System Interaction Complexity Metrics and The Application to Embedded Software Systems

Qi Van Eikema Hommes, Ph.D.
Research Scientist
Engineering Systems Division
Massachusetts Institute of Technology
qhommes@mit.edu

MITsdm system design and management

Dr. Qi Van Eikema Hommes

- Research Scientist and Lecturer (MIT Engineering Systems Division)
  - Lead Instructor for ESD.33 Systems Engineering
  - Lead Instructor for ESD.40--PDD in Spring 2010
  - Co-instructor for the PDD class in the MIT Portugal program
- 8 years of work experiences in US automotive companies (Ford and GM)
  - Senior Research Scientist at GM R&D
  - Powertrain Systems Engineer at Ford

Ph.D. and M.S. in Mechanical Engineering

B.S. in Mechanical Engineering

Presentation Outline

- Research Motivation
- Networks and Its Matrix Representation
- Review and Compare Complexity Measures
- Application to Vehicle Embedded Software Systems
- Summary

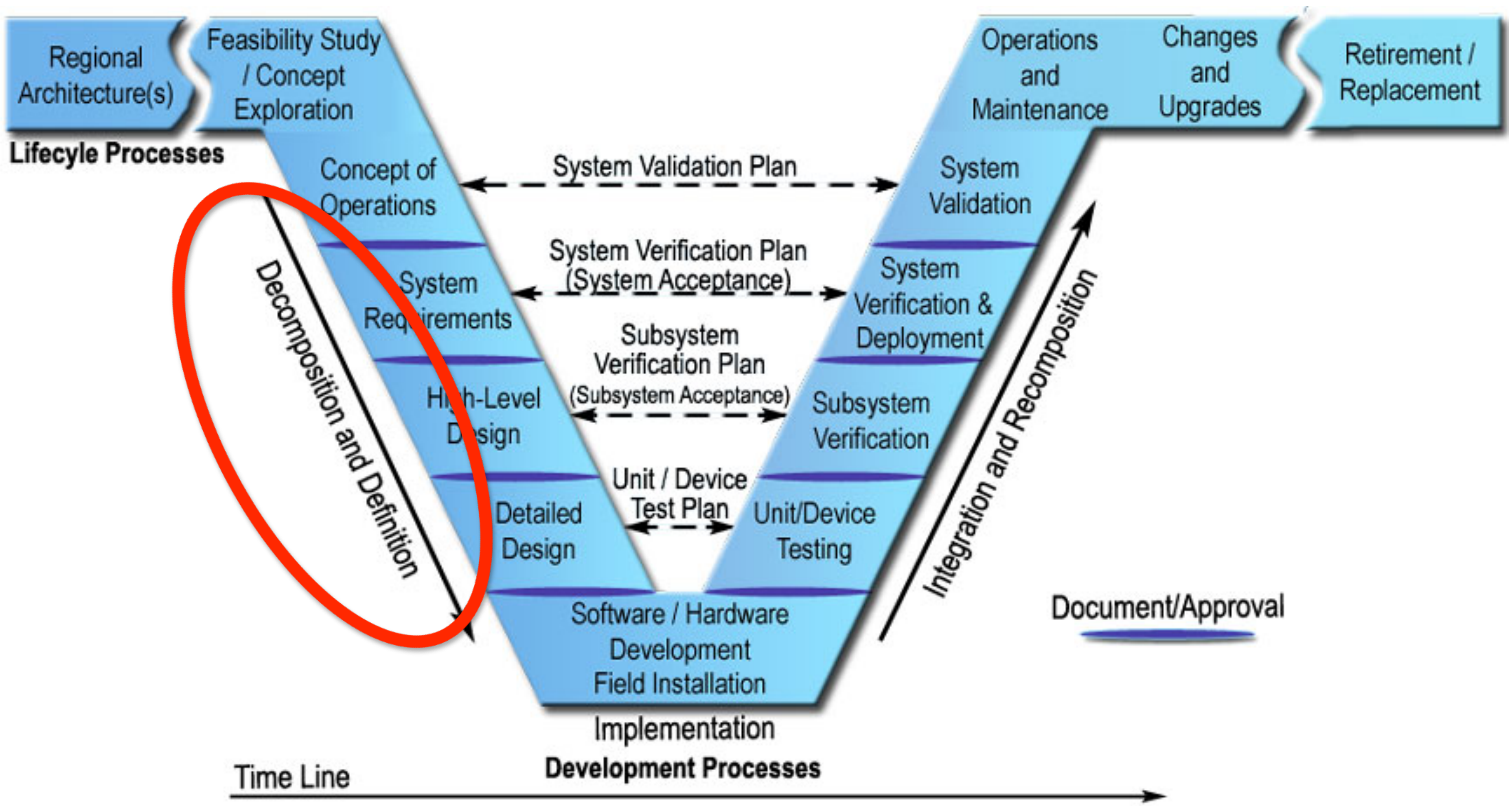
Three Types of System Complexity

Randomness	High	Unorganized Complexity (Can Use Statistics)	
	Low	Organized Simplicity (Can Use Analytic Deduction)	**Organized Complexity (Systems)**
		Low	High
		Complexity	

