_a} N_k P_{q-1} P_{q-1}^\top N_k^\top \mathrm{e}^{N_0^\top t_q}, \tag{8}$$

for $q > 1$, and

$$P_1 P_1^\top = \mathrm{e}^{N_0 t_1} BB^\top \mathrm{e}^{N_0^\top t_1}. \tag{9}$$

Furthermore, if (7) converges then P is the solution of the generalized Lyapunov equation

$$N_0 P + P N_0^\top + \sum_{k\in\mathcal{E}_a} N_k P N_k^\top + BB^\top = 0, \tag{10}$$

and is positive semi-definite.

The expression of (6) with (10) allows a more efficient computation of the $\mathcal{H}_2$-norm than its definition in (5), enabling its use in optimization problems. These, however, assume the convergence of the infinite sums in (5) and (7). In [29] the authors state the following assumption as sufficient for that:

Assumption 1. *The matrix N_0 is stable and for two numbers α and β, which satisfy the inequality $\|e^{N_0 t}\| \le \beta e^{-\alpha t}$ for all $t > 0$, we have $\sqrt{\sum_{k\in\mathcal{E}_a} \|N_k N_k^\top\|} < \sqrt{2\alpha}/\beta$.*

This assumption requires that the linear dynamics dominates over the worst-case bilinear dynamics. For bilinear networks, it sets a limit on the number of multiplicative disturbances or, alternatively, the maximum "energy" of each disturbance. As we have noted in previous works ([30], [31]), this assumption can be overly restrictive, as we will demonstrate in Section V.

To find a less conservative assumption, we can consider a bilinear system written as in equation (3) and assume that all inputs $\bar{u}$ are independently sampled white noise signals $\bar{u} = \eta = dW/dt$, with unitary covariance. This leads to a stochastic differential equation, which has been studied extensively in the literature [32]–[34]. Based on this literature, we can make the following assumption.

Assumption 2. *For a bilinear system as (3) with independently sampled, unitary covariance Gaussians as its inputs, N_0 is Hurwitz and the following holds:*

$$\lambda_{\max}\left(I \otimes N_0 + N_0 \otimes I + \sum_{k=1}^{m} N_k \otimes N_k\right) < 0 \tag{11}$$

Under Assumption 2, the solution to the bilinear SDE satisfies $\lim_{t\to\infty} \mathbb{E}(x(t)) = 0$ and $\lim_{t\to\infty} \mathrm{Cov}(x(t)) = P$, where P is the solution to the generalized Lyapunov equation (10).

The link between Assumption 2 and deterministic bilinear systems

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3269323

is shown in [20] where the author shows that Assumption 2 also guarantees the convergence of the reachability Gramian. In the same paper, we are shown the L_2 to $L_{\inf}$ relation of the $\mathcal{H}_2$ norm of bilinear systems through the following equation:

$$\sup_{t\geq 0}\|y(t)\|_2 \leq \left(\operatorname{trace}(CPC^\top)\right)^{\frac{1}{2}} \exp\left\{0.5\|u^0\|^2_{\mathcal{L}^2}\right\}\|u\|_{\mathcal{L}^2}, \quad (12)$$

where P, the reachability Gramian, is the unique solution to

$$N_0P + PN_0^\top + \sum_{k=1}^{m}\bar{N}_kP\bar{N}_k^\top = -\bar{B}\bar{B}^\top. \quad (13)$$

The vector u^0 is defined in [20] as $u_k^0(t) = u_k(t)$ if $N_k \neq 0$ and $u_k^0(t) = 0$ otherwise, that is u^0 is nonzero only for the inputs that affect the system in a bilinear way.

The first thing to notice from the result above is that it recovers the linear definition of the $\mathcal{H}_2$-norm. In general, for a bilinear system (12), it still establishes an L_2 to L_∞ relationship between input and output, although not as directly as an induced norm. In fact, a remark in [20] shows that as long as N_0 is Hurwitz, one can restrict the inputs to a small enough ball around 0 such that (2) holds, which is exactly the same condition proven in [35] as necessary and sufficient for a bilinear system to be iISS, a established condition for L_2 to L_∞ stability of nonlinear systems.

When comparing the two assumptions, we can show that Assumption 1 implies Assumption 2, but not the converse. To see that, take the L_2-norm as follows

$$\begin{aligned}\|\dot{P}\| &= \|N_0P + PN_0^\top + \sum_k N_kPN_k^\top + BB^\top\| \\ &\leq 2\|N_0P\| + \|\sum_k N_kPN_k^\top\| + \|BB^\top\| \\ &\leq 2\sigma_{max}(N_0)\|P\| + \|\sum_k N_kN_k^\top\|\|P\| + \|BB^\top\|,\end{aligned} \quad (14)$$

which is a monovariable deterministic ODE with stable solution if and only if $\|\sum N_kN_k^\top\| \leq 2\|\sigma_{max}(N_0)\|$. Therefore, if that upper bound on the norm of P is stable, then P converges.

The literature on stochastic bilinear systems establishes a direct relationship between the $\mathcal{H}_2$-norm and the covariance of the output under white noise inputs. Furthermore, results from the literature on deterministic bilinear systems have also shown that the bilinear $\mathcal{H}2$-norm reflects the relationship between the L_2-norm of the inputs and the L_∞-norm of the output. This emphasizes the usefulness of the $\mathcal{H}_2$-norm as a performance metric for bilinear systems. In the following section, we examine how the $\mathcal{H}_2$-norm can be used to solve the edge selection problem efficiently.

B. Supermodularity of the $\mathcal{H}_2$-norm

For the main theoretical result of this paper, consider the following definition

Definition 2. *Define a* family of bilinear digraphs $\mathcal{F}$ *generated by the ground set of m_v vulnerable edges $\mathcal{E}_v \subseteq \mathcal{V}\times\mathcal{V}$ as follows:*

$$\mathcal{F}(\mathcal{E}_v) := \left\{\mathcal{G} = (\mathcal{V},\mathcal{E},w,\mathcal{E}_a,\mathcal{V}_a)\middle|\ \mathcal{E}_a \in 2^{\mathcal{E}_v}\right\},$$

for given node set $\mathcal{V}$, edge set $\mathcal{E}$, weight function w, and attacked node set $\mathcal{V}_a$. We assume $\Sigma(\mathcal{E}_a)$ is the bilinear system (2) *induced by the corresponding bilinear digraph $(\mathcal{V},\mathcal{E},w,\mathcal{E}_a,\mathcal{V}_a) \in \mathcal{F}(\mathcal{E}_v)$. We can, then, define the square of the $\mathcal{H}_2$ norm as a set function $\rho_\Sigma(.) : 2^{\mathcal{E}_v} \to \mathbb{R}_+$ as*

$$\rho_\Sigma(\mathcal{E}_a) \ := \ \|\Sigma(\mathcal{E}_a)\|^2_{\mathcal{H}_2}, \quad \forall \mathcal{E}_a \in 2^{\mathcal{E}_v}. \quad (15)$$

In the following theorem, we characterize some functional properties of set function $\rho_\Sigma(.) : 2^{\mathcal{E}_v} \to \mathbb{R}_+$.

