2012 American Control Conference
Fairmont Queen Elizabeth, Montréal, Canada
June 27-June 29, 2012

Vulnerable Links and Secure Architectures in the Stabilization of Networks of Controlled Dynamical Systems

Anurag Rai*, David Ward †, Sandip Roy‡, Sean Warnick*

Abstract— **This work considers destabilization attacks acting on a single link in a system's logical interconnection structure. This structure, or signal architecture, is characterized by the causal relationships among exposed signals within a network of interconnected systems. The concept of a vulnerable link is thus characterized, and necessary and sufficient conditions for identifying vulnerable links are provided. The vulnerability of various system architectures are then characterized by the vulnerability of their weakest link, and it is shown that every transfer function has a completely secure architecture with no vulnerable links. A numerical example then illustrate these concepts with concrete architectures.**

I. Introduction

The Stuxnet virus attacked an Iranian nuclear power plant in 2010 and caused the centrifuge's rotors to malfunction [13]. It gained much news coverage as the first virus attack on industrial systems. Although no serious damage was done, it has highlighted the necessity of improving our understanding of the security of control systems.

Researchers have predicted that attacks on industrial control systems would increase [3], [1]. As the systems are becoming more networked, securing them has become much harder and attacking them has gotten easier. In the past, securing the physical plants was enough to secure the system, but now in the networked architecture the communication channels have to be secured too. This is almost impossible to achieve, especially when these systems are being connected to the Internet, with connection features as powerful as remote access to the control centers. Regardless of the improvement in the industry's secure communications, cryptography, etc., a simple human error like someone forgetting to change a default password on their account could give an attacker complete access to the resources necessary to carry out a complicated attack. Although the security risks are real, these systems being less networked in the future is highly unlikely because of the usability advantages that a networked setting offers.

As a result, considering the security of networked control systems has become very important. A good design should make detecting attacks easy, help understand the effects of an attack, make it difficult to execute an attack, and finally minimize the consequences if an attack is successful. This paper will contribute in designing more secure networked systems by identifying conditions when a link is completely secure against attacks that attempt to destabilize the system. Our result also gives a measure of link vulnerability, which corresponds to the minimum size of a destabilizing attack on the link. This can be a useful tool to understand the security of a networked system.

In this paper, we view security as a robustness issue and focus on making systems robust against perturbations on a single communication channel. First, we give a summary of the types of attacks that a system might suffer. Then, in Section V we present our main result. Finally in Section VI we give some examples to illustrate the applications of our theory.

This research was supported by AFRL FA8750-09-2-0219. The authors are with the * Information and Decision Algorithms Laboratories, Brigham Young University, Provo, UT 84602, USA. † Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742, USA. ‡ School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164, USA. Contact at: anurag1985 [at] gmail.com, ward.david.2 [at] gmail.com, sroy [at] eecs.wsu.edu, sean.warnick [at] gmail.com.

II. Attack Models

In the literature, attacks on control systems have been classified into two types: *denial of service attacks*, when the attacker jams a channel in order to destabilize the system, and *deception attacks*, when the attack adds perturbations on particular links in order to compromises the reliability of the controller's state estimates [11]. We consider a hybrid attack model where the attacker adds perturbations to the channels, not just jam them, in order to destabilize the system. We call this type of attack a *destabilizing attack*.

A. Denial of Service (DoS) Attack

Denial of service attacks prevent signals from reaching their intended destination. This is probably the easiest and most common attack, and it is modeled as removal of an edge in an interconnected structure. It might be done by jamming the communication channel, disrupting the transmitter/receiver, changing the routing protocol, saturating the receiver with extraneous signals, etc. The attacker's intent of such an attack could be to degrade the system performance or to completely destabilize it. [9] shows that performance of networked control systems could decrease significantly under a DoS attack. [11] gives a method to find an optimal controller that minimizes the effect of such an attack on linear control systems.

In [12], the authors study whether a DoS attack on certain links can make the system unreachable or uncontrollable. They also develop graph theoretic algorithms to identify the minimal number of edges which are necessary for preserving controllability and observability.