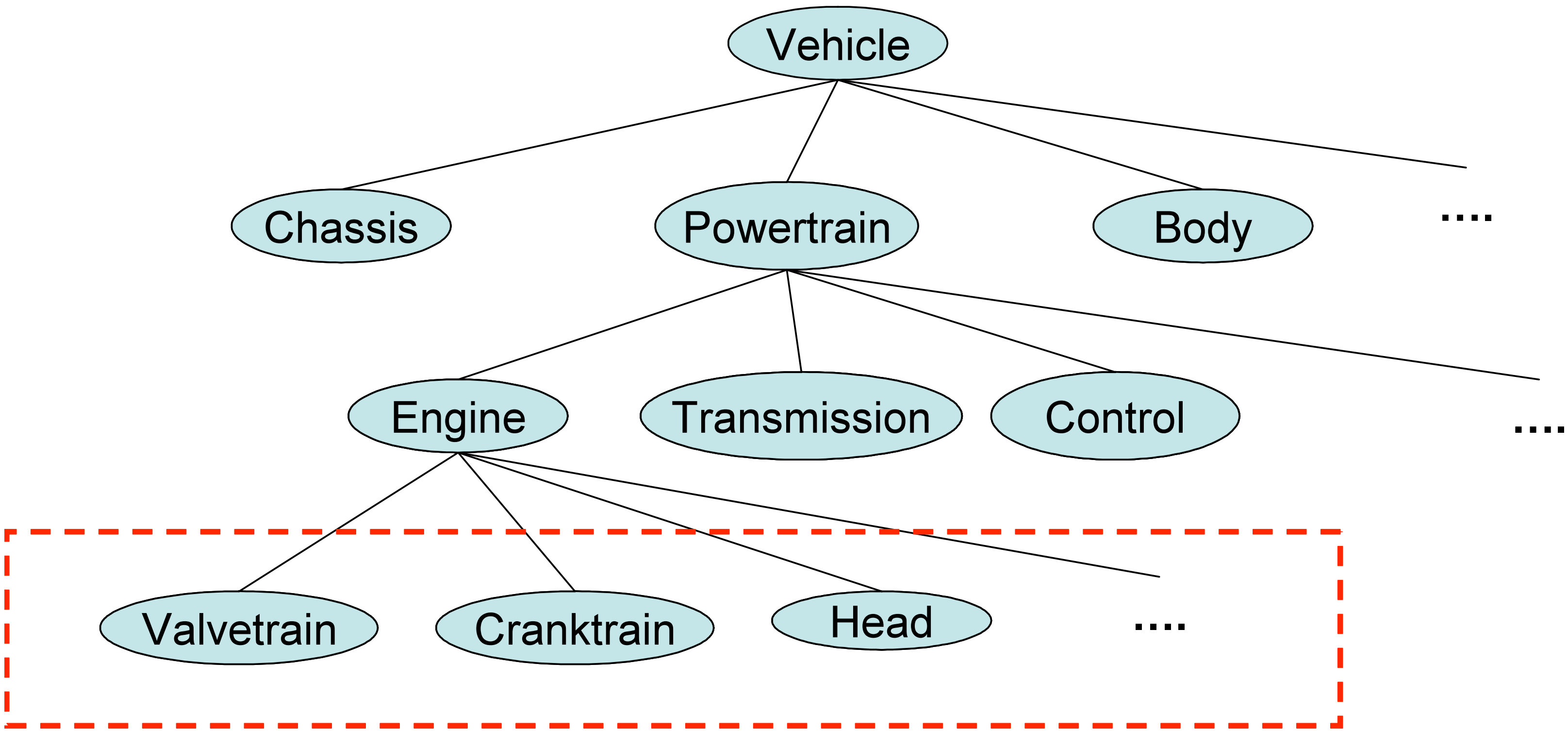
Adapted from Weinberg 1975.

Systems Engineering Decomposition

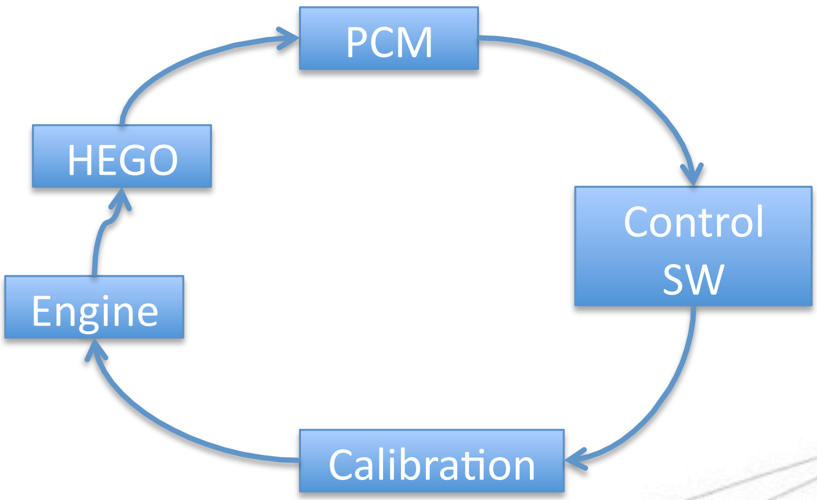
http://ops.fhwa.dot.gov/publications/seitsguide/section3.htm

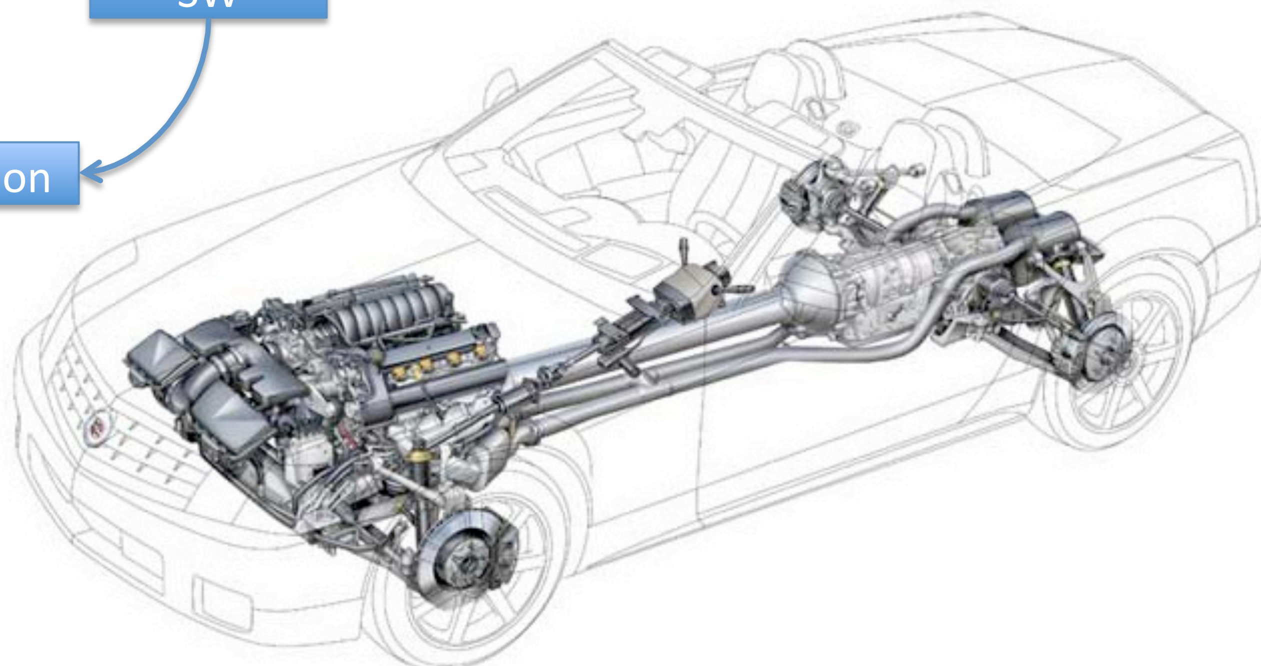


Example of Physical Decomposition



Does Decomposition Guarantee Success?