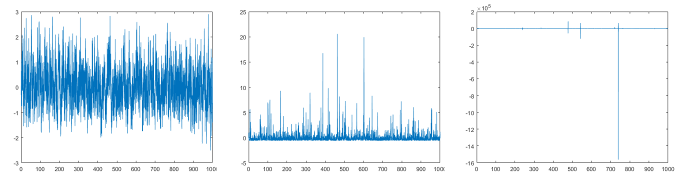

Fig. 2: This figure shows how changing the value of the bilinear term k in the single-input single-output (SISO) bilinear system (16) affects the evolution of state x over time. The system is subjected to Gaussian white noise disturbances η and v, with constants $a = 1$ and $b = 1$. The subplots from left to right correspond to k values of 0.1, $\sqrt{2}$, and 10.

Theorem 1. *Suppose that for a family of bilinear digraphs $\mathcal{F}$ the $\mathcal{H}_2$-norm is properly defined for everyone of its elements, then the square of the $\mathcal{H}_2$ norm defined as a set function $\rho_\Sigma(\mathcal{E}_a) : 2^{\mathcal{E}_v} \to \mathbb{R}_+$ is monotone and supermodular.*

Notice that the assumption made on the theorem (proper definition of the $\mathcal{H}_2$-norm) is satisfied if all elements of $\mathcal{F}$ satisfy either Assumption 1 or 2. To investigate the tightness and relationship between the two assumptions, and the behaviour of the system when they are broken, consider the following system:

$$\dot{x} = -ax + kx\eta + bv \quad (16)$$

where η and v are independently sampled white noise inputs and a, b and k are positive nonzero constants.

The generalized Lyapunov equation given by (10) can be solved for this system by $P = \frac{b^2}{2a-k^2}$. Assumption 1 can be written as $a > 0$ and $|k| < \sqrt{2a}$, or $2a - k^2 > 0$, and Assumption 2 becomes $-2a + k^2 < 0$. Looking at the formulation for the Gramian from equation (7), we can write $\bar{P}_1 = \int_0^\infty e^{-a\tau}bbe^{-a\tau}d\tau = \frac{b^2}{2a}$, $\bar{P}_2 = \int_0^\infty e^{-a\tau}k\bar{P}_1ke^{-a\tau}d\tau = \frac{b^2}{2a}\frac{k^2}{2a}$, and $\bar{P}_i = \int_0^\infty e^{-a\tau}k\bar{P}_{i-1}ke^{-a\tau}d\tau = \frac{b^2}{2a}\left(\frac{k^2}{2a}\right)^{i-1}$, with $P = \sum_{i=1}^\infty \bar{P}_i = \sum_{i=1}^\infty \frac{b^2}{2a}\left(\frac{k^2}{2a}\right)^{i-1}$.

This defines the infinite sum of a geometric progression with quotient $q = k^2/2a$ and initial value $a_0 = b^2/2a$. The necessary and sufficient convergence condition for the sum is $k^2/2a < 1 \iff 2a - k^2 > 0$ which coincides with Assumptions 1 and 2. This means that for this SISO bilinear system, both assumptions coincide and are necessary and sufficient for any positive values of k, a and b. Simulating this system with $a = 1$, $b = 1$ and different values of k allows us to observe how breaking the convergence condition affects the behaviour of the system.

The first simulation, shown in the left plot of Fig. 2 with $k = 0.1$, appears to satisfy both assumptions for this system. The simulation was done ten times and the averaged covariance is 0.5051, very closely lower bounded by the Gramian $P = 0.5025$. We actually observed some of the samples with a covariance bellow 0.5025, indicating the tightness of the bound for this system.

The second simulation, shown in the center plot of Fig. 2 with $k = \sqrt{2}$, is at the boundary of the assumptions made for the system. The estimated covariance of the state diverges, and we can see that the system exhibits high amplitude peaks, indicating a change in behavior. The covariance of the system also varied between different runs of the simulation, but it consistently resulted in high, sporadic peaks. This is accentuated in the third simulation, shown in the right plot of Fig. 2 with $k = 10$, where the peaks are more evident.

It is important to note that this system is iISS (with a quadratic integrand gain function), which means that its deterministic response to L_2 inputs can never be unstable, regardless of the value of k. The

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3269323

instability in the covariance of the state appears in the form of high peaks, which become more pronounced as k increases in relation to a and b.

IV. Approximation Algorithms

We now focus on the network synthesis problem. Our goal is to improve the performance of a bilinear network (2) by removing $\ell \geq 1$ edges from the vulnerable edge set $\mathcal{E}_v$. Specifically, we aim to find a subset of ℓ vulnerable edges $\mathcal{E}_p \subset \mathcal{E}_v$, with $|\mathcal{E}_p| \leq \ell$ that when protected minimizes the $\mathcal{H}_2$-norm of the system.

A. Edge Protection Problem Formulation

As we have assumed that an attacker will always attempt to compromise as many edges as possible, we must protect enough edges to ensure that either Assumption 1 or 2 holds. This leads to the following combinatorial optimization problem for edge protection:

$$\begin{aligned} \mathcal{E}_* = \arg \min_{\mathcal{E}_a \subset \mathcal{E}_v} \quad & \rho_\Sigma(\mathcal{E}_a) \\ \text{s.t.} \quad & |\mathcal{E}_a| \geq |\mathcal{E}_v| - \ell, \end{aligned} \tag{17}$$

where $\mathcal{E}_v$ is the set of vulnerable edges, and ℓ is the maximum number of edges that can be protected. The optimal protected edge set can be obtained by $\mathcal{E}_p = \mathcal{E}_v \backslash \mathcal{E}_*$ from (17).

In our optimization problem (17), we aim to find the member of the family of bilinear networks with the highest $\mathcal{H}_2$-norm, as measured by the function ρ_Σ, while also satisfying the constraints. According to Definition 2, ρ_Σ is used to evaluate the $\mathcal{H}_2$-norm of different members of the family of bilinear networks. Solving combinatorial optimization problems can be challenging, but we should note that for $\ell = 1$, the exact solution can be found by calculating the value of $\rho_\Sigma(\mathcal{E}_a)$ for all n sets of attack edges with $n-1$ elements. However, for larger values of ℓ, the number of sets with $n - \ell$ elements grows almost exponentially, especially when ℓ is close to $n/2$. This can be seen by the expression $\binom{n}{\lceil n/2 \rceil} \sim \frac{2^n}{\sqrt{n}}$. As a result, using a straightforward approach to solve the optimization problem may not be computationally efficient.