978-1-4577-1094-0/12/$26.00 ©2012 AACC

B. Deception Attack

The goal of a deception attack is to change the state estimates computed by a model-based controller. This type of attack is modeled as a stable additive perturbation to an edge in the network. All stabilizing controllers make the closed loop system stable, hence, a stabilizing controller is necessarily stabilizable from the plant. So, if at attacker gains access to the communication channel between the plant and the controller, state estimates of a model-based controller can be altered. To prevent this, many real systems such as power systems, sensor networks, etc., are equipped with a Bad Data Detector (BDD) [14], [7], [15]. A BDD, using the model of the plant, detects deviation of the state estimates from the expected and raises an alarm to notify the human operator. Because of the presence of measurement noise, this deviation is never zero, so the BDD ignores deviations that are smaller than a specified threshold. Hence, in the presence of BDDs, the attack has to change the state estimates without increasing the chance of raising an alarm.

In [14] the authors study this kind of attack in the context of a power system. They shown that it is in fact possible for an attacker to change the state estimates to a specific value without increasing the chance of being detected. [15] studies a similar problem in the scenario of a wireless sensor networks. It maps an approximation of the set of all possible values the attacker could drive the estimates to.

[7] studies a slightly different problem. Here, the goal of the attacker is to change the estimate of one of the states without increasing the chance of being detected. The authors recognize that while doing this the attacker might want to use the fewest channels possible or might try to keep the magnitude of the attack signal small. For each type of attack, the authors then give a formulation of a *security index* of the system.

C. Destabilizing Attack

Like deception attacks, these attacks effectively arise as an additive perturbation on a link in the system interconnection structure. Unlike deception attacks, however, they seek to destabilize the system rather than simply move the system state to a desired value without being detected. BDDs are clearly capable of detecting the destabilization resulting from such attacks, nevertheless serious damage and even complete plant shut-down may result by the time system operators are able to do anything about it.

A rich literature in systems and control theory explores the destabilization of systems due to additive perturbations, see for example [6] and the references therein. Security analysis of destabilizing attacks thus appears to be a robustness problem with respect to certain classes of perturbations. Indeed, we adopt this point of view, and consider security problems to be essentially robustness problems of various types.

The contribution of this work, then, applied to this class of attacks, is in the solution of a certain class of robustness problems over a particular kind of link model–corresponding to logical, rather than the physical, links of a system–and with respect to a specific class of perturbations. Unlike standard robustness measures that generally consider destabilizing perturbations acting over all channels and nodes of a system, here we restrict our attention specifically to perturbations that disrupt a single link in the system's signal structure. Our analysis then considers such single-link perturbations over all possible system links. In the next section we explore our link model in detail.

III. Link Models

The destabilizing attacks considered here are additive perturbations acting on a single link in a system's logical interconnection structure. There are many characterizations of a system's structure, see for example [5], [4]. One characterization would consider the interconnection structure among subsystems. This definition of structure, also called the system's subsystem structure, would represent the physical interconnection between physical components of a particular networked system. Under this notion of structure, a *link* would represent the signal passing between two subsystem nodes within the subsystem interconnection architecture. In contrast to the subsystem structure, this work considers another definition of system structure and, consequently, a different notion of a system link.

In this work, we consider a partition on signals of the system into two categories: exposed signals and hidden signals. The logical interconnection structure, or architecture–also called the system's signal structure–is the causal relationship between exposed signals in the system. In this definition of structure, a *link* is a system describing the causal dependency between two exposed signal nodes of the logical interconnection architecture.

Some important consequences of this definition of link include the fact that a link may represent a very indirect and complicated pathway–through various hidden signals that may be components of other links in the system. Thus a link is associated with a particular set of dynamics–a system–that characterizes how the input signal is transformed into the output signal. The fact that hidden signals may be shard between links, however, is an important distinction between signal and subsystem interconnection structures. Note that a state of one subsystem, interconnected with others in a subsystem architecture (such as a standard feedback interconnection between two blocks), is never shared with other subsystems; the subsystem architecture effectively partitions the states of the networked system. In contrast, states on the links of the signal structure may, in fact, be shared with those of other links. This degree of abstraction is important for security problems because an additive perturbation on a link of the signal structure does not represent the corruption of a particular channel, as it would in the subsystem structure, but rather the idea that an attacker infiltrated a particular dependency between specific manifest variables.

