Vulnerability Analysis of Feedback Systems

Nathan Woodbury
Advisor: Dr. Sean Warnick

Honors Thesis Defense
11/14/2013

IDeA Labs
Information & Decision Algorithms Laboratories

Acknowledgements

- Advisor
  - Dr. Sean Warnick
- Honors Committee
  - Dr. Scott Steffensen
  - Dr. Sandip Roy
- IDeA Labs
  - Anurag Rai
  - Vasu Chetty
  - Phil Paré
- My Family

Outline

- Introduction: Vulnerability
- Mathematical Preliminaries
  - Three System Representations & Their Structures
- Open-Loop Results: Secure Structures (DAGs)
- Closed-Loop Results: Can Fight Fire with Fire
- Conclusions

Introduction: Vulnerability

Preliminaries: Systems and Structure

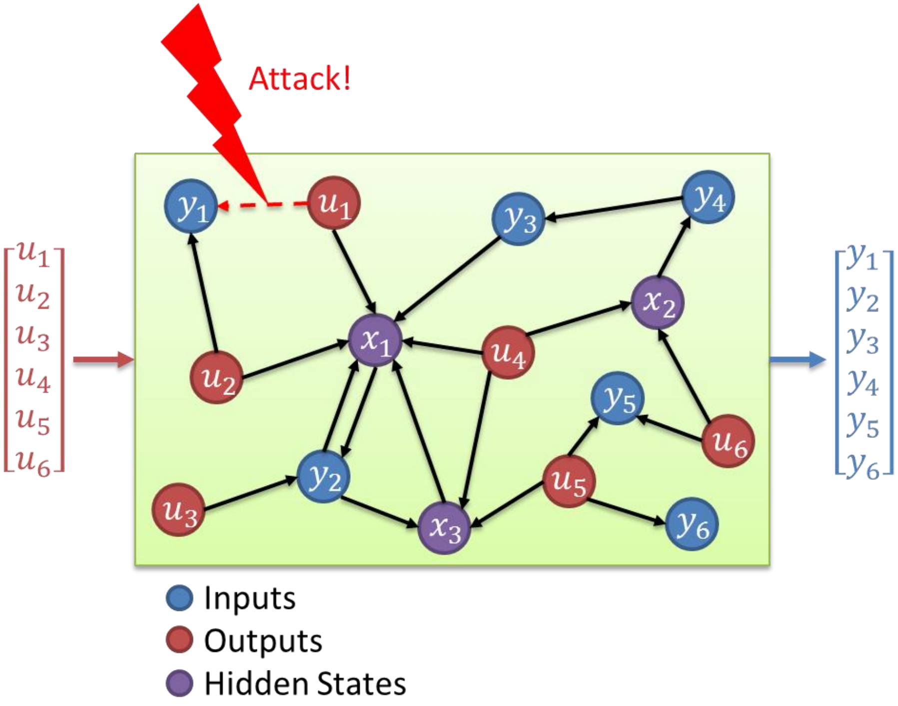
Vulnerability in Open-Loop Systems

Vulnerability in Closed-Loop Systems

INTRODUCTION: VULNERABILITY

Attack Models

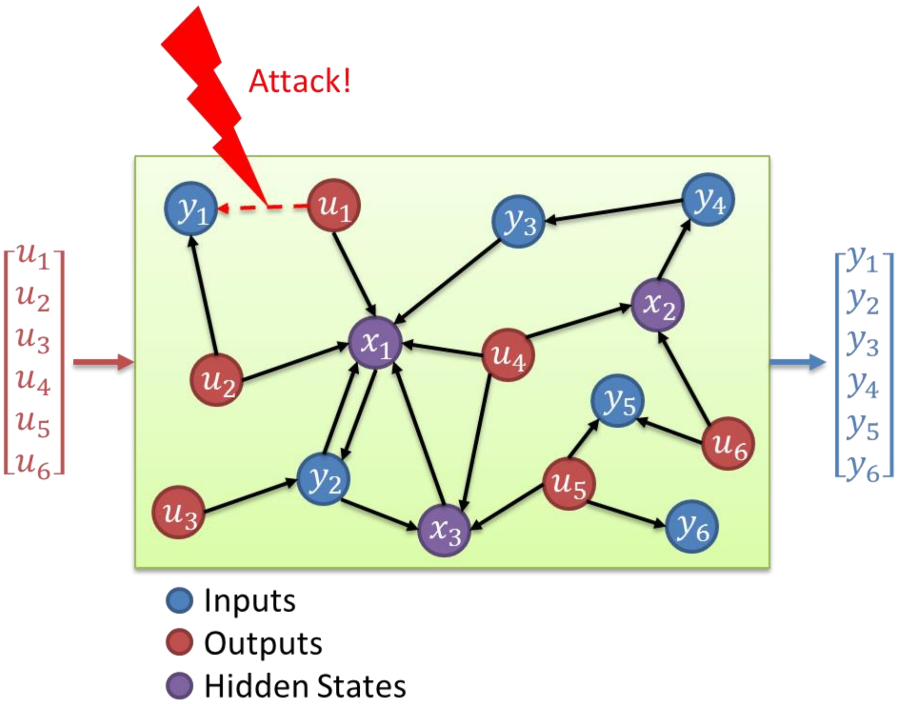
- Denial of Service
  - Removal of Link
- Deception
  - Interception and Modification of a Link



Information & Decision Algorithms Laboratories

Attack Models

- Denial of Service
  - Removal of Link
- Deception
  - Interception and Modification of a Link



Underlying Perspective:
Both models involve a distributed system where an enemy does bad things on a link.