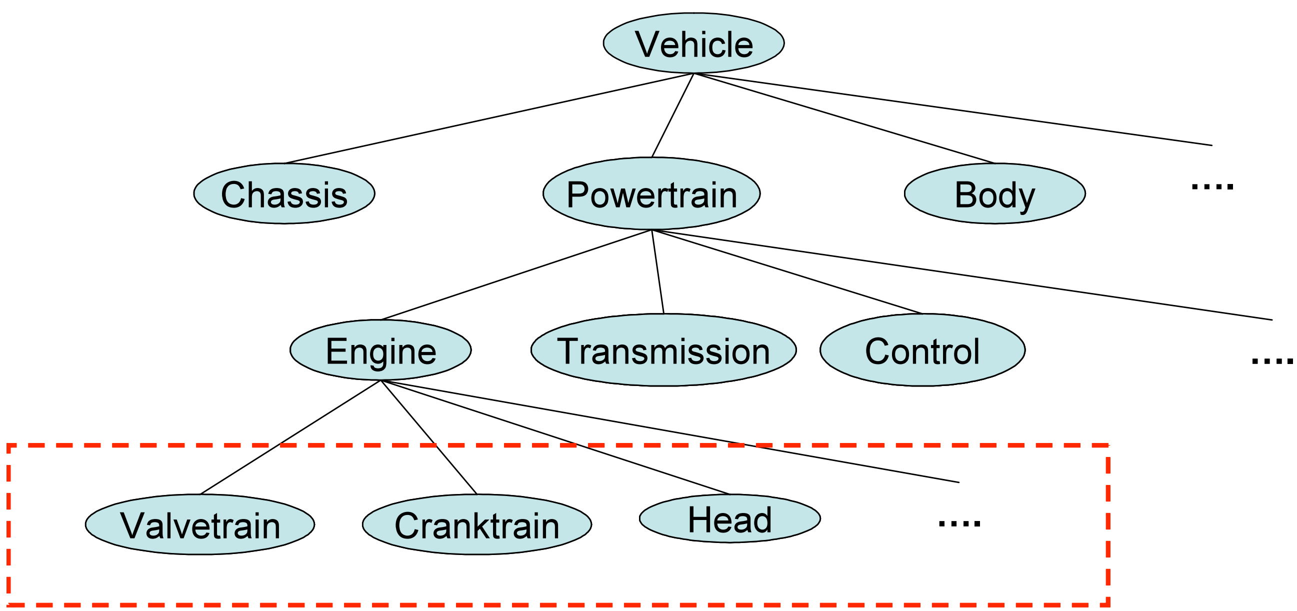
http://johnwrenolds.com/images/powertrain.jpg

Emerging Properties of a Systems

- The whole is not equal to the sum of its parts.
  - Sometimes more
  - Sometimes less
- Complex behavior of the system arise because of the interactions between the components of a system.
- Today's complex systems have very large numbers of interactions that are beyond the human comprehension.
- More couplings ➔ more system interactions ➔more emergent behaviors (some are unanticipated and undesirable)

System Interaction Complexity Metrics

- Metrics that can quantify the extend of component interactions in a system, at a specified level of the system hierarchy.

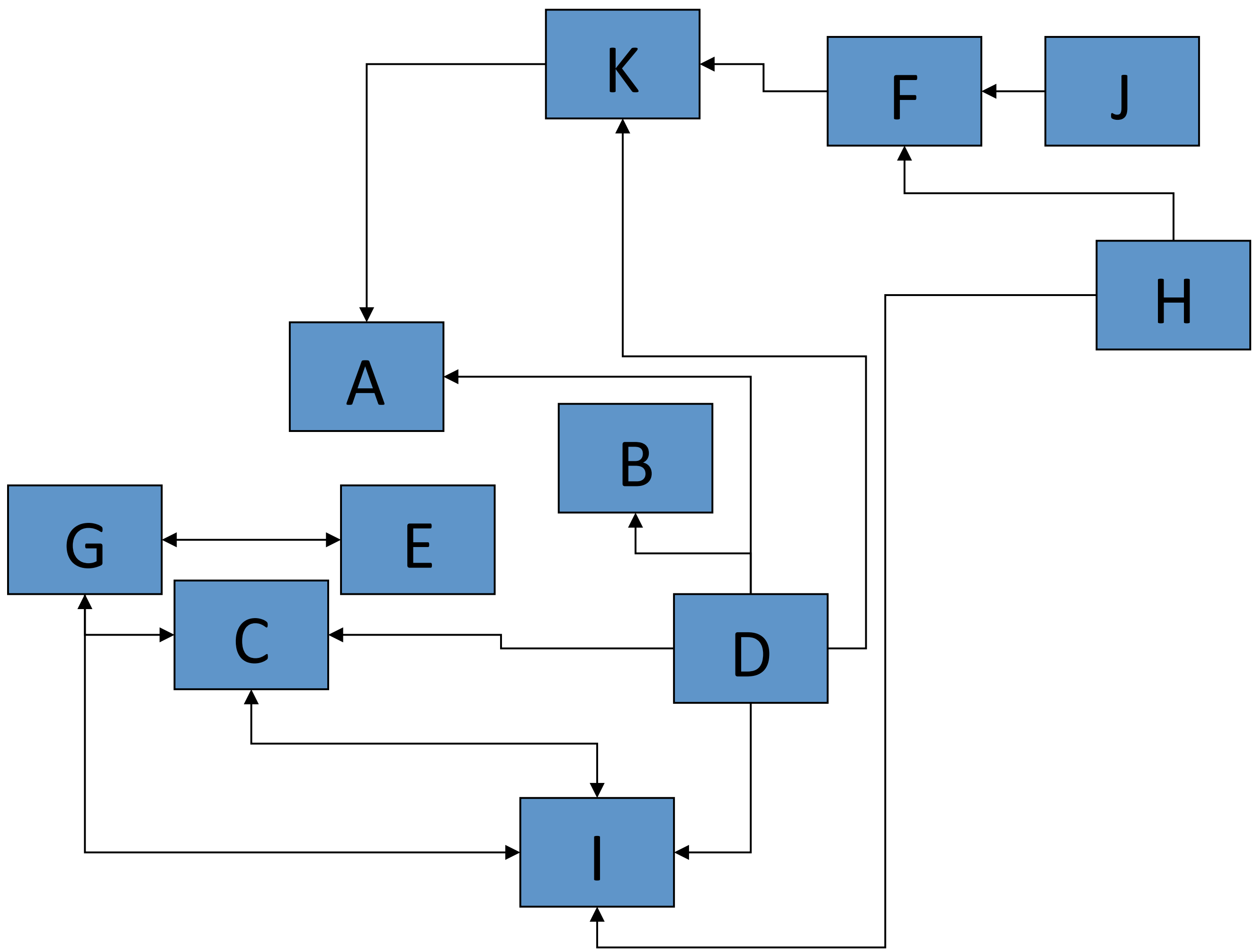


Presentation Outline

- ✓ Research Motivation
- ❑ Networks and Its Matrix Representation
- ❑ Review and Compare Complexity Measures
- ❑ Application to Vehicle Embedded Software Systems
- ❑ Summary