The next section provides some background on Dynamical Structure Functions (DSF), which are used to represent the signal structure of a system. The DSF is a system representation that describes more structure, in the logical

interconnection sense, than the transfer function provides, but less than the state space realization would reveal. Specifically, the DSF describes exactly the causal dependencies between manifest variables without offering any indication of the structure relating hidden variables. As a result, although every state space realization specifies a unique DSF, and every DSF specifies a unique transfer function, there are many DSF architectures consistent with any specific transfer function, and many state space realizations consistent with any specific DSF.

IV. Background: Dynamical Structure Function

Before developing the main theorem, we will present a concise derivation of the dynamical structure function, and explain its relevance to the security of a networked system. For a complete derivation and results on different representations of structure see [4], [10], [8].

Let us consider a state-space LTI system

$$\begin{aligned}\begin{bmatrix}\dot{z_1}\\ \dot{z_2}\end{bmatrix} &= \begin{bmatrix}\bar{A}_{11} & \bar{A}_{12}\\ \bar{A}_{21} & \bar{A}_{22}\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix} + \begin{bmatrix}\bar{B}_1\\ \bar{B}_2\end{bmatrix} u \\ y &= \begin{bmatrix}\bar{C}_1 & \bar{C}_2\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix},\end{aligned} \tag{1}$$

where $\begin{bmatrix}\bar{C}_1 & \bar{C}_2\end{bmatrix}$ has full row rank. This system can be transformed to:

$$\begin{aligned}\begin{bmatrix}\dot{y}\\ \dot{x}\end{bmatrix} &= \begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}\begin{bmatrix}y\\ x\end{bmatrix} + \begin{bmatrix}B_1\\ B_2\end{bmatrix} u \\ y &= \begin{bmatrix}I & 0\end{bmatrix}\begin{bmatrix}y\\ x\end{bmatrix},\end{aligned} \tag{2}$$

Here y are the states that are measured, and x are the hidden states. Now, taking Laplace Transforms of the signals in (2), we get

$$\begin{bmatrix}sY\\ sX\end{bmatrix} = \begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}\begin{bmatrix}Y\\ X\end{bmatrix} + \begin{bmatrix}B_1\\ B_2\end{bmatrix} U. \tag{3}$$

Solving for X in the second equation of 3 gives

$$X = (sI - A_{22})^{-1}A_{21}Y + (sI - A_{22})^{-1}B_2U$$

Substituting into the first equation of (3) we get,

$$sY = WY + VU,$$

where $W = A_{11} + A_{12}(sI - A_{22})^{-1}A_{21}$ and $V = A_{12}(sI - A_{22})^{-1}B_2 + B_1$. Let D be a diagonal matrix with the diagonal entries of W. Then,

$$(sI - D)Y = (W - D)Y + VU.$$

Now we can rewrite this equation as,

$$Y = QY + PU, \tag{4}$$

where

$$Q = (sI - D)^{-1}(W - D)$$

and

$$P = (sI - D)^{-1}V.$$

The matrix Q is a matrix of transfer functions from Y_i to Y_j, $i \neq j$, or relating each measured signal to the other measured signals. Note that Q is zero on the diagonal and either zero or a strictly proper transfer function on the off diagonal. The matrix P is a matrix of zeros or strictly proper transfer functions from each input to each output without depending on any additional measured states. Together, the pair $(Q(s), P(s))$ is called the *dynamical structure function* for system (1).

The transfer function matrix for this system is given by

$$G = (I - Q)^{-1}P = C(sI - A)^{-1}B.$$

Hence, G_{ij} is the closed loop transfer function from input j to state i. In this paper, we will also refer to the closed loop transfer function between states. A transfer function from state j to state i is represented by H_{ij}, where

$$H = (I - Q)^{-1}.$$

Note that the transfer function from a state to an input is always zero.