Attack Models

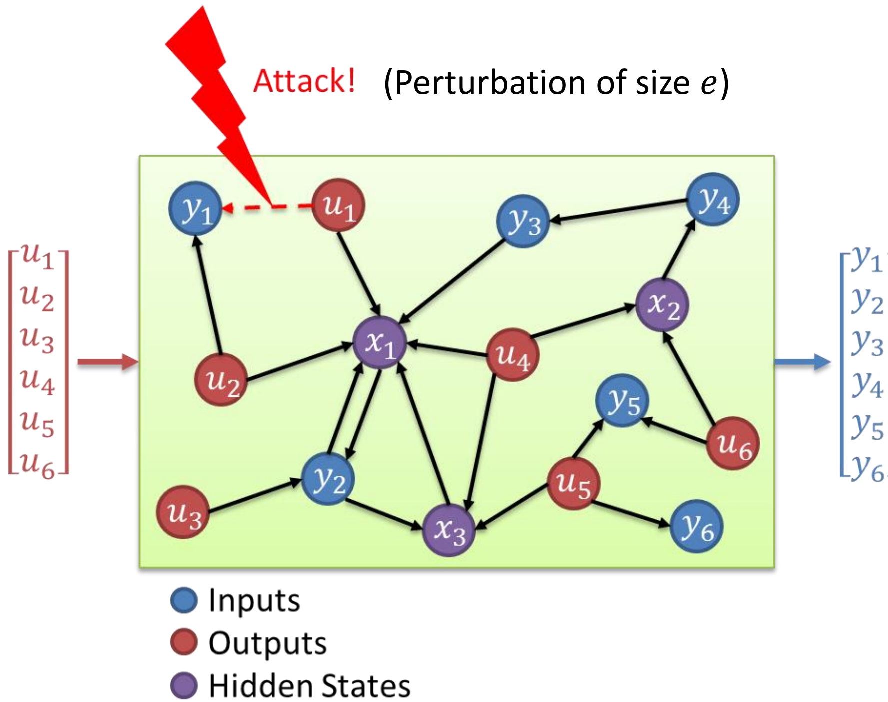
- Denial of Service
  - Removal of Link
- Deception
  - Interception and Modification of a Link

Underlying Perspective:
Both models involve a distributed system where an enemy does bad things on a link.

- Destabilization Attack
  - Attack a Single Link
  - Destabilize Entire System
    - Link Failure
    - Malicious Attack
- Vulnerability
  - Sensitivity of stability to link perturbations
  - Depends on Structure

Definition of Vulnerability

- Define e: (attacker) effort
  - smallest signal attacker can place on a particular link to destabilize system.
- Link Vulnerability: $\frac{1}{e}$
  - More effort to destabilize→less vulnerable
  - Less effort to destabilize→more vulnerable
- System Vulnerability:
  - Max vulnerability over all links
- System Representation defines notion of "link"



Introduction: Vulnerability

Lesson: Vulnerability is a Property of Links

Preliminaries: Systems and Structure

Vulnerability in Open-Loop Systems

Vulnerability in Closed-Loop Systems

SYSTEMS AND STRUCTURE

What is Structure

- System structure is represented by a graph
  - Shows flow of information
- One system can be represented by many structures
  - We will discuss three (State Space Representations, Transfer Functions, and Dynamical Structure Functions)

Information & Decision Algorithms Laboratories

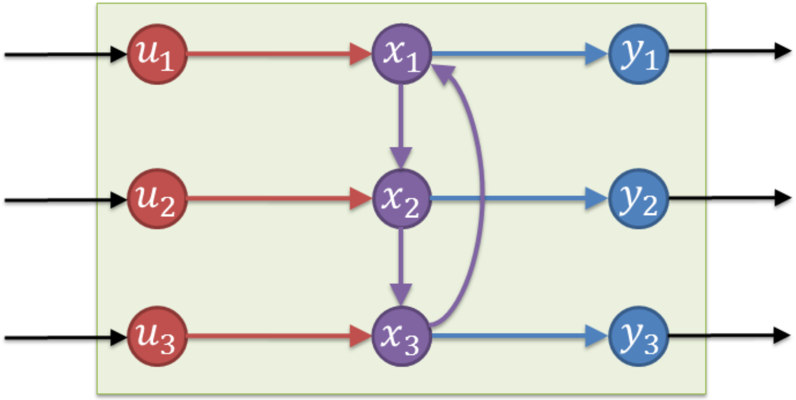
State Representations

- Inputs, outputs, and internal states

$$\dot{x} = \begin{bmatrix} -1 & 0 & -1 \\ -2 & -3 & 0 \\ 0 & -2 & -3 \end{bmatrix} x + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x$$

The "internal wiring" of the system.