Qi Van Eikema Hommes,
qhommes@mit.edu

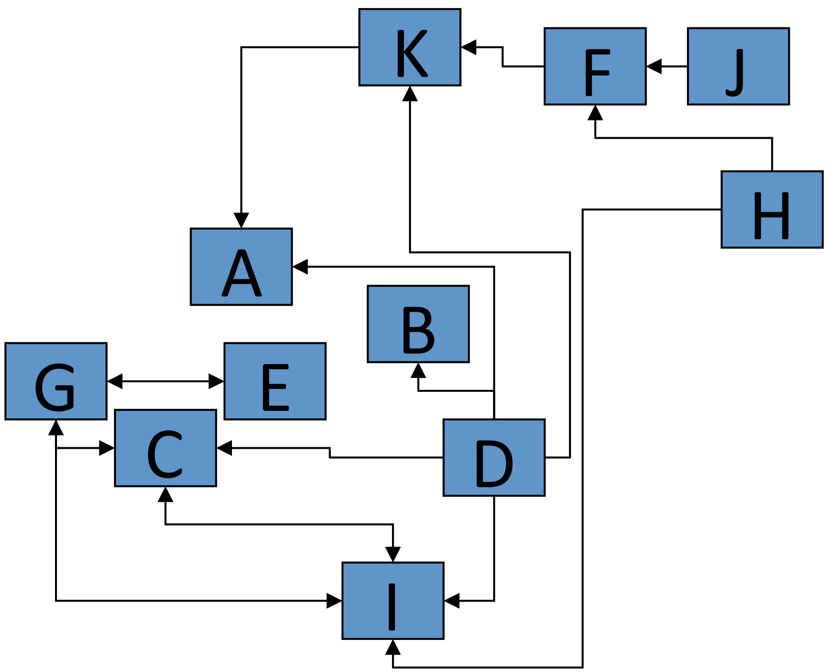
Components Interactions Represented in a Network Diagram



Matrix Form of the Network Diagram

- Social Network Analysis—Affiliation Matrix
- Mathematics and Computer Science—Adjacency Matrix
- Product Design and Development—Design Structure Matrix

Matrix Representation of a Network



	A	B	C	D	E	F	G	H	I	J	K
A				1							1
B				1							
C				1			1		1		
D											
E							1				
F								1		1	
G			1		1						
H											
I			1	1			1	1			
J											
K				1		1					

Presentation Outline

- ✓ Research Motivation
- ✓ Networks and Its Matrix Representation
- ❑ Review and Compare Complexity Measures
- ❑ Application to Vehicle Embedded Software Systems
- ❑ Summary

Literature Survey

1. **Whitney Index** (Whitney, Dong, Judson, Mascoli 1999)
2. **Change Cost** (MacCormack, Rusnak, Baldwin 2004)
3. Singular value Modularity Index (Holtta, de Weck, 2005)
4. Visibility-dependence signature plot (Sharman, Yassine 2004)
5. **Network Centrality** (Sosa, Eppinger, Rowles 2007, Borgatti, Everett, and Freeman, 2002, UCINET)
   – Degree modularity
   – Distance modularity
   – Bridge modularity
6. Software Complexity Metrics—because the case study is embedded software system.

Test Matrices

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1					1
D			1	1			1	
E				1	1			
F					1	1		
G				1		1	1	
H							1	1

Matrix1

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1					1
D			1	1			1	
E				1	1			
F					1	1		
G	1			1		1	1	
H							1	1

Matrix2

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1	1	1	1	1	1
D			1	1	1	1	1	1
E			1	1	1	1	1	1
F			1	1	1	1	1	1
G			1	1	1	1	1	1
H			1	1	1	1	1	1

Matrix3

	A	B	C	D	E	F	G	H
A	1	1	1	1				
B	1	1	1	1				
C	1	1	1	1				
D	1	1	1	1				
E					1	1	1	1
F					1	1	1	1
G					1	1	1	1
H					1	1	1	1

Matrix4

	A	B	C	D	E	F	G	H
A	1	1	1	1	1	1	1	1
B	1	1						
C	1		1					
D	1			1				
E	1				1			
F	1					1		
G	1						1	
H	1							1

Matrix5

	A	B	C	D	E	F	G	H
A	1							
B	1	1						
C		1	1					
D			1	1				
E				1	1			
F					1	1		
G						1	1	
H							1	1

Matrix6

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1	1					
C		1	1	1				
D			1	1	1			
E				1	1	1		
F					1	1	1	
G						1	1	1
H							1	1

Matrix7

	A	B	C	D	E	F	G	H
A	1	1						1
B	1	1	1					
C		1	1	1				
D			1	1	1			
E				1	1	1		
F					1	1	1	
G						1	1	1
H	1						1	1

Matrix8

Whitney Index (WI)

(Whitney, Dong, Judson, Mascoli 1999)

$$WI = \frac{\#\_of\_Interactions\_in\_a\_DSM}{\#\_of\_elements\_in\_a\_DSM}$$

- WI reflects the density of path length=1 connectivity.
- WI does not capture the ripple effects of the system interactions.

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1	1	1	1	1	1
D			1	1	1	1	1	1
E			1	1	1	1	1	1
F			1	1	1	1	1	1
G			1	1	1	1	1	1
H			1	1	1	1	1	1

Matrix3

WI=5

	A	B	C	D	E	F	G	H
A	1	1						1
B	1	1	1					
C		1	1	1				
D			1	1	1			
E				1	1	1		
F					1	1	1	
G						1	1	1
H	1						1	1

Matrix8

WI=3

Change Cost (CC)

(MacCormack, Rusnak, Baldwin 2004)