Definition 1: Given a system 1 with signal structure characterized by the dynamical structure function (P,Q), a *link* (i, j) of the system corresponds to any nonzero entry in P or Q.

Note that P gives the links from the inputs to the measured states, and Q gives the links that represent the dependencies between the measured states. The next section will introduce the notion of vulnerability and characterize vulnerable links in the system's architecture characterized by (P, Q).

V. Vulnerable Links

In this work, vulnerability refers to the destabilization of a system resulting from the corruption of a single link in its signal architecture. We begin with a definition of a vulnerable link.

Definition 2: Given a system 1 with signal structure characterized by the dynamical structure function (P,Q), a link in (P,Q) is called *vulnerable* if there exists a stable perturbation on the link that makes the system unstable.

Example 1: Let us consider a system with

$$P = \begin{bmatrix}\frac{1}{s+2} & 0\\ 0 & \frac{1}{s+2}\end{bmatrix}, \text{ and } Q = \begin{bmatrix}0 & \frac{1}{s+2}\\ \frac{1}{s+2} & 0\end{bmatrix}.$$

This system is stable because the transfer function,

$$G = \frac{1}{s^2 + 4s + 3}\begin{bmatrix}s+2 & 1\\ 1 & s+2\end{bmatrix}$$

Now let us add a perturbation $\Delta = \frac{3}{s+2}$ to the link Q_{12} as shown in Figure 1. The resulting transfer function is

$$\bar{G} = \frac{1}{s(s+4)}\begin{bmatrix}s+2 & 1\\ 4 & s+2\end{bmatrix},$$

which is unstable. Hence the link Q_{12} is a vulnerable link. Similarly, it can be shown that Q_{21} is vulnerable, although neither P_{11} nor P_{22} are vulnerable.

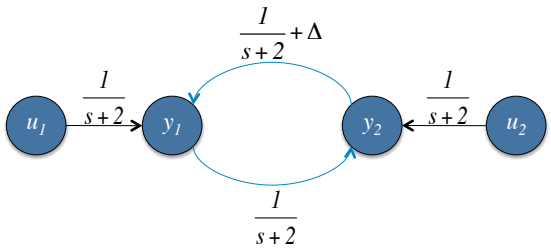


Fig. 1. The system with the perturbation Δ. Black arrows indicate secure links, while blue arrows indicate vulnerable links.

A. Condition for Vulnerability

Given that an attacker has the knowledge of the dynamical structure function representation of a system, we will derive a necessary and sufficient condition for a link to be vulnerable.

Theorem 1: Let us consider a stable system (P, Q). There exists a stable additive perturbation Δ on a link from node i to node j, either in P or Q, that makes the system unstable if and only if the closed loop transfer function from node j to i is nonzero.

Proof: The system with the perturbation Δ can be represented as the linear fractional transformation in Figure 2, where T is the associated closed loop transfer function, and w_i, w_j represent the signals at node i and j respectively. This system is stable if and only if the system in Figure 3 is stable (see [6]). If $T_{ij} = 0$, any stable Δ does not affect the stability of the system in Figure 3. Thus the closed loop system in Figure 2 is stable for all Δ.

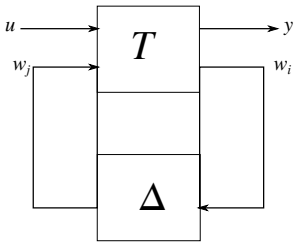


Fig. 2. System with the perturbation $\Delta e_i e_j^T$

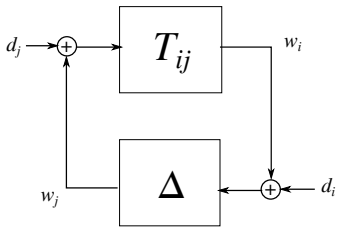


Fig. 3. Necessary and sufficient condition for stability of the system in Figure 2