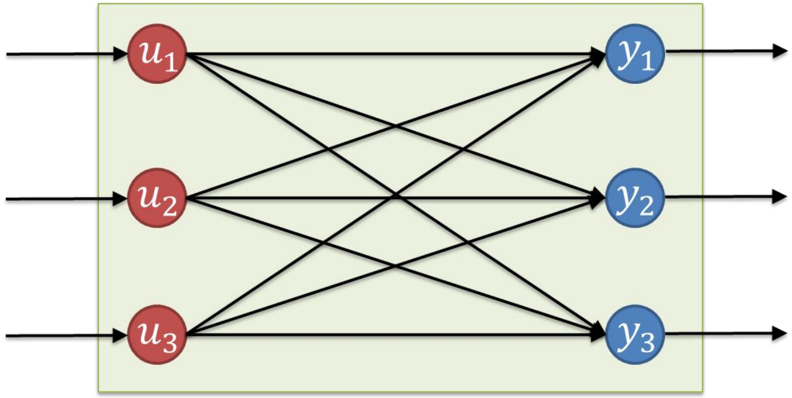


Transfer Functions

- Input-output behavior
- "Black Box"

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

$$f(s) = s^3 + 7s^2 + 15s + 13.$$

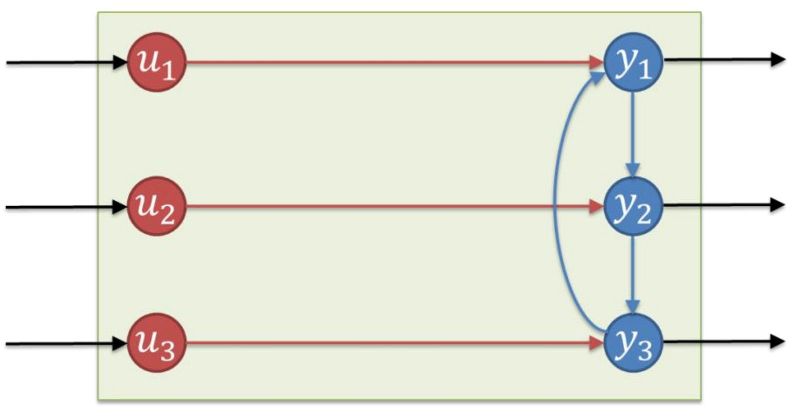


The "design" of the system doesn't worry about implementation, only its input-output behavior

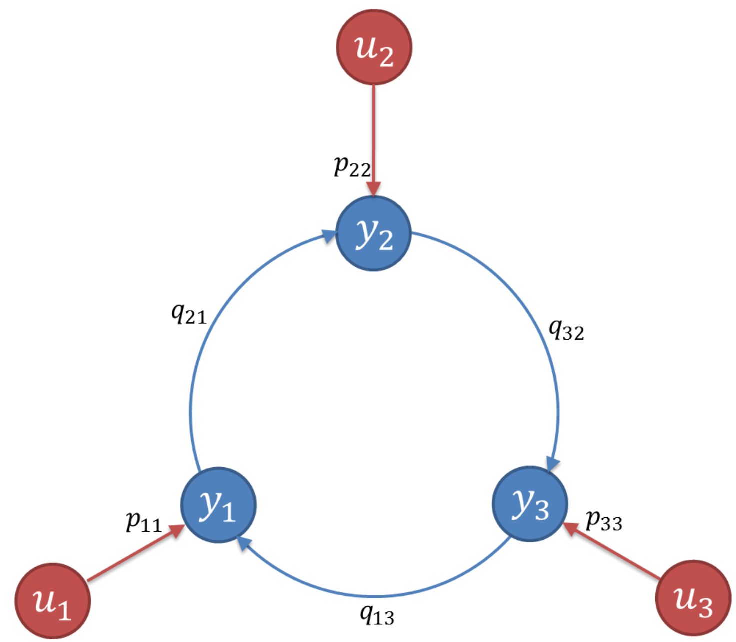
Dynamical Structure Functions (DSFs)

- Factorization of the Transfer Function
  $G = (I - Q)^{-1}P$

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



An implementation of a system.



Introduction: Vulnerability

Lesson: Vulnerability is a Property of Links

Preliminaries: Systems and Structure

Lesson: Definition of Link Depends on Structure, which depends on Implementation

Vulnerability in Open-Loop Systems

Vulnerability in Closed-Loop Systems

VULNERABILITY IN OPEN-LOOP SYSTEMS

Open-Loop Problem Formulation

- English: Given a system design, design its structured implementation to minimize system vulnerability
  - Fact: Links in P don't matter
- Math: Given a fixed TF G, Choose DSF Q (with $P = (I - Q)G$) such that the system vulnerability is minimized:

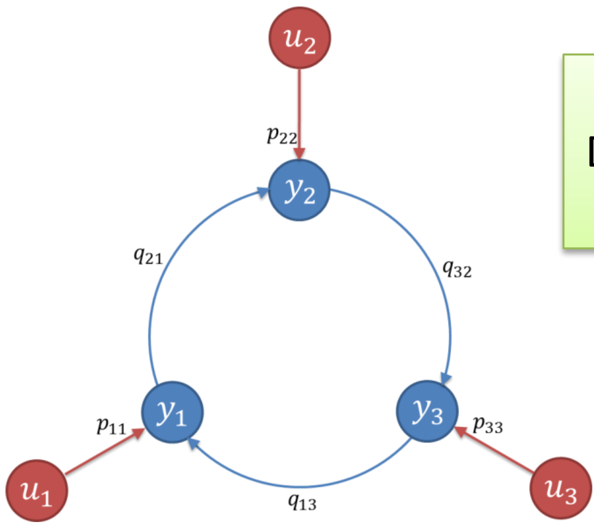
$$\min_{Q} \|(I - Q)^{-1}\|_{1-\infty}$$

$1 - \infty$: Size of matrix element (i, j) with largest norm

Generally, this is a hard problem to solve (non-convex)