M^0

	A	B	C	D	E	F
A	**1**	0	0	0	0	0
B	0	**1**	0	0	0	0
C	0	0	**1**	0	0	0
D	0	0	0	**1**	0	0
E	0	0	0	0	**1**	0
F	0	0	0	0	0	**1**

M^1

	A	B	C	D	E	F
A	0	**1**	**1**	0	0	0
B	0	0	0	**1**	0	0
C	0	0	0	0	**1**	0
D	0	0	0	0	0	0
E	0	0	0	0	0	**1**
F	0	0	0	0	0	0

M^2

	A	B	C	D	E	F
A	0	0	0	**1**	**1**	0
B	0	0	0	0	0	0
C	0	0	0	0	0	**1**
D	0	0	0	0	0	0
E	0	0	0	0	0	0
F	0	0	0	0	0	0

M^3

	A	B	C	D	E	F
A	0	0	0	0	0	**1**
B	0	0	0	0	0	0
C	0	0	0	0	0	0
D	0	0	0	0	0	0
E	0	0	0	0	0	0
F	0	0	0	0	0	0

M^4

	A	B	C	D	E	F
A	0	0	0	0	0	0
B	0	0	0	0	0	0
C	0	0	0	0	0	0
D	0	0	0	0	0	0
E	0	0	0	0	0	0
F	0	0	0	0	0	0

$V = \Sigma M^n \; ; \; n = [0,4]$

	A	B	C	D	E	F
A	**1**	**1**	**1**	**1**	**1**	**1**
B	0	**1**	0	**1**	0	0
C	0	0	**1**	0	**1**	**1**
D	0	0	0	**1**	0	0
E	0	0	0	0	**1**	**1**
F	0	0	0	0	0	**1**

$$CC = \frac{Average\_of\_the\_Sum\_for\_Rows\_in\_V}{\#\_of\_System\_Elements}$$

- CC captures the average ripple effects in the system when a change happens.
- CC does not reflect the density of direct interactions.

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1	1	1	1	1	1
D			1	1	1	1	1	1
E			1	1	1	1	1	1
F			1	1	1	1	1	1
G			1	1	1	1	1	1
H			1	1	1	1	1	1

Matrix3

WI=5
CC=63%

	A	B	C	D	E	F	G	H
A	1	1						1
B	1	1	1					
C		1	1	1				
D			1	1	1			
E				1	1	1		
F					1	1	1	
G						1	1	1
H	1						1	1

Matrix8

WI=3
CC=100%

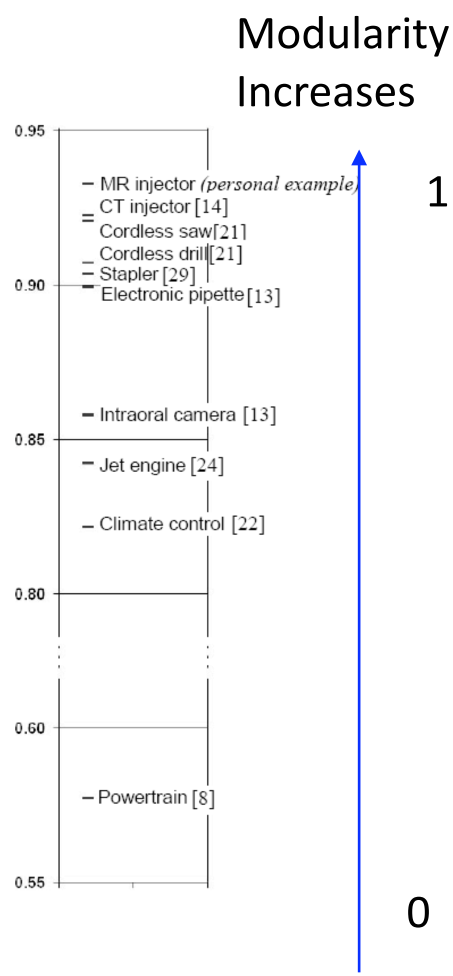
Singular Value Modularity Index (SMI)

(Holtta, et al. 2005)

$$DSM = U \cdot \Sigma_{DSM} \cdot V^T$$

$$SMI = \frac{1}{N} \arg\min_{\alpha} \sum_{i=1}^{N} \left| \frac{\sigma_i}{\sigma_1} - e^{-[i-1]/\alpha} \right|$$

- SMI values have produced arguable results in the test DSM cases.
- SMI values are not reliable measures of system modularity.



	A	B	C	D	E	F	G	H
A	1	1						1
B	1	1	1					
C		1	1	1				
D			1	1	1			
E				1	1	1		
F					1	1	1	
G						1	1	1
H	1						1	1

Matrix8

SMI=0.8

	A	B	C	D	E	F	G	H
A	1	1	1	1				
B	1	1	1	1				
C	1	1	1	1				
D	1	1	1	1				
E					1	1	1	1
F					1	1	1	1
G					1	1	1	1
H					1	1	1	1

Matrix4

SMI=0.1

Visibility Dependence Signature Plot

(Sharman, Yassine 2004)

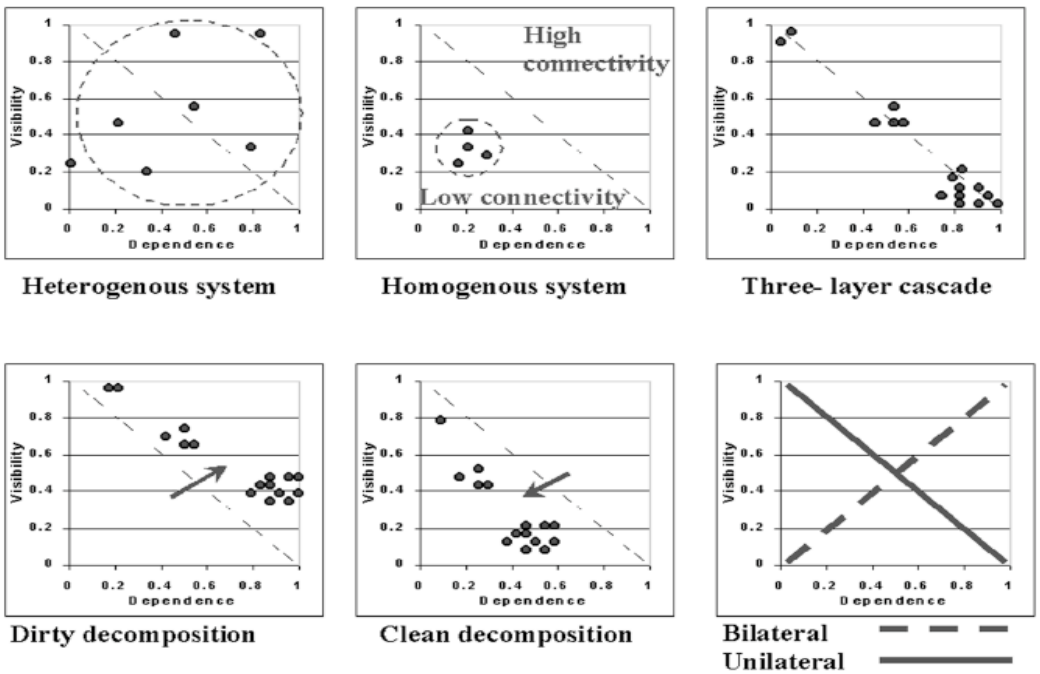


Figure 16. Characteristic types of visibility-dependence signatures.