If $T_{ij} \neq 0$, then the system in Figure 3 is unstable if any of the transfer functions of $\begin{bmatrix} d_j \\ d_i \end{bmatrix} \rightarrow \begin{bmatrix} w_j \\ w_i \end{bmatrix}$ is unstable. We have,

$$w_j = \frac{1}{1 - T_{ij}\Delta} \begin{bmatrix} T_{ij}\Delta & \Delta \end{bmatrix} \begin{bmatrix} d_j \\ d_i \end{bmatrix}.$$

Let $T_{ij} = \frac{N}{D}$ and $\Delta = \frac{\delta_N}{\delta_D}$, then

$$w_j = \frac{D\delta_D}{D\delta_D - N\delta_N} \begin{bmatrix} \frac{N\delta_N}{D\delta_D} & \frac{\delta_N}{\delta_D} \end{bmatrix} \begin{bmatrix} d_j \\ d_i \end{bmatrix}. \tag{5}$$

For a polynomial to be stable it is necessary that all its coefficients are of the same sign. In the case of the polynomial

$$R(s) = D\delta_D - N\delta_N, \tag{6}$$

it is easy to see that a properly designed Δ can zero out at least one of the terms. Thus, there exists a Δ that destabilizes these transfer functions. ■

Note that when we are considering the vulnerability of the links in Q, $T = H = (I - Q)^{-1}$, gives the closed loop transfer functions. Now, we will present some implications of this result.

Corollary 1: None of the links in P are vulnerable.

Proof: This is true because the transfer function from the states to the input is always zero. ■

Corollary 2: If T_{ij} is nonzero, there exists a perturbation $\Delta \in \mathbb{R}$ that destabilizes the system in Figure 3.

Proof: Let $\Delta \in \mathbb{R}$, $l_{ij} = \frac{N_l}{D_l}$. Thus, $\frac{\delta_n}{\delta_d} = \frac{\Delta D_l + N_l}{D_l}$, and the polynomial in (6) becomes $D_l D - N(D_l\Delta + N_l)$. We can see that at least one of the terms in this polynomial can be zeroed out by choosing appropriate Δ, making the polynomial unstable. ■

Corollary 3: Let us consider a stable system,

$$\begin{aligned} \dot{x} &= Ax + Iu, \\ y &= Ix, \end{aligned} \tag{7}$$

where $A \in \mathbb{R}^{n \times n}$ and let $G = (sI - A)^{-1}$. There exists a perturbation $K = \Delta e_i e_j^T, \Delta \in \mathbb{R}$, such that $(A + K)$ is not Hurwitz, if and only if the transfer function from input u_i to output y_j, G_{ji}, is nonzero.

Proof: If the perturbation is on the diagonal entry of A, then it is easy to see that a destabilizing perturbation always exists and G_{ii} is never zero. Let $D = diag(A_{11}, A_{22}, ..., A_{nn})$. The dynamical structure function of the system is given by $P = (sI - D)^{-1}$ and $Q = (sI - D)^{-1}(A - D)$. Any perturbation $K = \Delta e_i e_j^T, i \neq j$ effects only the link Q_{ij}. Hence, the perturbation can make the system unstable if and only if the transfer function H_{ji} is nonzero. Also, the diagonal entries of P are nonzero, and $G = HP$. Thus, the transfer function H_{ij} is nonzero if and only if G_{ji} is nonzero.

■

Example 2: Let us consider a system of the form 7 with

$$A = \begin{bmatrix} -1 & 0 & -4 & 3 \\ 2 & -2 & 0 & 0 \\ 3 & 0 & -2 & -4 \\ 0 & 3 & -2 & -5 \end{bmatrix}.$$

Here the eigenvalue of A are $\sigma = \{-1.5000 + 3.4278j, -1.5000 - 3.4278j, -6.7016, -0.2984\}$. Hence, the system is stable. In this system, the link from x_4 to x_1 is not vulnerable because $G_{41} = 0$. Notice that this example is

not a trivial example, like a diagonal or a triangular system, since there are cycles that contain both nodes x_1 and x_4.