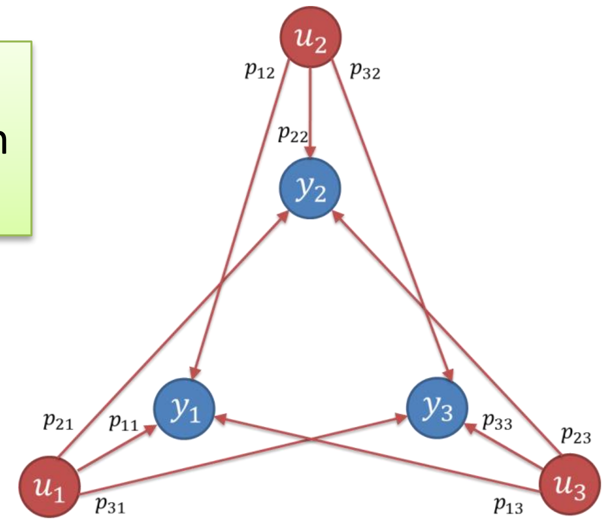
Conditions of Vulnerability

- *Theorem 1:* A link is vulnerable if and only if it is part of a cycle.
- *Theorem 2:* It is always possible to implement a completely secure open-loop system.



Vulnerable Architecture
Q has internal feedback
$P = (I - Q)G$

Same system
Different Implementation
Different Vulnerabilities



One Secure Architecture
$Q = 0$ (No Blue Links)
$P = (I - Q)G = G$

Introduction: Vulnerability

Lesson: Vulnerability is a Property of Links

Preliminaries: Systems and Structure

Lesson: Definition of Link Depends on Structure, which depends on Implementation

Vulnerability in Open-Loop Systems

Lesson: Links in cycles are vulnerable

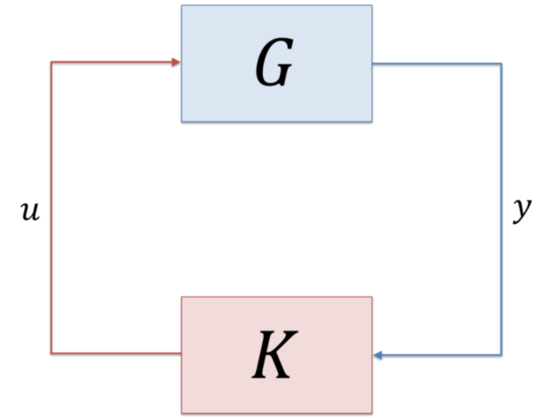
To remove vulnerability, remove cycles

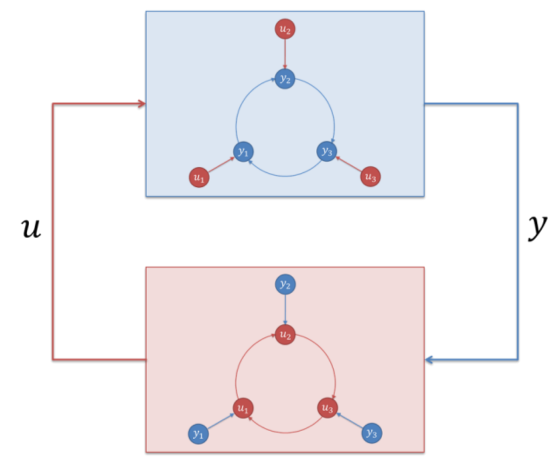
Vulnerability in Closed-Loop Systems

VULNERABILITY IN CLOSED-LOOP SYSTEMS

Motivation

- Sometimes, feedback is necessary
- Given system G, design second system K so that
  - G and K are connected in feedback
  - The combined system behaves well
- Our Goal
  - Decide best structure, or implementation, of K to minimize vulnerability





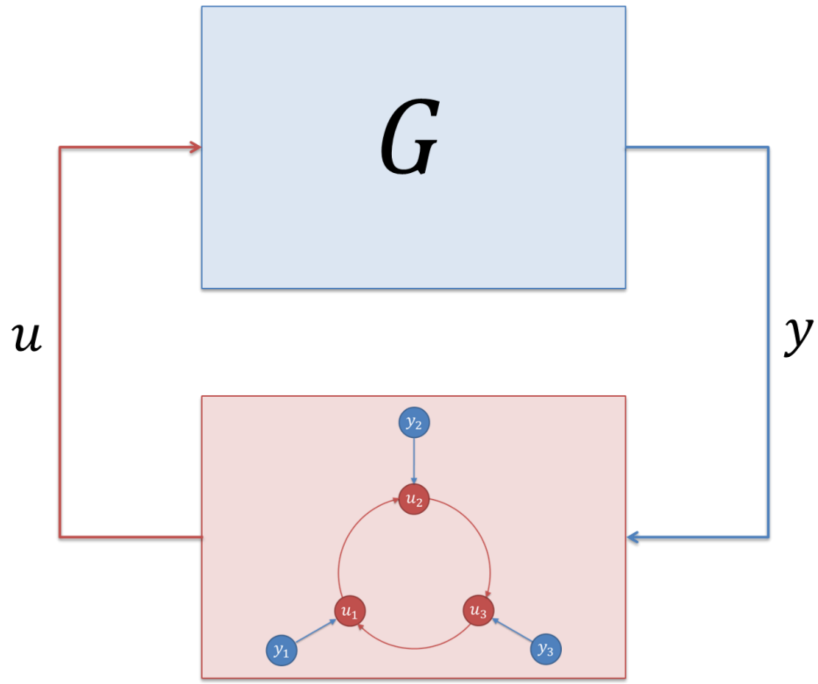
Closed-Loop Problem Formulation

- English: Given two systems in feedback, design the structure of one to minimize the vulnerability of the combined system.
- Math: Given fixed TFs G and K, design structure (P, Q) of K such that the system vulnerability is minimized.

$$\min_{Q} \left\| \begin{bmatrix} G(I - KG)^{-1} \\ (I - KG)^{-1} \end{bmatrix} (I - Q)^{-1} \right\|_{1-\infty}$$

Result 1: Decoupling of Vulnerability

- *Theorem 3:* Vulnerabilities on links in one system do not depend on the structure of the other system.
  - Only on other system's "black box" behavior
  - Does depend on its own structure



Result 2: We can Fight Fire with Fire

- We know that cycles create vulnerability
- When feedback is necessary, it is possible to use cycles within systems to reduce the vulnerability of the combined system
- There may be a “universal structure” of Q that uses cycles to minimize vulnerability, independent of G and K.