B. Linearized Edge Selection

One intuitive approach is to linearize the cost function $\rho_\Sigma(.)$ at some operating point. To do this, we can consider a relaxed cost function ρ_Σ defined as follows:

$$\begin{aligned} \rho_\Sigma(c_k) &= \text{trace}(P(c_k)) = \sum_{i=1}^n e_i^\top P(c_k) e_i = \sum_{i=1}^n e_i^\top \otimes e_i^\top \text{vec}(P(c_k)) \\ &= \sum_{i=1}^n e_i^\top \otimes e_i^\top (A \otimes I + I \otimes A + \sum_{k=1}^{m_e} c_k^2 N_k \otimes N_k)^{-1} \\ &\times \text{vec}(BB^\top), \end{aligned} \tag{18}$$

where $c \in \mathbb{R}^n$, $c = [c_1, c_2, \ldots, c_{m_e}]^\top$, with $0 \leq c_k \leq 1$, for $k = 1, 2, \ldots, m_e$. It is easy to verify that such relaxed function is the extention of the set function ρ_Σ to the polyhedron with vertices at the original domain set. Defining $W(c)$ as

$$W(c) = A \otimes I + I \otimes A + \sum_{k=1}^{m_e} c_k^2 N_k \otimes N_k, \tag{19}$$

Algorithm 1: A linearized-based algorithm to select protected edges.

Input : Σ, $\mathcal{E}_v$, and ℓ
Output: $\mathcal{E}_*$

```
ℰ_a ← {}
for k = 1 to m_e do
    rho(k) ← compute ∂ρ_Σ/∂c_k for c̄ = [1, 1, ..., 1]^⊤
end
i ← indexes of ℓ highest elements of rho
ℰ_* ← ℰ_v(i)
return ℰ_*
```

Algorithm 2: A greedy heuristic to sequentially pick protected edges.

Input : Σ, $\mathcal{E}_v$, and ℓ
Output: $\mathcal{E}_*$

```
ℰ_a ← {}
for k = 1 to ℓ do
    {e} ← find an edge in ℰ_v that returns the minimum value for
              ρ_Σ(ℰ_a ∪ {e}) − ρ_Σ(ℰ_a)
    ℰ_a ← ℰ_a ∪ {e}
    ℰ_v ← ℰ_v \ {e}
end
ℰ_* ← ℰ_a
return ℰ_*
```

we take the partial derivative with respect to c_k at some operation point $\bar{c}$ as

$$\begin{aligned} \frac{\partial \rho_\Sigma}{\partial c_k}(\bar{c}) &= \sum_{i=1}^n (e_i^\top \otimes e_i^\top) \frac{\partial (W(\bar{c}))^{-1}}{\partial c_k} \text{vec}(BB^\top) \\ &= \sum_{i=1}^n (e_i^\top \otimes e_i^\top) W(\bar{c})^{-1} \frac{\partial W}{\partial c_k}(\bar{c}) W(\bar{c})^{-1} \text{vec}(BB^\top), \end{aligned} \tag{20}$$

where

$$\frac{\partial W}{\partial c_k}(\bar{c}) = 2 c_k N_k \otimes N_k. \tag{21}$$

To select the ℓ edges to protect, we evaluate (20) at $\bar{c} = [1, \ 1, \ldots, \ 1]^\top$ for all $k \in \mathbb{N}_{\leq m_e}$, and choose the ℓ highest values. The algorithm 1 is used to implement this approach in the simulations.

C. Greedy Edge Selection

A second approach is to use a greedy algorithm hat takes advantage of our ability to solve the problem for $\ell = 1$ and the existence of theoretical bounds for the greedy minimization of supermodular functions subject to cardinality constraints. The greedy algorithm 2 is used in the simulations.

It is known that for the maximization of submodular functions is NP-hard and the greedy algorithm does not deliver the optimal solution in general. However optimality gaps are given in the literature allowing the leverage of efficient algorithms to find a good suboptimal solution. In [36], the authors present a greedy-based algorithm with complexity $\mathcal{O}(n\ell)$ and that guarantees that the resulting solution is at least 35.6% of the maximum value of the function. However, while it is reassuring to know the optimality gap is bounded, 35.6% is a large margin for suboptimality. To evaluate a better margin, in the

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3269323

next section we relax the combinatorial optimization to obtain a closer lower bound for the optimal solution of this problem.

D. A Lower Bound for the Optimal Cost

Next we compute a lower bound for the optimal solution of our problem. With this we can evaluate an upper bound for the optimality gap and check if the greedy solution is good.

To do this, we first relax the integer problem of selecting a subset $\mathcal{E}_a$ of a set of vulnerable edges $\mathcal{E}_v$, as in (17), to find a vector $c \in \mathbb{R}^m$, where $m = |\mathcal{E}_v|$, $c = [c_1, c_2, \dots, c_m]^\top$, as below.

$$\begin{aligned} \min_{P,c} \quad & \mathrm{trace}(P) \\ \text{s.t.} \quad & N_0 P + P N_0^\top + \sum_{k=1}^{m} c_k^2 N_k P N_k + BB^\top = 0, \\ & \sum_k c_k \geq m_v - \ell, \\ & 0 \leq c_k \leq 1, \ \forall k = 1, \dots, m. \end{aligned} \tag{22}$$

Notice that the first constraint is quadratic and bilinear in the parameters. To solve the first problem, we can redefine $\bar{c} = [c_1^2, \ c_2^2, \dots, c_m^2]^\top$. To solve the second problem, first notice that we can solve the generalized Lyapunov equation for P by using the Kronecker product as

$$\mathbf{vec}(P) = -W(\bar{c})^{-1}\mathbf{vec}(BB^\top), \tag{23}$$

where

$$W(\bar{c}) = N_0 \otimes I + I \otimes N_0 + \sum_k \bar{c}_k N_k \otimes N_k. \tag{24}$$

Next, we point that for $N_0 \in \mathbb{R}^{n\times n}$,

$$\mathrm{trace}(N_0) = \sum_{k=1}^{n} e_i^\top N_0 e_i = \sum_{k=1}^{n} (e_i^\top \otimes e_i^\top)\mathbf{vec}(N_0), \tag{25}$$

where e_i is the i-th vector in the canonical base of the vector space. Using (23) and (25), we can rewrite the relaxed problem (22) as

$$\begin{aligned} \min_{\bar{c}} \quad & -\sum_{k=1}^{n} (e_k^\top \otimes e_k^\top) W(\bar{c})^{-1}\mathbf{vec}(BB^\top) \\ \text{s.t.} \quad & \sum_k \bar{c}_k \geq m_v - \ell, \ \ 0 \leq \bar{c}_k \leq 1, \ \forall k = 0, \dots, m \end{aligned} \tag{26}$$

The conversion of the constraint $\sum_k c_k \geq m_v - \ell$ to $\sum_k c_k^2 \geq m_v - \ell$ does not result in the same feasibility set, but it is easy to verify that $\sum_k c_k^2 \geq m_v - \ell \to \sum_k c_k \geq m_v - \ell$ if $0 \leq c_k \leq 1$. As such a solution to (26) is a lower bound to the solution of (22), which is a lower bound to the solution of the original combinatorial optimization problem.