Corollary 4: Let $A \in \mathcal{R}^{n\times n}$. A perturbation on the $(i,j)^{th}$ entry of A changes its eigenvalues if and only if the $G_{ji} \neq 0$, where $G = (sI - A)^{-1}$ is the transfer function matrix i.e. the (i,j) minor of $(sI - A)$ is nonzero.

Proof: Take the system from Corollary 3. We can see that a perturbation on the $(i,j)^{th}$ entry has no effect on the system if $G_{ij} = 0$. Also, if $G_{ji} \neq 0$, the perturbation forms a closed loop system, such as the one given in Figure 3, in which case Δ definitely changes the poles of the system. ■

If we take the A matrix from Example 2, note that its eigenvalues stay unchanged for any perturbation on the $(1,4)^{th}$ entry.

B. Structure and Vulnerability

To perform the vulnerability analysis of a system, we assume that the attacker can only modify existing links and cannot create new links. With this assumption, we can see that systems where the output nodes do not form a cycle are always secure, because in such a case the nodes can be permuted to obtain a triangular Q matrix. A triangular Q matrix gives a triangular H, and by applying Theorem 1 we can see that all the existing links are secure. Note that secure links doesn't always mean they are from a triangular system. For example, the link Q_{14} is secure in the system given in Figure 4, which is the signal structure architecture of the state-space system in Example 2.

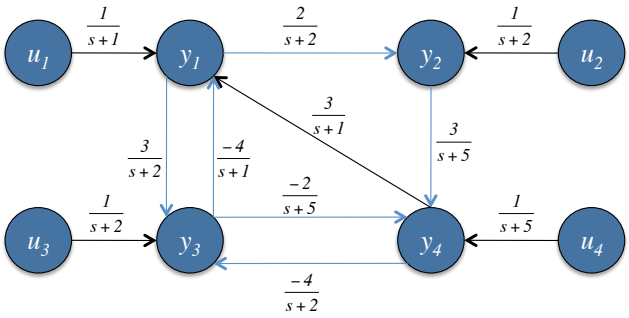


Fig. 4. A system with a secure link in a cycle. Black arrows represent the secure links.

Noting that certain graphical structures result in secure links begs the question of whether there are particular dynamics that contribute to secure or vulnerable links in the system's architecture. The following theorem answers this question.

Theorem 2: Every transfer function G has a completely secure architecture $(\bar{P}, \bar{Q})$.

Proof: For any transfer function G, note that $(P = G, Q = 0)$ is an admissible Dynamical Structure Function since $G = (I - 0)^{-1}G$. From Corollary 1, we see that none of the links in P are vulnerable, and since Q has no links, the system is secure. ■

This result shows that the vulnerability of a system is structure dependent and not a function of the system dynamics. This fact highlights one difference between the vulnerability, which depends on the system structure and not the dynamics, and the robustness, which depends on the dynamics and not the system structure.

C. Measure of Vulnerability

Feedback is very common in both natural and engineered systems. Nevertheless, such structures usually generate vulnerable links. Thus, a measure of vulnerability is essential to understand the security of the system.

Given a signal architecture (P, Q) with associated closed loop transfer function T, the vulnerability of link (i,j) is given by

$$v_{ji} = ||T_{ij}||_\infty, \tag{8}$$

which is the inverse of the smallest perturbation required on link (i,j) to destabilize the system. Since all the links in P are secure, we only consider the links in Q while computing the vulnerability, hence, $T = H$. The vulnerability of the system is given by

$$V = \max_{(i,j)\in Q} v_{ji} \tag{9}$$

$$= \max_{(i,j)\in Q} ||T_{ij}||_\infty \tag{10}$$

This measure allows us to associate a size of the smallest destabilizing perturbation with every link in the system architecture. Secure links thus have a vulnerability of 0. Note that V, the system vulnerability, is less than or equal to the inverse of the size of the smallest destabilizing perturbation for the system, since link perturbations are restricted to act on a single link only.