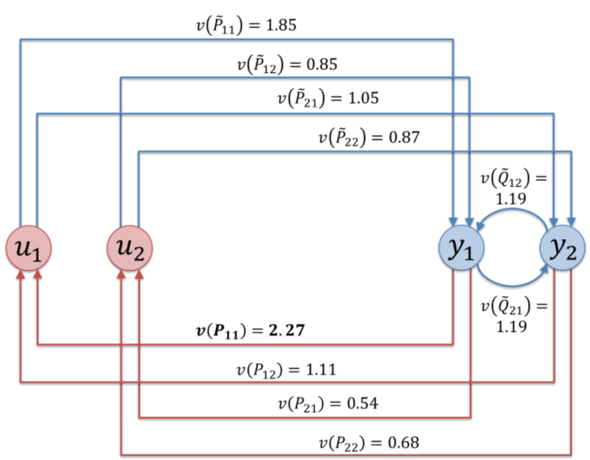
Examples

- Let $G = \begin{bmatrix} \frac{2}{s-1} & \frac{1}{s-1} \\ \frac{1}{s-1} & \frac{2}{s-1} \end{bmatrix}$,
- Let $K = \frac{1}{(s+1)(s+3)} \begin{bmatrix} -3s-4 & -2s-1 \\ -2s-1 & -3s-4 \end{bmatrix}$

Example 1: Fight Fire with Fire

Empty Q

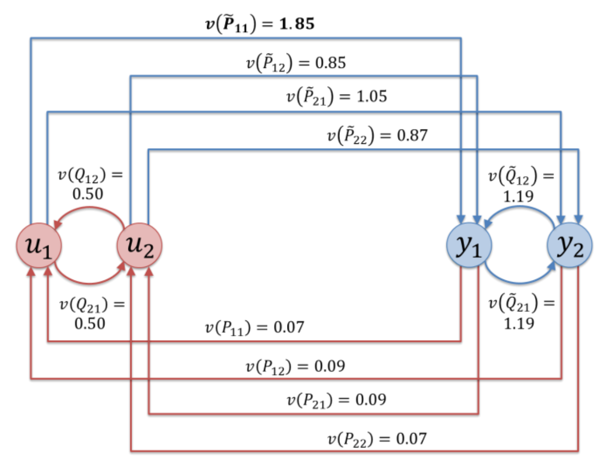
- $Q = 0, P = K.$
- Max Vulnerability = 2.27



Max Vulnerability = 2.27

A Q with Internal Feedback

- $Q = \frac{1}{s+1}\begin{bmatrix} 0 & 32 \\ 32 & 0 \end{bmatrix}$
- $P = \frac{1}{f(s)}\begin{bmatrix} -3s^2 + 57s + 28 & -2s^2 + 93s + 127 \\ -2s^2 + 93s + 127 & -3s^2 + 57s + 28 \end{bmatrix},$
  $f(s) = (s+1)^2(s+3)$

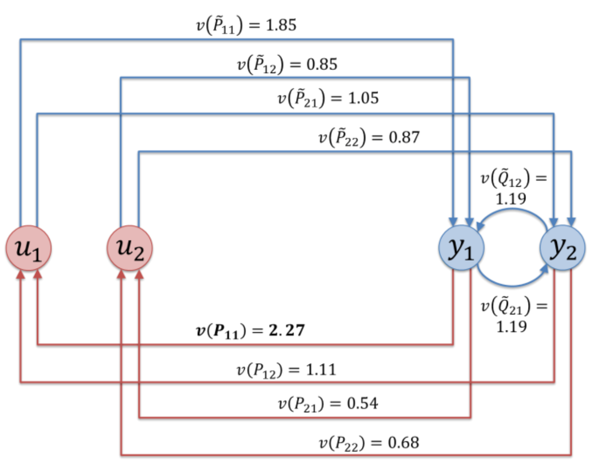


Max Vulnerability = 1.85

Example 2: A Word of Caution

Empty Q

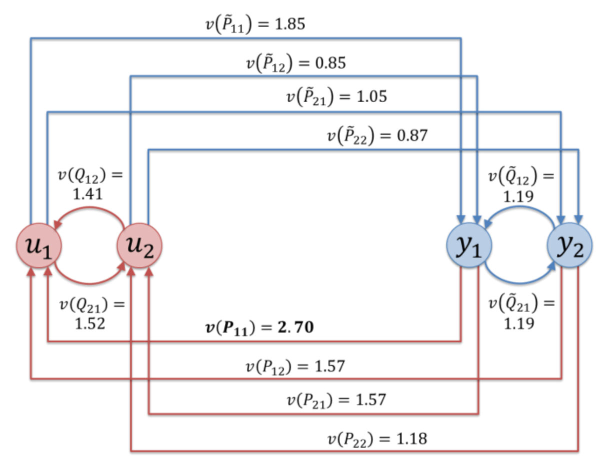
- $Q = 0, P = K.$
- Max Vulnerability = 2.27