The final step for being able to efficiently solve our lower bound problem is to show that its cost function is convex. Consider the functions $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathcal{S}_{n^2}^+ \to \mathbb{R}_+$ and $h : \mathbb{R}^n \to \mathcal{S}_{n^2}^+$ defined as below

$$h(\bar{c}) = -W(\bar{c}) = -(N_0 \otimes I + I \otimes N_0 + \sum_k \bar{c}_k N_k \otimes N_k), \tag{27}$$

$$g(W) = \sum_{k=1}^{n} (e_k^\top \otimes e_k^\top) W^{-1}\mathbf{vec}(BB^\top) \tag{28}$$

$$f(\bar{c}) = g(h(\bar{c})). \tag{29}$$

One can easily verify that f defined as above is the cost function of our relaxed problem, and that h is affine on its arguments. One can also conclude that any $\bar{c}$ in the domain of definition of the $\mathcal{H}_2$ norm results in a positive definite value for $h(\bar{c})$ (since $W(\bar{c})$ needs to be negative definite for Assumption 1 to hold). For showing convexity of g consider the following lemma:

Lemma 2. *Function $g(.) : \mathcal{S}_n^+ \to \mathbb{R}_+$ given by (28) is convex.*

Proof. Function g is convex if and only if for any $\lambda \in [0, 1]$

$$g(\lambda W_1 + (1-\lambda)W_2) \leq \lambda g(W_1) + (1-\lambda) g(W_2). \tag{30}$$

Defining $\bar{g}(W) = W^{-1}$ we rewrite the inequality above as

$$\begin{aligned} &\sum_{k=1}^{n} (e_k^\top \otimes e_k^\top) \bar{g}(\lambda W_1 + (1-\lambda)W_2)\mathbf{vec}(BB^\top) \\ \leq &\sum_{k=1}^{n} (e_k^\top \otimes e_k^\top)(\lambda \bar{g}(W_1) + (1-\lambda)\bar{g}(W_2))\mathbf{vec}(BB^\top) \end{aligned} \tag{31}$$

which allow us to conclude that g is convex if $\bar{g}$ is convex in the positive definite sense. To show convexity of $\bar{g}$ we need to show that for two positive definite matrices W_1 and W_2,

$$\lambda W_1^{-1} + (1-\lambda)W_2^{-1} \succcurlyeq (\lambda W_1 + (1-\lambda)W_2)^{-1}, \tag{32}$$

which is equivalent to saying that for any $v \in \mathbb{R}^n$

$$\lambda v^\top W_1^{-1} v + (1-\lambda) v^\top W_2^{-1} v \geq v^\top (\lambda W_1 + (1-\lambda)W_2)^{-1} v, \tag{33}$$

Defining $\tilde{g}(\lambda) = u^\top(\lambda W_1 + (1-\lambda)W_2)u$ for any positive definite matrices W_1 and W_2 and nonzero vector u allows us to rewrite the inequality above as.

$$\lambda \tilde{g}(0) + (1-\lambda)\tilde{g}(1) \geq \tilde{g}(\lambda). \tag{34}$$

Define $Z(\lambda) = \lambda W_1 + (1-\lambda)W_2$ and compute the second derivative of $\tilde{g}$ with respect to λ gives

$$\frac{d^2 g}{d\lambda^2} = 2(ZZ^{-1}u)^\top Z^{-1} Z Z^{-1} u = v^\top Z^{-1} v \geq 0 \tag{35}$$

since the inverse of the convex combination of positive definite matrices is positive definite. Therefore, $\tilde{g}(\lambda)$ is convex for any u and any positive definite W_1 and W_2 for $\lambda \in [0, 1]$ which implies $\bar{g}$ is convex and, therefore, g is convex. □

With this we can conclude that $f = g \circ h$ is convex since it is the composition of a convex function with an affine one. Since our cost function is convex and our constraints affine, we solve this relaxed problem for a given number ℓ of attacked edges using a gradient descent algorithm to get a lower bound on the optimal point as well as an argument vector c. By picking the ℓ largest directions of c we can obtain a rounded solution as the third method for solving (17).

V. Simulations

In this section, we demonstrate the effectiveness of our proposed algorithms on two different graphs: a simple one that we can solve using brute force and a more complex one. For each network, we consider a range of budgets from protecting a single edge to protecting all edges. We compare the solutions obtained using each method with the lower bound derived in Subsection IV-D and with the expected gain from randomly selecting the edges to protect. We also assume that every edge of the network is vulnerable, that is $\mathcal{E}_v = \mathcal{E}$, and that a fixed set of nodes (indicated in the diagrams in red) is also under attack. Note that, although we focus only on attacked edges, the interaction between multiplicative and additive disturbances has a great effect over the behavior of the network, as we pointed out in a previous publication [31].

Along this section, the drift matrix N_0 in the bilinear dynamics (2) will be given by

$$N_0 = -L - \frac{1}{n}\mathbb{1}_{n\times n}, \tag{36}$$

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3269323

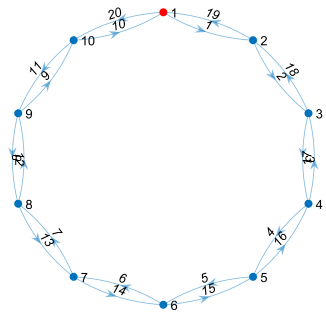

Fig. 3: The directed 10-node ring graph used in Example 1. Node 1, highlighted in red, is the one under the effect of an additive disturbance and all edges are vulnerable.

where L is the Laplacian matrix of the corresponding graph at each section (c.f. Figs. 3, and 5) and $\mathbb{1}$ is the matrix of all ones. For each graph the nodes indicated in red (as node 1 in Fig. 3) are disturbed by an additive attack (i.e., B is the column composition of elementary vectors e_i for all $i \in \mathcal{V}_a$). Finally, in (2), each N_k corresponds to an unprotected vulnerable edge (i, j) where $(i, j) \in \mathcal{E}_v \backslash \mathcal{E}_p$ and defined by $N_k := E_{ij} + E_{ji}$ if the graph is undirected and $N_k := E_{ji}$ otherwise, where E_{ij} is an elementary matrix as defined in Subsection II-A.

Along our simulations, unless specified otherwise, we weight all the N_ks by $\bar{\lambda}_2/m$ to make sure Assumption 1 is satisfied, where $m = |\mathcal{E}_v|$ is the number of vulnerable edges, and $\bar{\lambda}_2 = \min(1, \lambda_2)$ is the smallest between the smallest nonzero eigenvalue of L and 1.