VI. Numerical Example

Let us consider a system with the architecture given in Figure 5(a) where,

$$P = \begin{bmatrix} \frac{1}{s+1} & 0 & 0 \\ 0 & \frac{1}{s+1} & 0 \\ 0 & 0 & \frac{1}{s+1} \end{bmatrix}$$

and

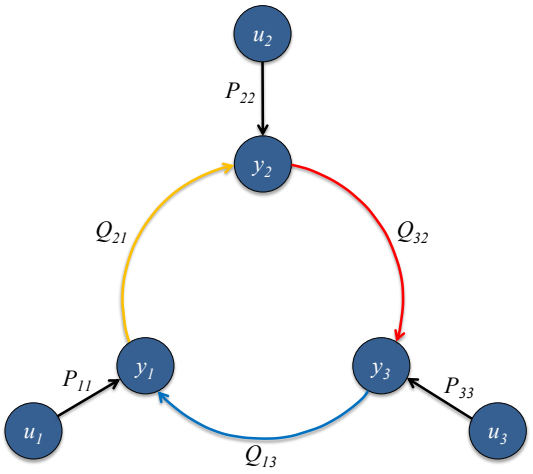
$$Q = \begin{bmatrix} 0 & 0 & \frac{1}{s+1} \\ \frac{1}{s+2} & 0 & 0 \\ 0 & \frac{1}{s+3} & 0 \end{bmatrix}.$$

The transfer function matrix for the system is given by

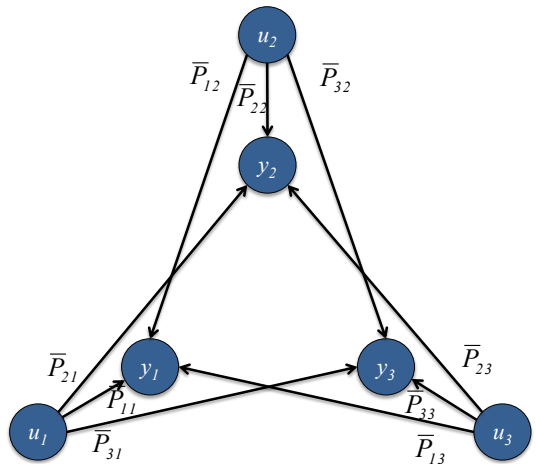
$$G = \begin{bmatrix} \frac{s^3+6s^2+11s+6}{d(s)} & \frac{s+2}{d(s)} & \frac{s^2+5s+6}{d(s)} \\ \frac{s^2+4s+3}{d(s)} & \frac{s^3+6s^2+11s+6}{d(s)} & \frac{s+3}{d(s)} \\ \frac{s+1}{d(s)} & \frac{s^2+3s+2}{d(s)} & \frac{s^3+6s^2+11s+6}{d(s)} \end{bmatrix},$$

where $d(s) = s^4 + 7s^3 + 17s^2 + 16s + 5$. By small gain theorem, the size of the smallest destabilizing perturbation is $||G||_\infty^{-1} = 0.42$ could destabilize the system.

Let $H = (I - Q)^{-1}$ represent the transfer function between the measured states y_i. Since the links in P are not vulnerable, we consider the perturbations on the links in Q which are the links (y_1, y_2), (y_2, y_3), and (y_3, y_1).



(a) Vulnerable architecture



(b) Completely secure architecture

Fig. 5. Vulnerable and secure architectures for the same transfer function. Black links are secure, vulnerable links are colored blue, yellow, and red in the increasing order of their vulnerability.

To compute the vulnerability of these links we need the following transfer functions:

$$H_{12} = \frac{s+2}{s^3+6s^2+11s+5}$$
$$H_{23} = \frac{s+3}{s^3+6s^2+11s+5}$$
$$H_{31} = \frac{s+1}{s^3+6s^2+11s+5}.$$

For this system $v_{21} = 0.4$, $v_{32} = 0.6$, and $v_{13} = 0.2$. Hence, $V = v_{23} = 0.6 < ||G||_\infty$, and the smallest perturbation on a single link that can destabilize this system must have a gain of $\frac{1}{V} = 1.67$.