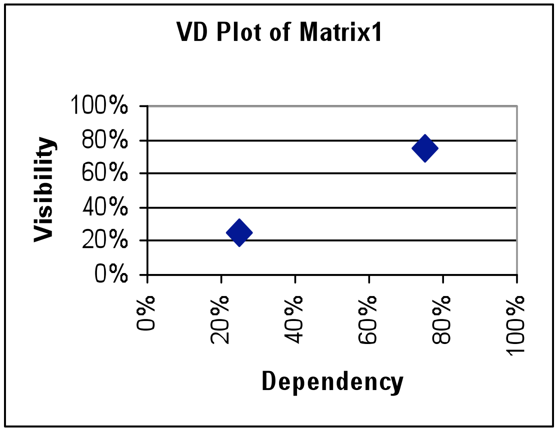
Visibility=average of column of the visibility matrix

Dependence=average of row of the visibility matrix

VD Plot Test Case Results

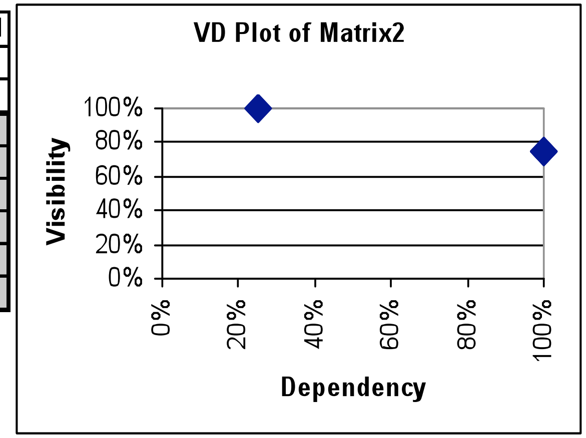
	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1					1
D			1	1			1	
E				1	1			
F					1	1		
G				1		1	1	
H							1	1

Matrix1



	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1					1
D			1	1			1	
E				1	1			
F					1	1		
G	1			1		1	1	
H							1	1

Matrix2



- **VD plots did not produce conclusions that are consistent with the system characteristics.**
- Unilateral means the system can be re-sequenced to force relationships below diagonal. Matrix 1 and 2 are different by only 1 entry (G,A). Why is Matrix 1 bilateral and 2 is unilateral?
- Matrix 1 and 2 are very clean decomposition. But based on VD plots, it was hard to tell which one is clean decomposition.
- **The computation to get VD plots are very similar to CC Index, and the plotting effort to produce these VD plots are cumbersome.**

Network Centrality

(Sosa, Eppinger, Rowles 2007, Borgatti, Everett, and Freeman, 2002, UCINET)

- Network centrality metrics can identify the few elements that have the largest impact on the system.
- If the network has central players, the network may be bus-modular.
- If the network does not have central player, the network system is either not connected, or highly integral.
- Central players can be the priority for system complexity reduction strategy.

Network Centrality—Degree Centrality

(Sosa, Eppinger, Rowles 2007, Borgatti, Everett, and Freeman, 2002, UCINET)

In degree—how many others pass information to the element of interest.
Out degree—how many others depend on the element of interest for information.

Degree Centrality identifies which few elements, if any, in the system have a central effect on the rest of the systems.

By its definition, the overall system centrality measures doesn't correlate well with components interaction complexity.

	A	B	C	D	E	F	G	H	Freeman Centrality in degree	Freeman Centrality out degree	Freeman Centrality Overall in	Freeman Centrality Overall out
A	1	1	1	1					3	3	0%	0%
B	1	1	1	1					3	3		
C	1	1	1	1					3	3		
D	1	1	1	1					3	3		
E					1	1	1	1	3	3		
F					1	1	1	1	3	3		
G					1	1	1	1	3	3		
H					1	1	1	1	3	3		

Matrix4

	A	B	C	D	E	F	G	H	Freeman Centrality in degree	Freeman Centrality out degree	Freeman Centrality Overall in	Freeman Centrality Overall out
A	1	1	1	1	1	1	1	1	7	7	85.70%	85.70%
B	1	1							1	1		
C	1		1						1	1		
D	1			1					1	1		
E	1				1				1	1		
F	1					1			1	1		
G	1						1		1	1		
H	1							1	1	1		

Matrix5

	A	B	C	D	E	F	G	H	Freeman Centrality in degree	Freeman Centrality out degree	Freeman Centrality Overall in	Freeman Centrality Overall out
A	1	1						1	2	2	0%	0%
B	1	1	1						2	2		
C		1	1	1					2	2		
D			1	1	1				2	2		
E				1	1	1			2	2		
F					1	1	1		2	2		
G						1	1	1	2	2		
H	1						1	1	2	2		

Matrix8

Network Centrality—Freeman Farness/Closeness

(Sosa, Eppinger, Rowles 2007, Borgatti, Everett, and Freeman, 2002, UCINET)

Freeman distance measures—ratio between the sum of the incoming/outgoing geodesics of the component of interest with all other components in the product and the maximum geodesics a component can have in the network.

Past literature (Sosa et al. 2007) suggests that a high value of freeman distance measure indicates the component is far from others and hence more modular.

However, a overall system distance metric cannot be computed if the entire system is not all connected (examples below). Hence, this metric is not so useful as a system level metric.

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1					1
D			1	1			1	
E				1	1			
F					1	1		
G				1		1	1	
H							1	1

Matrix1

	A	B	C	D	E	F	G	H
A	1	1						
B	1	1						
C			1					1
D			1	1			1	
E				1	1			
F					1	1		
G	1			1		1	1	
H							1	1

Matrix2

	A	B	C	D	E	F	G	H
A	1							
B	1	1						
C		1	1					
D			1	1				
E				1	1			
F					1	1		
G						1	1	
H							1	1

Matrix6

Network Centrality—Betweenness Modularity

(Sosa, Eppinger, Rowles 2007, Borgatti, Everett, and Freeman, 2002, UCINET)

- Bridge modularity—number of times a component appears in the path between two other components
- This measure is useful to identify the important elements in the system that pass information along.
- However, the overall system bridge modularity measure only relfect the difference of the betweenness values of the elements, and is not helpful to compare the overall modularity of the system.