Example 1. *The Ring Graph: In the first simulation we use a 10-node ring digraph labeled as in Fig. 3. Due to its simplicity, we can compute the value of ρ_Σ defined in (15) for any subset of attacked edges and find the actual optimum through brute force. Notice that, despite the symmetry of the ring graph, some edges have a greater effect on the cost function than others. This happens because of the additive disturbance acting on node one, which introduces an important imbalance in the system. We can see from Fig. 4 that for this system, all proposed algorithms obtained the brute force minimum every time.*

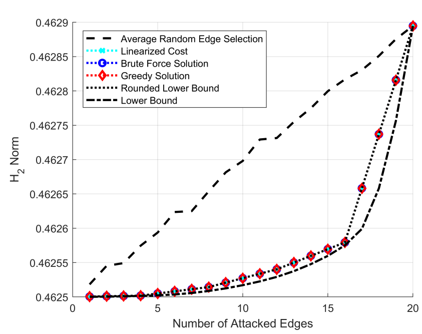

Fig. 4: This figure presents results for bilinear dynamics (2) with topology given by the 10-node ring digraph (Fig. 3). We compare the three proposed methods (linearized cost, greedy algorithm and rounding of lower bound solution) for solving the optimization problem (17) with the actual global minimum obtained through brute force.

We can also see that protecting the four most vulnerable edges results in the greatest reduction in the value of the $\mathcal{H}_2$-norm. This provides valuable information when planning cost-effective protection strategies for a system with the 10-node ring digraph.

Example 2. *The Barábasi-Albert Random Graph: For a more complex randomly generated network, we simulate the 20-node Barabási-Albert graph shown in Fig. 5. Similarly to the previous simulations, our dynamics are given by (2) and the drift matrix by (36), where L is the Laplacian for this graph, and the N_k's are given for each unprotected vulnerable edge.*

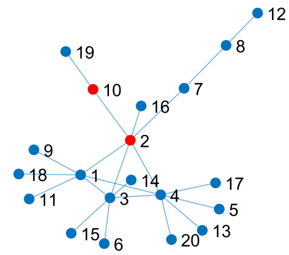

Fig. 5: This figure shows the Barabasi-Albert graph used in Example 2, where nodes 2 and 10 (highlighted in red) are subjected to additive disturbances and all edges are vulnerable.

Due to the larger dimension and increasse complexity of the network when compared to the previous simulations (5 and 10 nodes ring graphs), computing the brute force solution becomes prohibitively time-consuming, therefore the lower bound becomes our reference when analysing suboptimality of our results. We also assume, as with the previous simulations, that every edge is independently disturbed in an undirected manner, unless protected.

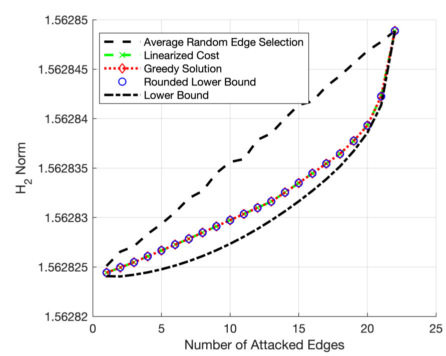

Fig. 6: This figure presents results for bilinear dynamics (2) with topology given by Fig. 5. We compare the three proposed methods (linearized cost, greedy algorithm and rounding of lower bound solution) for solving the optimization problem (17) with the proposed lower bound and the random edge selection. In this case, the bilinear matrices N_k's are weighted to satisfy Assumption 1 and all methods perform similarly.

We can see from Fig. 6 that, as it was with the ring graph simulations, the three proposed algorithms have exactly the same results. This is likely due to the fact that Assumption 1 is too restrictive. By asking that the linear dynamics be dominant in the worst case (all vulnerable edges attacked), we make it so that all three algorithms have the same results, as they would if the dynamics were perfectly linear to begin with. To observe the effects of the bilinear dynamics, we run a second simulation where we do not weight the N_k's to respect Assumption 1, but we verify that it still respects Assumption 2. The results are presented in Fig. 7.

Notice from the second set of simulations that the solutions from our three approximation algorithms differ for a different number of attacked edges, with the greedy edge selection being consistently the best performing algorithm.

VI. Conclusions and Future Works

In this paper, we proposed a way to evaluate the influence of multiplicative disturbances to the overall stability of the system by making

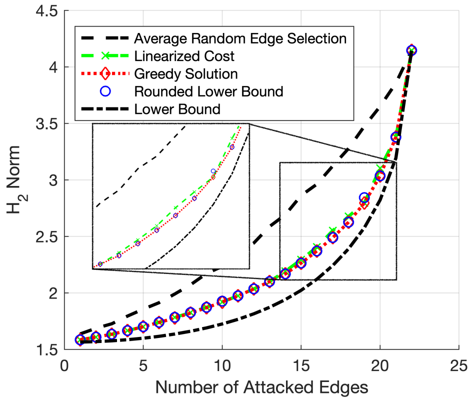

Fig. 7: This figure presents results for bilinear dynamics (2) with topology given by Fig. 5. We compare the three proposed methods (linearized cost, greedy algorithm and rounding of lower bound solution) for solving the optimization problem (17) with the proposed lower bound and the random edge selection. In this simulation, the bilinear matrices N_k's are unweighted but the dynamics still respect Assumption 2.

use of the $\mathcal{H}_2$-norm defined for bilinear systems. We discussed the meanings of the $\mathcal{H}_2$-norm and why it is an interesthing metric, and showed that in the context of edge selection, it is supermodular, which allows us to use efficient selection algorithms with guaranteed known optimality gaps. Important to our $\mathcal{H}_2$-based edge selection method, we discussed how Assumption 1 might be too restrictive by, in practice, requiring the dominance of the linear dynamics over the bilinear one. Furthermore, we discuss a possible relaxation of that assumption, originally derived for Gaussian disturbances, where the $\mathcal{H}_2$-norm still maintains its relationship to the steady-state covariance matrix of the states.

On a more practical note, we proved that the problem of edge selection for maximizing the $\mathcal{H}_2$-norm is supermodular, which gives approximation guarantees for greedy approximations. We also proposed a general lower bound for the optimal solution through the continuous relaxation of the combinatorial optimization. We then simulated two network topologies: one of a simple ring digraph, chosen so that we could compare our greedy solution and the lower bound to the actual minimum; and one of a more complex 20 node Barabási-Albert graph. While all algorithms performed similarly, the greedy solution was the lowest one for all the simulated cases.

We believe that our edge-perturbation results will be of interest in other areas besides classical optimization ones such as transportation networks. In fact, we were largely motivated to pursue this work by the analysis of the impact of edge knock-outs in biological networks carried out in [37], which used CRISPR technology to target microRNA pathways that control processes ranging from cell growth to stress responses. That paper emphasized how edge perturbations play a critical role in cell function, analogously to how node perturbations lead to global behavioral changes through network effects [38].

References

[1] R. R. Mohler, "Natural bilinear control processes," *IEEE Transactions on Systems Science and Cybernetics*, vol. 6, no. 3, pp. 192–197, 1970.