Max Vulnerability = 2.27

A Q with Internal Feedback

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



Max Vulnerability = 2.70

The High-Gain Heuristic (Universal Structure)

- We don't yet know how to choose Q to minimize the vulnerability of the combined system.
  - But we have a good idea
- Let

$$Q = \begin{bmatrix} 0 & \frac{n}{p(s)} & \cdots & \frac{n}{p(s)} \\ \frac{n}{p(s)} & 0 & & \frac{n}{p(s)} \\ \vdots & & \ddots & \vdots \\ \frac{n}{p(s)} & \frac{n}{p(s)} & \cdots & 0 \end{bmatrix}$$

- In all of our tests, when $n \in \mathbb{R}$ grows large:
  - The vulnerabilities on the links in P approach 0
  - The vulnerabilities on the links in Q approach $\frac{1}{rows(Q)}$
- Internal stability may be an issue

Example 3: High Gain Heuristic

- $Q = \frac{1}{s+1}\begin{bmatrix} 0 & 10000 \\ 10000 & 0 \end{bmatrix}$

  $P = \frac{1}{f(s)}\begin{bmatrix} g(s) & h(s) \\ h(s) & g(s) \end{bmatrix}$

  $f(s) = (s+1)^2(s+3)$

  $g(s) = -3s^2 + 19993s + 9996$

  $h(s) = -2s^2 + 29997s + 39999$

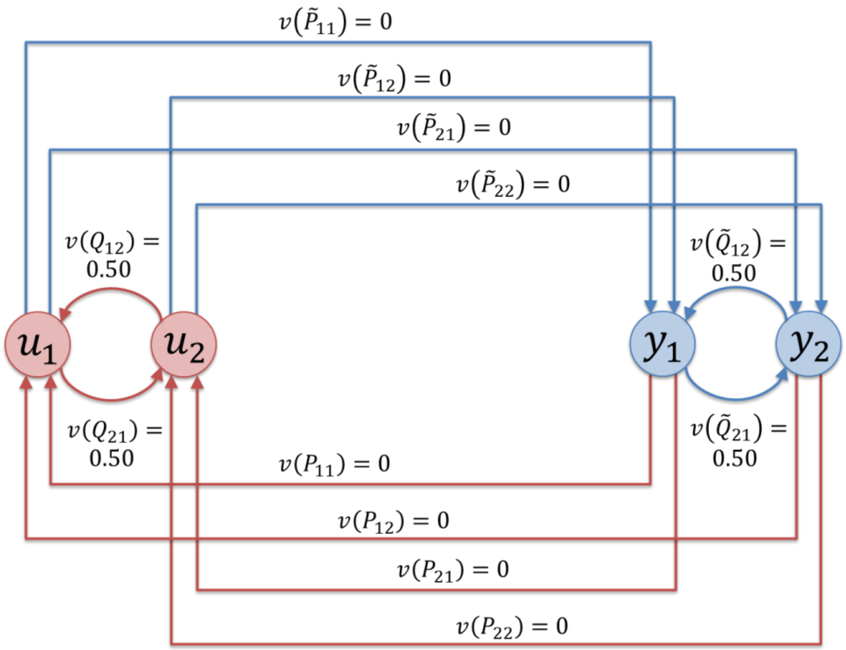
- $\tilde{Q} = \frac{1}{s-1}\begin{bmatrix} 0 & 10000 \\ 10000 & 0 \end{bmatrix}$

  $\tilde{P} = \frac{1}{f(s)}\begin{bmatrix} g(s) & h(s) \\ h(s) & g(s) \end{bmatrix}$

  $f(s) = (s+1)^2(s+3)$

  $g(s) = 2s^2 - 10000s + 9998$

  $h(s) = s^2 - 20002s + 19999$



Introduction: Vulnerability

Lesson: Vulnerability is a Property of Links

Preliminaries: Systems and Structure

Lesson: Definition of Link Depends on Structure, which depends on Implementation

Vulnerability in Open-Loop Systems

Lesson: Links in cycles are vulnerable

To remove vulnerability, remove cycles

Vulnerability in Closed-Loop Systems

Lesson: Can use cycles to minimize vulnerability caused by feedback

CONCLUSIONS

Next Steps

- Is there a universal structure?
- If there is a universal structure, is it the high-gain heuristic?
- If not, how do we design Q to minimize vulnerability?
- What other characteristics of systems should we explore (maintainability, adaptability, cost)?

QUESTIONS?

APPENDICES

Derivation of a DSF

- Consider a state-space LTI system
$$\begin{bmatrix}\dot{z}_1\\ \dot{z}_2\end{bmatrix} = \begin{bmatrix}\bar{A}_{11} & \bar{A}_{12}\\ \bar{A}_{21} & \bar{A}_{22}\end{bmatrix}\begin{bmatrix}z_1\\ z_2\end{bmatrix} + \begin{bmatrix}\bar{B}_1\\ \bar{B}_2\end{bmatrix}u$$
$$y = [\bar{C}_1 \quad \bar{C}_2]\begin{bmatrix}z_1\\ z_2\end{bmatrix},$$
where $[\bar{C}_1 \quad \bar{C}_2]$ has full row rank.
- The system can be transformed to
$$\begin{bmatrix}\dot{y}\\ \dot{x}\end{bmatrix} = \begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}\begin{bmatrix}y\\ x\end{bmatrix} + \begin{bmatrix}B_1\\ B_2\end{bmatrix}u$$
$$y = [I \quad 0]\begin{bmatrix}y\\ x\end{bmatrix},$$
where y are the states that are measured.
- Taking the Laplace transform, we get
$$\begin{bmatrix}sY\\ sX\end{bmatrix} = \begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}\begin{bmatrix}Y\\ X\end{bmatrix} + \begin{bmatrix}B_1\\ B_2\end{bmatrix}U$$
- Solving for X we get
$$X = (sI - A_{22})^{-1}A_{21}Y + (sI - A_{22})^{-1}B_2U,$$
which yields
$$sY = WY + VU$$
$$W = A_{11} + A_{12}(sI - A_{22})^{-1}A_{21}$$
$$V = A_{12}(sI - A_{22})^{-1}B_2 + B_1.$$
- Let D be a diagonal matrix with the diagonal entries of W. Then
$$(sI - D)Y = (W - D)Y + VU.$$
Therefore,
$$Y = QY + PU$$
where
$$Q = (sI - D)^{-1}(W - D)$$
$$P = (sI - D)^{-1}V$$
- It can be checked that
$$G = (I - Q)^{-1}P = C(sI - A)^{-1}B.$$