	A	B	C	D	E	F	G	H	Freeman Betweeness	Overall System Freeman Betweeness
A	1	1							0	14.97%
B	1	1							0	
C			1					1	4	
D			1	1			1		10	
E				1	1				4	
F					1	1			4	
G				1		1	1		10	
H							1	1	4	

Matrix1

	A	B	C	D	E	F	G	H	Freeman Betweeness	Overall System Freeman Betweeness
A	1	1							0	0%
B	1	1							0	
C			1	1	1	1	1	1	0	
D			1	1	1	1	1	1	0	
E			1	1	1	1	1	1	0	
F			1	1	1	1	1	1	0	
G			1	1	1	1	1	1	0	
H			1	1	1	1	1	1	0	

Matrix3

	A	B	C	D	E	F	G	H	Freeman Betweeness	Overall System Freeman Betweeness
A	1	1						1	4.5	0%
B	1	1	1						4.5	
C		1	1	1					4.5	
D			1	1	1				4.5	
E				1	1	1			4.5	
F					1	1	1		4.5	
G						1	1	1	4.5	
H	1						1	1	4.5	

Matrix8

Summary of Comparisons

Comparison Criteria	WI	CC	SMI	VD plot	Freeman Degree Centrality	Freeman Farness Centrality	Freeman Node Betweenness Centrality
Capable of producing a consistent interaction complexity measure for the overall system	+	+					
Capable of assessing the density of immediate interactions	+						
Capable of assessing the propagation of interactions		+					
Identifies key elements in the system that contributes to interaction complexity					+		+

Software Complexity Metrics

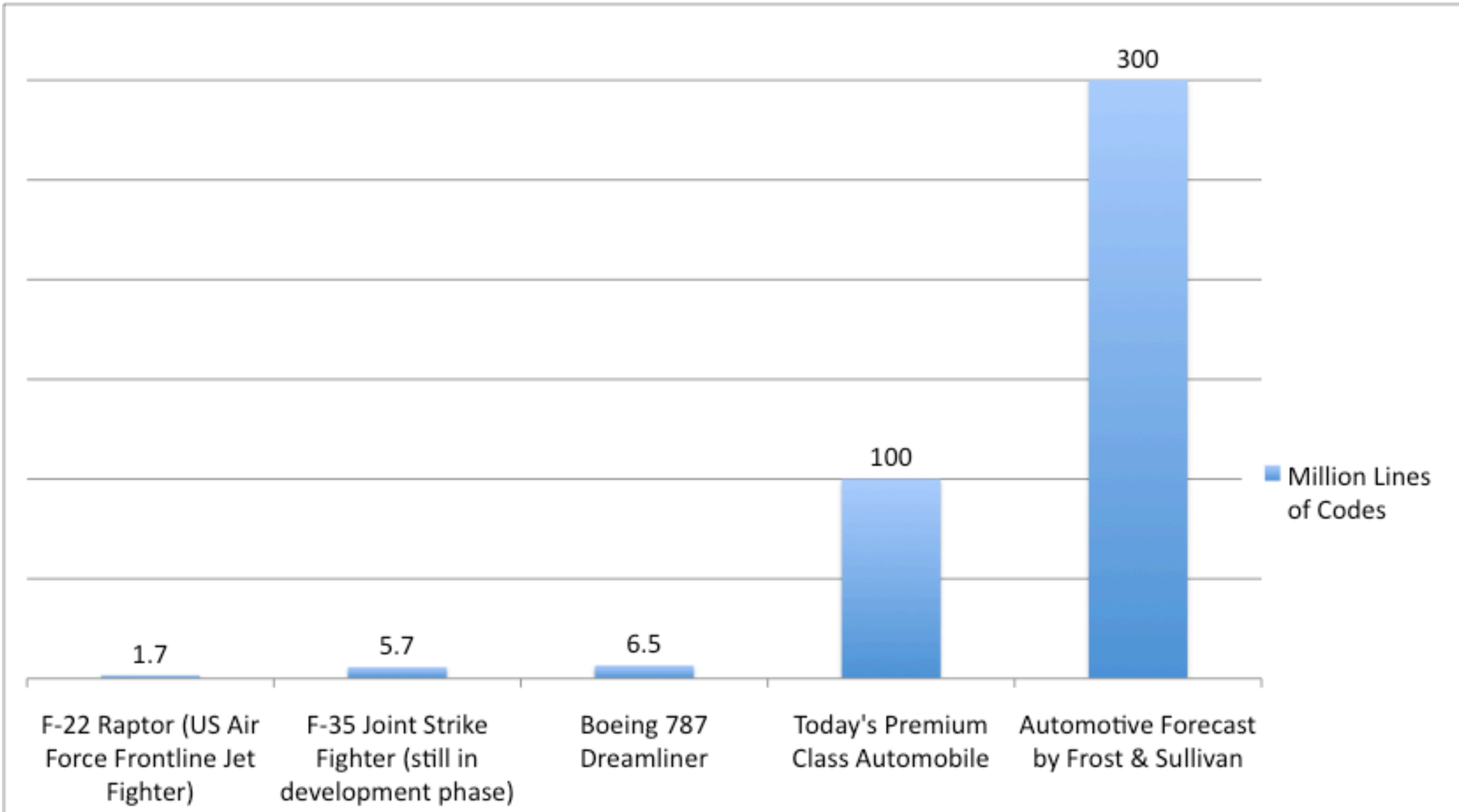
- Measuring Lexical Entities in a Software Program:
  - The number of lines of code (LOC)
  - McCabe's Cyclomatic Complexity (McCabe 1976)--the number of linearly independent execution paths through the program.
  - Halstead metrics (Halstead 1977)—a function of the number of operators and operands in the program.
  - These three metrics are all computed based on the lexical entities in a program, and hence correlate to one another very well (Kearny 1986). However, these metrics are measures only at the software code level. The purpose of this research is to study the architecture decision's implication on module interaction complexity. Hence these three metrics don't apply.
- Measuring Structure of the Software Program (Henry and Kafura 1984):
  - Length * (fan-in * fan-out) ^2
  - However, the length of the code is undefined at architecture level.
  - Fan-in and fan-out measures are captured in the WI.