[2] D. Elliot, "Bilinear control systems," *Applied Mathematical Science*, vol. 169, 2009.

[3] E. D. Sontag, *Mathematical control theory: deterministic finite dimensional systems*. Springer Science & Business Media, 2013, vol. 6.

[4] S. Yu, C. Shen, and T. Ersal, "Autonomous driving using linear model predictive control with a koopman operator based bilinear vehicle model," *IFAC-PapersOnLine*, vol. 55, no. 24, pp. 254–259, 2022, 10th IFAC Symposium on Advances in Automotive Control AAC 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896322023254

[5] J.-H. Kim, J. Jun, and B.-T. Zhang, "Bilinear attention networks," *Advances in neural information processing systems*, vol. 31, 2018.

[6] H. Pirsiavash, D. Ramanan, and C. Fowlkes, "Bilinear classifiers for visual recognition," *Advances in neural information processing systems*, vol. 22, 2009.

[7] Y. Shoukry, M. Chong, M. Wakaiki, P. Nuzzo, A. Sangiovanni-Vincentelli, S. A. Seshia, J. P. Hespanha, and P. Tabuada, "SMT-based observer design for cyber-physical systems under sensor attacks," *ACM Transactions on Cyber-Physical Systems*, vol. 2, no. 1, pp. 1–27, 2018.

[8] M. S. Chong, M. Wakaiki, and J. P. Hespanha, "Observability of linear systems under adversarial attacks," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 2439–2444.

[9] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Transactions on Automatic control*, vol. 59, no. 6, pp. 1454–1467, 2014.

[10] Z. Tang, M. Kuijper, M. Chong, I. Mareels, and C. Leckie, "Sensor attack correction for linear systems with known inputs," *IFAC-PapersOnLine*, vol. 51, no. 23, pp. 206–211, 2018.

[11] B. Bamieh, M. Jovanović, P. Mitra, and S. Patterson, "Coherence in large-scale networks: Dimension-dependent limitations of local feedback," *Automatic Control, IEEE Transactions on*, vol. 57, no. 9, pp. 2235 –2249, Sept. 2012.

[12] M. Siami and N. Motee, "Fundamental limits and tradeoffs on disturbance propagation in linear dynamical networks," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 4055–4062, 2016.

[13] A. Jadbabaie and A. Olshevsky., "Combinatorial bounds and scaling laws for noise amplification in networks," in *European Control Conference (ECC)*, July 2013, pp. 596–601.

[14] T. H. Summers, F. L. Cortesi, and J. Lygeros, "On submodularity and controllability in complex dynamical networks," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 1, pp. 91–101, 2015.

[15] T. Summers, I. Shames, J. Lygeros, and F. Dörfler, "Topology design for optimal network coherence," in *2015 European Control Conference (ECC)*. IEEE, 2015, pp. 575–580.

[16] A. Olshevsky, "On (non) supermodularity of average control energy," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1177–1181, 2017.

[17] T. Summers and I. Shames, "Correction to "rigid network design via submodular set function optimization"," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2019.

[18] M. Siami and N. Motee, "Growing linear dynamical networks endowed by spectral systemic performance measures," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 2091–2106, 2017.

[19] Y. Zhao and J. Cortés, "Gramian-based reachability metrics for bilinear networks," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 3, pp. 620–631, 2016.

[20] M. Redmann, "Bilinear systems—a new link to $\mathcal{H}_2$-norms, relations to stochastic systems, and further properties," *SIAM Journal on Control and Optimization*, vol. 59, no. 4, pp. 2477–2497, 2021.

[21] P. Benner and P. Goyal, "Balanced truncation model order reduction for quadratic-bilinear control systems," *arXiv preprint arXiv:1705.00160*, 2017.

[22] Y. Zhao and J. Cortés, "Reachability metrics for bilinear complex networks," in *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 4788–4793.

[23] T. Huang, Z. Zhang, and J. Zhang, "Fibinet: combining feature importance and bilinear feature interaction for click-through rate prediction," in *Proceedings of the 13th ACM Conference on Recommender Systems*, 2019, pp. 169–177.

[24] J. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Transactions on circuits and systems*, vol. 25, no. 9, pp. 772–781, 1978.

[25] M. C. Varona and R. Gebhart, "Impulse response of bilinear systems based on Volterra series representation," *arXiv preprint arXiv:1812.05360*, 2018.

[26] S. Boyd, L. O. Chua, and C. A. Desoer, "Analytical foundations of Volterra series," *IMA Journal of Mathematical Control and Information*, vol. 1, no. 3, pp. 243–282, 1984.

[27] T. Siu and M. Schetzen, "Convergence of Volterra series representation and BIBO stability of bilinear systems," *International Journal of Systems Science*, vol. 22, no. 12, pp. 2679–2684, 1991.

[28] M. Schetzen, *The Volterra and Wiener theories for nonlinear systems*. Krieger Publishing Company, 2006.

[29] L. Zhang and J. Lam, "On $\mathcal{H}_2$ model reduction of bilinear systems," *Automatica*, vol. 38, no. 2, pp. 205–216, 2002.

[30] A. C. B. de Oliveira, M. Siami, and E. D. Sontag, "Edge selection in bilinear dynamical networks," 2020.

[31] A. C. B. De Oliveira, M. Siami, and E. D. Sontag, "Eminence in noisy bilinear networks," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 4835–4840.

[32] T. Damm, "Rational matrix equations in stochastic control," *Lecture Notes in Control and Information Sciences*, vol. 297, 01 2004.

[33] M. B. Priestley, "Non-linear and non-stationary time series analysis," *London: Academic Press*, 1988.

[34] I. Karatzas, I. Karatzas, S. Shreve, and S. E. Shreve, *Brownian motion and stochastic calculus*. Springer Science & Business Media, 1991, vol. 113.

[35] E. D. Sontag, "Comments on integral variants of ISS," *Systems & Control Letters*, vol. 34, no. 1-2, pp. 93–100, 1998.

[36] N. Buchbinder, M. Feldman, J. Naor, and R. Schwartz, "Submodular maximization with cardinality constraints," in *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, 2014, pp. 1433–1452.

[37] Y. Li, C. M. Nowak, D. Withers, A. Pertsemlidis, and L. Bleris, "CRISPR-Based editing reveals edge-specific effects in biological networks," *CRISPR J*, vol. 1, pp. 286–293, 08 2018.

[38] T. Kang, R. Moore, Y. Li, E. Sontag, and L. Bleris, "Discriminating direct and indirect connectivities in biological networks," *Proc Natl Acad Sci USA*, vol. 112, pp. 12 893–12 898, 2015.

This article has been accepted for publication in IEEE Transactions on Automatic Control. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAC.2023.3269323

APPENDIX

To prove Theorem 1, we first prove the following three Lemmas.