Vulnerability of Links in a DSF

- Given a DSF (P, Q) and $H = (I - Q)^{-1}$, the vulnerability of a link (i, j) in Q is

$$v(q_{ij}) = \|h_{ji}\|_{\infty}$$

- The vulnerability of the system is

$$V = \max_{(i,j)\in Q} \left(v(q_{ij}) \right) = \|h_{ji}\|_{1-\infty}$$

Proof of Theorem 1

- A system with a stable additive transformation on the link from node i to j can be represented as the linear fractional transformation shown to the right in Figure 1.1
- T is the associated closed-loop transfer function
- w_i and w_j represent signals at nodes i and j

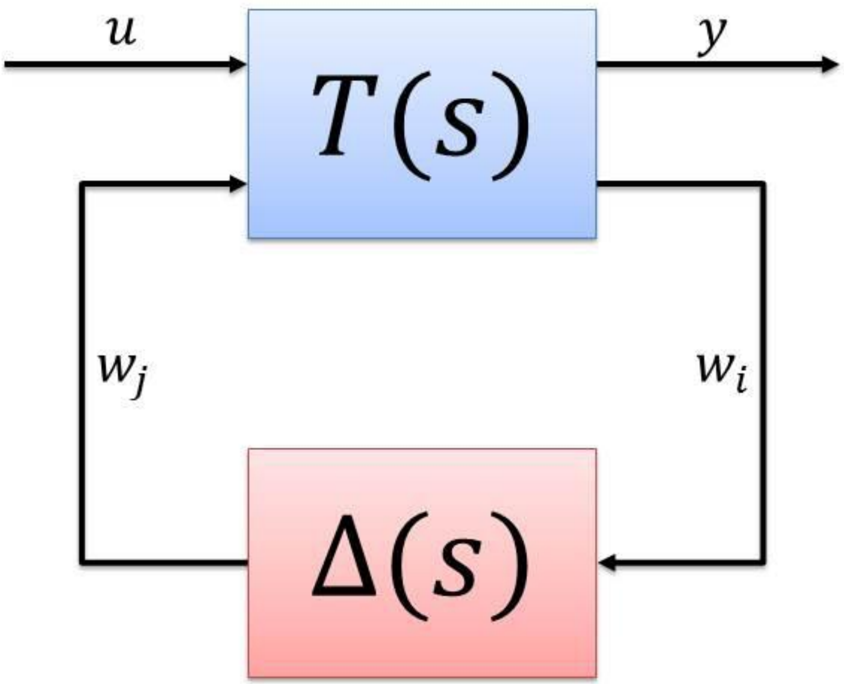


Figure 1.1

Proof of Theorem 1

- Let T_{ij} be the closed-loop transfer function from j to i
- Then the system represented in figure 1.1. is stable if and only if the system represented by the figure 1.2 to the right is stable

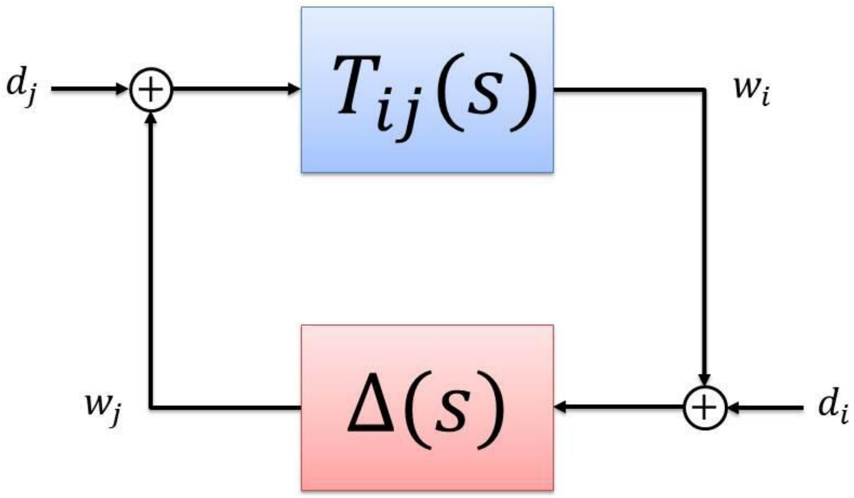


Figure 1.2

Proof of Theorem 1

- Assume $T_{ij}(s) = 0$
- Then the system in Figure 1.2 is comprised only of the feed-forward term $\Delta(s)$ and is stable for all stable perturbations $\Delta(s)$.
- Hence the link from i to j is not vulnerable.