This system can also be implemented as shown in Figure 5(b), where $\bar{P} = G$. This is one of the secure implementations of the system in Figure 5(a). From this example we thus observe the following:

- The same transfer function can exhibit both vulnerable and secure architectures,
- System robustness, characterized by the size of the smallest destabilizing perturbation (0.42 in this example), is not equivalent to the inverse of the system vulnerability, characterized by the size of the smallest destabilizing perturbation on a single link (about 1.67 in this example),
- Only links in Q can be vulnerable.

VII. Conclusion

This paper explored the notion of a vulnerable link in a network of controlled linear dynamical systems. The architecture of the system was characterized by its Dynamical Structure Function, representing the logical interconnection structure of the system. Vunerability was then defined as the size of the smallest destabilizing perturbation acting on a single link. The main results of the paper provided necessary and sufficient conditions for the vulnerability of a link and then demonstrated that any transfer function has a completely secure architecture. This result highlights the idea that while robustness is a property of a system's dynamics, security (in the sense discussed here) is a property of its signal architecture. Future work will explore parallel notions for subsystem interconnection structure of networked systems.

References

[1] A. A. Cardenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," in the Proceedings of the 3rd conference on Hot topics in security, pages 16, Berkeley, CA, USA, 2008.

[2] D. Ward. "Decentralized control problems: a weak structural approach." Undergraduate Honors Thesis, Department of Computer Science, Brigham Young University, 2010.

[3] E. Byres and J. Lowe, "The myths and facts behind cyber security risks for industrial control systems," in the Proceedings of the VDE Kongress, VDE Congress, 2004.

[4] E.Yeung, J. Goncalves, H. Sandberg, S. Warnick, "Mathematical relationships between representations of structure in linear interconnected dynamical systems," to appear in the Proceedings of the American Control Conference, 2011.

[5] E. Yeung, J. Goncalves, H. Sandberg, S. Warnick, "Representing Structure in Linear Interconnected Dynamical Systems," in the Proceedings of IEEE Conference on Decision and Control, Atlanta, USA, December, 2010.

[6] G. E. Dullerud and F. Paganini. *A course in robust control theory, a convex approach*,' Springer, 2000.

[7] H. Sandberg, A. Teixeira, and K. H. Johansson, On security indices for state estimators in power networks," in Preprints of the First Workshop on Secure Control Systems, CPSWEEK, 2010.

[8] J. Goncalves, S. Warnick. "Necessary and Sufficient Conditions for Dynamical Structure Reconstruction of LTI Networks," IEEE Transactions on Automatic Control, August 2008.

[9] M. Long, C.H. Wu, J. Y. Hung, "Denial of Service Attacks on Network-Based Control Systems: Impact and Mitigation," in IEEE Transactions on Industrial Informatics, vol 1, no. 2, May 2005.

[10] R. Howes. "Application and Properties of Dynamical Structure Functions". Undergraduate Honors Thesis, Department of Computer Science, Brigham Young University, 2008.

[11] S. Amin, A. Cardenas, and S. S. Sastry, "Safe and secure networked control systems under denial-of-service attacks," in Hybrid Systems: Computation and Control, pages 3145. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, April 2009.

[12] V. Pichai, M.E. Sezer, D.D. Siljak, "Vulnerability of dynamic systems," in International Journal of Control, 1981, pp. 1049-1060.

[13] Wikipedia, *Stuxnet*. `http://en.wikipedia.org/wiki/Stuxnet`, retrieved 3/17/2011.

[14] Y. Liu, P. Ning, and M. Reiter, "False data injection attacks against state estimation in electric power grids," in the Proceedings of the 16th ACM conference on Computer and communications security, 2009.

[15] Y. Mo, E. Garone, A. Casavola, B. Sinopoli, "False data injection attacks against state estimation in wireless sensor networks," in the Proceedings of 49th IEEE Conference on Decision and Control, Atlanta, GA, USA, 2010.