Lemma 3. *Let us define $\tilde{P}_q(\mathcal{E}_a, \bar{t}_q) = P_q(\mathcal{E}_a, \bar{t}_q) P_q^\top(\mathcal{E}_a, \bar{t}_q)$, where $\bar{t}_q = [t_1, \ldots, t_q]$ and $P_q P_q^\top$ is given by* (8)*. Then for $q \geq 2$, $\mathcal{A}$, $\mathcal{B} \subset \mathcal{E}_v$, $\mathcal{A} \cap \mathcal{B} = \emptyset$, we have*

$$\tilde{P}_q(\mathcal{A} \cup \mathcal{B}, \bar{t}_q) = \tilde{P}_q(\mathcal{A}, \bar{t}_q) + \tilde{P}_q(\mathcal{B}, \bar{t}_q) + \tilde{R}_q(\mathcal{A}, \mathcal{B}, \bar{t}_q), \tag{37}$$

where $\tilde{R}_q(\mathcal{A}, \mathcal{B}, \bar{t}_q) \succcurlyeq 0$

Proof. We prove the Lemma by induction. First notice that for $q = 1$, $\tilde{P}_1(\mathcal{A} \cup \mathcal{B}, \bar{t}_1) = e^{N_0 t_1} B B^\top e^{N_0^\top t_1}$, is constant and independent from the set of attacked edges, so the relationship from the lemma does not hold, since $\tilde{P}_1(\mathcal{A} \cup \mathcal{B}, \bar{t}_1) = \tilde{P}_1(\mathcal{A}, \bar{t}_1) = \tilde{P}_1(\mathcal{B}, \bar{t}_1) = \tilde{P}_1(\bar{t}_1)$. For $q \geq 2$ we can write

$$\begin{aligned}
&\tilde{P}_2(\mathcal{A} \cup \mathcal{B}, \bar{t}_2) = e^{N_0 t_2} \sum_{k \in \mathcal{A} \cup \mathcal{B}} N_k \tilde{P}_1(\bar{t}_1) N_k^\top e^{N_0^\top t_2} \\
&= e^{N_0 t_2} \sum_{k \in \mathcal{A}} N_k \tilde{P}_1(\bar{t}_1) N_k^\top e^{N_0^\top t_2} + e^{N_0 t_2} \sum_{k \in \mathcal{B}} N_k \tilde{P}_1(\bar{t}_1) N_k^\top e^{N_0^\top t_2} \\
&= \tilde{P}_2(\mathcal{A}, \bar{t}_2) + \tilde{P}_2(\mathcal{B}, \bar{t}_2)
\end{aligned} \tag{38}$$

which proves the base case, since $\tilde{R}_2 = 0$ is positive semidefinite. Furthermore, if we assume the following

$$\begin{aligned}
\tilde{P}_{q-1}(\mathcal{A} \cup \mathcal{B}, \bar{t}_{q-1}) = \tilde{P}_{q-1}(\mathcal{A}, \bar{t}_{q-1}) + \tilde{P}_{q-1}(\mathcal{B}, \bar{t}_{q-1}) \\
+ \tilde{R}_{q-1}(\mathcal{A}, \mathcal{B}, \bar{t}_{q-1})
\end{aligned} \tag{39}$$

with $\tilde{R}_{q-1} \succcurlyeq 0$, then

$$\begin{aligned}
&\tilde{P}_q(\mathcal{A} \cup \mathcal{B}, \bar{t}_q) = e^{N_0 t_q} \left(\sum_{k \in \mathcal{A}} N_k \tilde{P}_{q-1}(\mathcal{A} \cup \mathcal{B}, \bar{t}_{q-1}) N_k^\top \right. \\
&\left. + \sum_{k \in \mathcal{B}} N_k \tilde{P}_{q-1}(\mathcal{A} \cup \mathcal{B}, \bar{t}_{q-1}) N_k^\top \right) e^{N_0^\top t_q} \\
&= \tilde{P}_q(\mathcal{A}, \bar{t}_q) + \tilde{P}_q(\mathcal{B}, \bar{t}_q) + e^{N_0 t_q} \left(\sum_{k \in \mathcal{A}} N_k \tilde{P}_{q-1}(\mathcal{B}, \bar{t}_{q-1}) N_k^\top \right. \\
&\left. + \sum_{k \in \mathcal{B}} N_k \tilde{P}_{q-1}(\mathcal{A}, \bar{t}_{q-1}) N_k^\top + \sum_{k \in \mathcal{A} \cup \mathcal{B}} N_k \tilde{R}_{q-1}(\mathcal{A}, \mathcal{B}, \bar{t}_{q-1}) N_k^\top \right) e^{N_0^\top t_q} \\
&= \tilde{P}_q(\mathcal{A}, \bar{t}_q) + \tilde{P}_q(\mathcal{B}, \bar{t}_q) + R_q(\mathcal{A}, \mathcal{B}, \bar{t}_q)
\end{aligned} \tag{40}$$

To finish the induction step we can verify that R_q is positive semi-definite since it is assumed $R_{q-1} \succcurlyeq 0$, and positive semi-definiteness is invariant under congruence transformations and matrix addition, and since $\tilde{P}_q \succcurlyeq 0$, then $R_q \succcurlyeq 0$. This completes the proof. □

Lemma 4. *Given $\mathcal{A}$, $\mathcal{B} \subset \mathcal{E}_v$, $\mathcal{A} \cap \mathcal{B} = \emptyset$, and P defined by* (7)*, the associated Gramian $P(\mathcal{A} \cup \mathcal{B})$ can be rewritten as follows*

$$P(\mathcal{A} \cup \mathcal{B}) = P(\mathcal{A}) + P(\mathcal{B}) + R(\mathcal{A}, \mathcal{B}) - C \tag{41}$$

where $R(\mathcal{A}, \mathcal{B}) \succcurlyeq 0$, and $C = \int_0^\infty e^{N_0 t} B B^\top e^{N_0^\top t} dt \succcurlyeq 0$ is a constant matrix that is independent of the set of attacked edges.

Proof. Applying Lemma 3 to (7), we get

$$\begin{aligned}
&P(\mathcal{A} \cup \mathcal{B}) = \sum_{q=1}^{\infty} \int_0^\infty \cdots \int_0^\infty \tilde{P}_q(\mathcal{A} \cup \mathcal{B}, \bar{t}_q) dt_1 \ldots dt_q \\
&= \sum_{q=2}^{\infty} \int_0^\infty \cdots \int_0^\infty \left(\tilde{P}_q(\mathcal{A}, \bar{t}_q) + \tilde{P}_q(\mathcal{B}, \bar{t}_q) + \tilde{R}_q(\mathcal{A}, \mathcal{B}, \bar{t}_q) \right) dt_1 \ldots dt_q \\
&\quad + 2 \int_0^\infty \tilde{P}_1(\bar{t}_1) dt_1 - \int_0^\infty \tilde{P}_1(\bar{t}_1) dt_1 \\
&= P(\mathcal{A}) + P(\mathcal{B}) + \underbrace{\sum_{q=2}^{\infty} \int_0^\infty \cdots \int_0^\infty \tilde{R}_q(\mathcal{A}, \mathcal{B}, \bar{t}_q) dt_1 \ldots dt_q}_{R(\mathcal{A}, \mathcal{B})} - \\
&\quad - \int_0^\infty \tilde{P}_1(\bar{t}_1) dt_1.
\end{aligned} \tag{42}$$

The positive semi-definiteness of R is immediate from the fact that such property is invariant under matrix addition and integration. □

From this point forward, the dependency of $\tilde{P}_q$ on $\bar{t}_q$ and similar functions is suppressed for better readability of the equations.