Proof of Theorem 1

- Assume $T_{ij}(s) \neq 0$
- Then the system in Figure 1.2 is unstable if any of the transfer functions $\begin{bmatrix} d_j \\ d_i \end{bmatrix} \rightarrow \begin{bmatrix} w_j \\ w_i \end{bmatrix}$ are unstable.
- We have

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

- Let $T_{ij}(s) = \frac{t_n(s)}{t_d(s)}$ and $\Delta(s) = \frac{\delta_n(s)}{\delta_d(s)}$ (each being polynomials in s)
- Then

$$w_j = \frac{t_d(s)\delta_d(s)}{t_d(s)\delta_{d(s)} - t_n(s)\delta_n(s)} \begin{bmatrix} \frac{t_n(s)\delta_n(s)}{t_d(s)\delta_d(s)} & \frac{\delta_n(s)}{\delta_d(s)} \end{bmatrix} \begin{bmatrix} d_j \\ d_i \end{bmatrix}$$

- According to the Routh-Hurwitz Stability Criterion, $t_d(s)\delta_{d(s)} - t_n(s)\delta_n(s)$ is stable if all coefficients are of the same sign.
- A properly designed Δ can zero out at least one of these terms; hence the Routh-Hurwitz Stability Criterion fails.
- Therefore there exists a stable Δ on the link from i to j that destabilizes the system and the link is vulnerable.

Proof of Theorem 2

- It is sufficient to let $Q(s) = 0$ and $P(s) = G(s)$.
- All links are in $P(s)$, and by design, no link in $P(s)$ is in a cycle; therefore by Theorem 1, all links are completely secure.

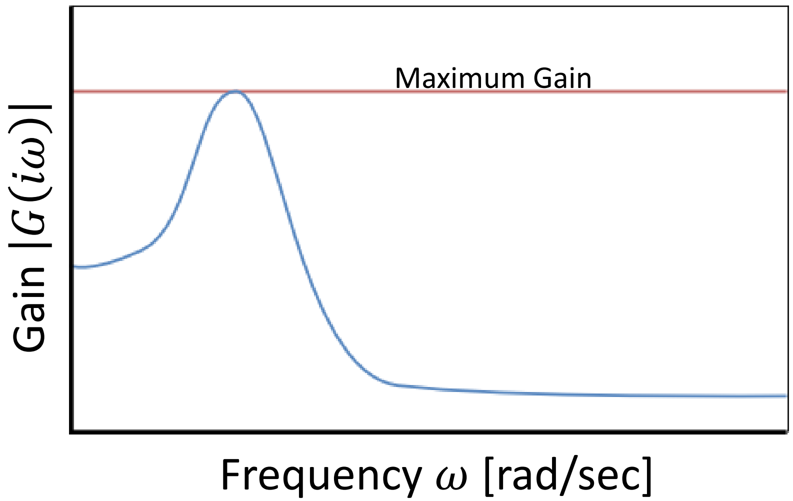
Proof of Theorem 3

- The inverse of H is defined such that $\begin{bmatrix} I-\tilde{Q} & -\tilde{P} \\ -P & I-Q \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$.
  - $(I-\tilde{Q})B - PD = 0$, therefore $B = (I-\tilde{Q})^{-1}PD = GD$
  - $(I-Q)C - PA = 0$, therefore $C = (I-Q)^{-1}PA = KA$
  - $(I-\tilde{Q})A - \tilde{P}C = (I-\tilde{Q})A - \tilde{P}KA = I$, therefore $A = (I-\tilde{Q}-\tilde{P}K)^{-1}$
  - $(I-Q)D - PB = (I-Q)D - PGD = I$, therefore $D = (I-Q-PG)^{-1}$
  - Thus $(I-Q)^{-1} = \begin{bmatrix} (I-\tilde{Q}-\tilde{P}K)^{-1} & G(I-Q-PG)^{-1} \\ K(I-\tilde{Q}-\tilde{P}K)^{-1} & (I-Q-PG)^{-1} \end{bmatrix}$
- Note that all links in the controller are represented in the bottom rows of $\hat{Q}$. Since the vulnerability any link (i,j) in the combined system are defined by the h_∞ norm of entry (j,i) in $H = (I-\hat{Q})^{-1}$, the vulnerability of the links in the controller are contained entirely in the equations in the right column of H given above and are expressed only in terms of P, Q, and G.
- Therefore, the vulnerability of the links in the controller are independent of the structure $(\tilde{P},\tilde{Q})$of the links in the plant.
- Note that similarly, the vulnerability of the links in the plant are independent of the structure (P,Q) of the links in the controller.

IDeA Labs
Information & Decision Algorithms Laboratories

The One-Infinity Norm

- The problem formulation of both open-loop and closed-loop systems involve
  $$\min_{Q} \|X\|_{1-\infty},$$
  where X is a matrix of rational functions of form $\frac{p(s)}{q(s)}$.
- The infinity norm computes the maximum gain seen by each entry of X (see figure to the right)
  - Corresponds to the size minimum signal required to destabilize the system.
- The one norm chooses the largest of the computed infinity norms.
- Therefore the one-infinity norm computes the vulnerability of the system, which we wish to minimize by choosing a good Q.



References

- A. Rai, D. Ward, S. Roy and S. Warnick, "Vulnerable Links and Secure Architectures in the Stabilization of Networks of Controlled Dynamical Systems," *American Control Conference*, Montreal, Canada, accepted for publication, 2012.
- A. Rai, "Analysis and Design Tools for Structured Feedback Systems," M.S. Thesis, Brigham Young Univ., Provo, UT, 2012.