Lemma 5. *The function R given by Lemma 4 is monotonic. That is, given $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C} \subset \mathcal{E}_v$, all disjoint sets, then $R(\mathcal{A} \cup \mathcal{C}, \mathcal{B}) \succcurlyeq R(\mathcal{A}, \mathcal{B})$.*

Proof. The Lemma is true if the following holds:

$$R(\mathcal{A} \cup \mathcal{C}, \mathcal{B}) = R(\mathcal{A}, \mathcal{B}) + U(\mathcal{A}, \mathcal{B}, \mathcal{C}) \tag{43}$$

with $U \succcurlyeq 0$. The inequality for monotonicity then becomes:

$$R(\mathcal{A}, \mathcal{B}) + U(\mathcal{A}, \mathcal{B}, \mathcal{C}) - R(\mathcal{A}, \mathcal{B}) = U(\mathcal{A}, \mathcal{B}, \mathcal{C}) \succcurlyeq 0. \tag{44}$$

To show that (43) holds it is enough to show that

$$\tilde{R}_q(\mathcal{A} \cup \mathcal{C}, \mathcal{B}) = \tilde{R}_q(\mathcal{A}, \mathcal{B}) + \tilde{U}_q(\mathcal{A}, \mathcal{B}, \mathcal{C}) \tag{45}$$

with $\tilde{U}_q \succcurlyeq 0$ holds for all $\tilde{R}_q$s that compose R in (42). For $q = 2$, $\tilde{R}_2 = 0$, which holds the inequality trivially, proving the base case of induction. For an arbitrary q, assuming it holds for $q - 1$, we have

$$\begin{aligned}
&\tilde{R}_q(\mathcal{A} \cup \mathcal{C}, \mathcal{B}) = \\
&= e^{N_0 t_q} \left(\sum_{k \in \mathcal{A} \cup \mathcal{C}} N_k \tilde{P}_{q-1}(\mathcal{B}) N_k^\top + \right. \\
&\qquad + \sum_{k \in \mathcal{B}} N_k \tilde{P}_{q-1}(\mathcal{A} \cup \mathcal{C}) N_k^\top + \\
&\qquad \left. + \sum_{k \in \mathcal{A} \cup \mathcal{C} \cup \mathcal{B}} N_k \tilde{R}_{q-1}(\mathcal{A} \cup \mathcal{C}, \mathcal{B}) N_k^\top \right) e^{N_0^\top t_q} \\
&= e^{N_0 t_q} \left(\sum_{k \in \mathcal{A}} N_k \tilde{P}_{q-1}(\mathcal{B}) N_k^\top + \sum_{k \in \mathcal{C}} N_k \tilde{P}_{q-1}(\mathcal{B}) N_k^\top \right. \\
&\qquad + \sum_{k \in \mathcal{B}} N_k \left(\tilde{P}_{q-1}(\mathcal{A}) + \tilde{P}_{q-1}(\mathcal{C}) + \tilde{R}_{q-1}(\mathcal{A}, \mathcal{C}) \right) N_k^\top \\
&\qquad + \sum_{k \in \mathcal{A} \cup \mathcal{B}} N_k \left(\tilde{R}_{q-1}(\mathcal{A}, \mathcal{B}) + \tilde{U}_{q-1}(\mathcal{A}, \mathcal{B}, \mathcal{C}) \right) N_k^\top \\
&\qquad \left. + \sum_{k \in \mathcal{C}} N_k \tilde{R}_{q-1}(\mathcal{A} \cup \mathcal{C}, \mathcal{B}) N_k^\top \right) e^{N_0^\top t_q} \\
&= \tilde{R}_q(\mathcal{A}, \mathcal{B}) + e^{N_0 t_q} \left(\sum_{k \in \mathcal{C}} N_k \tilde{P}_{q-1}(\mathcal{B}) N_k^\top \right. \\
&\qquad + \sum_{k \in \mathcal{B}} N_k \left(\tilde{P}_{q-1}(\mathcal{C}) + \tilde{R}_{q-1}(\mathcal{A}, \mathcal{C}) \right) N_k^\top \\
&\qquad + \sum_{k \in \mathcal{A} \cup \mathcal{B}} N_k \left(\tilde{U}_{q-1}(\mathcal{A}, \mathcal{B}, \mathcal{C}) \right) N_k^\top \\
&\qquad \left. + \sum_{k \in \mathcal{C}} N_k \tilde{R}_{q-1}(\mathcal{A} \cup \mathcal{C}, \mathcal{B}, \bar{t}_{q-1}) N_k^\top \right) e^{N_0^\top t_q} \\
&= \tilde{R}_q(\mathcal{A}, \mathcal{B}) + \tilde{U}_q(\mathcal{A}, \mathcal{B}, \mathcal{C}),
\end{aligned} \tag{46}$$

completing the induction step of the proof (positive semidefiniteness of $\tilde{U}_q$ is proved in exactly the same way as for $\tilde{R}_q$ in Lemma 4). □

Using Lemmas 3, 4, and 5, we prove Theorem 1 as follows.

Proof. A set function $\rho_\Sigma : 2^{\mathcal{E}_v} \to \mathbb{R}^+$ is supermodular if and only if it satisfies $\rho_\Sigma(\mathcal{B} \cup \{e\}) - \rho_\Sigma(\mathcal{B}) \geq \rho_\Sigma(\mathcal{A} \cup \{e\}) - \rho_\Sigma(\mathcal{A})$. for every $\mathcal{A}$, $\mathcal{B} \subset \mathcal{E}_v$ with $\mathcal{A} \subset \mathcal{B}$ and every $e \in \mathcal{E}_v / \mathcal{B}$. From Lemma 4, we can rewrite the left hand side of inequality above as $\rho_\Sigma(\mathcal{B} \cup \{e\}) - \rho_\Sigma(\mathcal{B}) = \rho_\Sigma(e) + \mathrm{trace}(R(\mathcal{B} \cup e))$, and the right hand side similarly. The supermodularity definition can be simplified to $\mathrm{trace}(R(\mathcal{B} \cup e)) \geq \mathrm{trace}(R(\mathcal{A} \cup e))$,where R is defined as in Lemma 5. This inequality always holds for our function because $\mathcal{A} \subset \mathcal{B}$ satisfies the conditions for Lemma 5. □