Presentation Outline

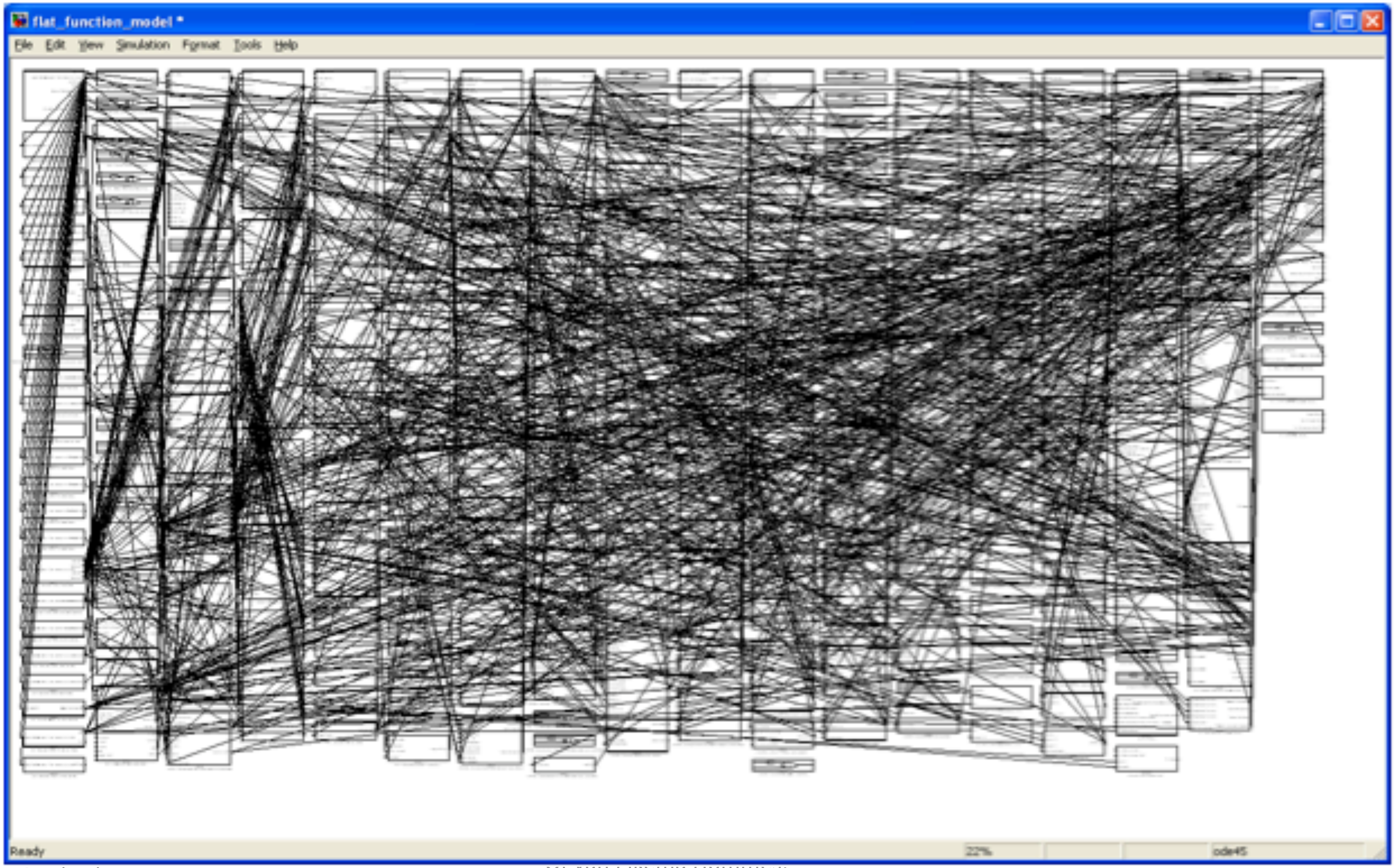
- ✓ Research Motivation
- ✓ Networks and Its Matrix Representation
- ✓ Review and Compare Complexity Measures
- ❑ Application to Vehicle Embedded Software Systems
- ❑ Summary

Lines of Code Comparison

(Data based on Charatte 2009)

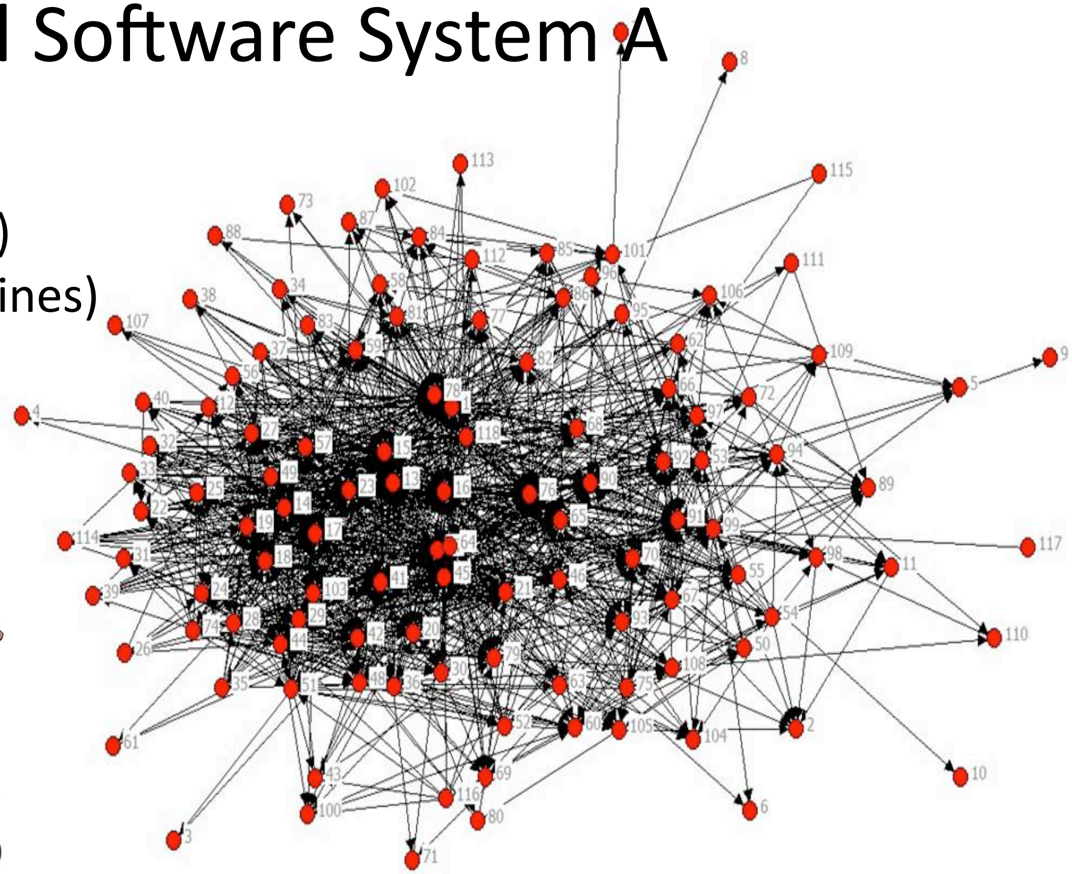


Interactions Among Vehicle Software Functions



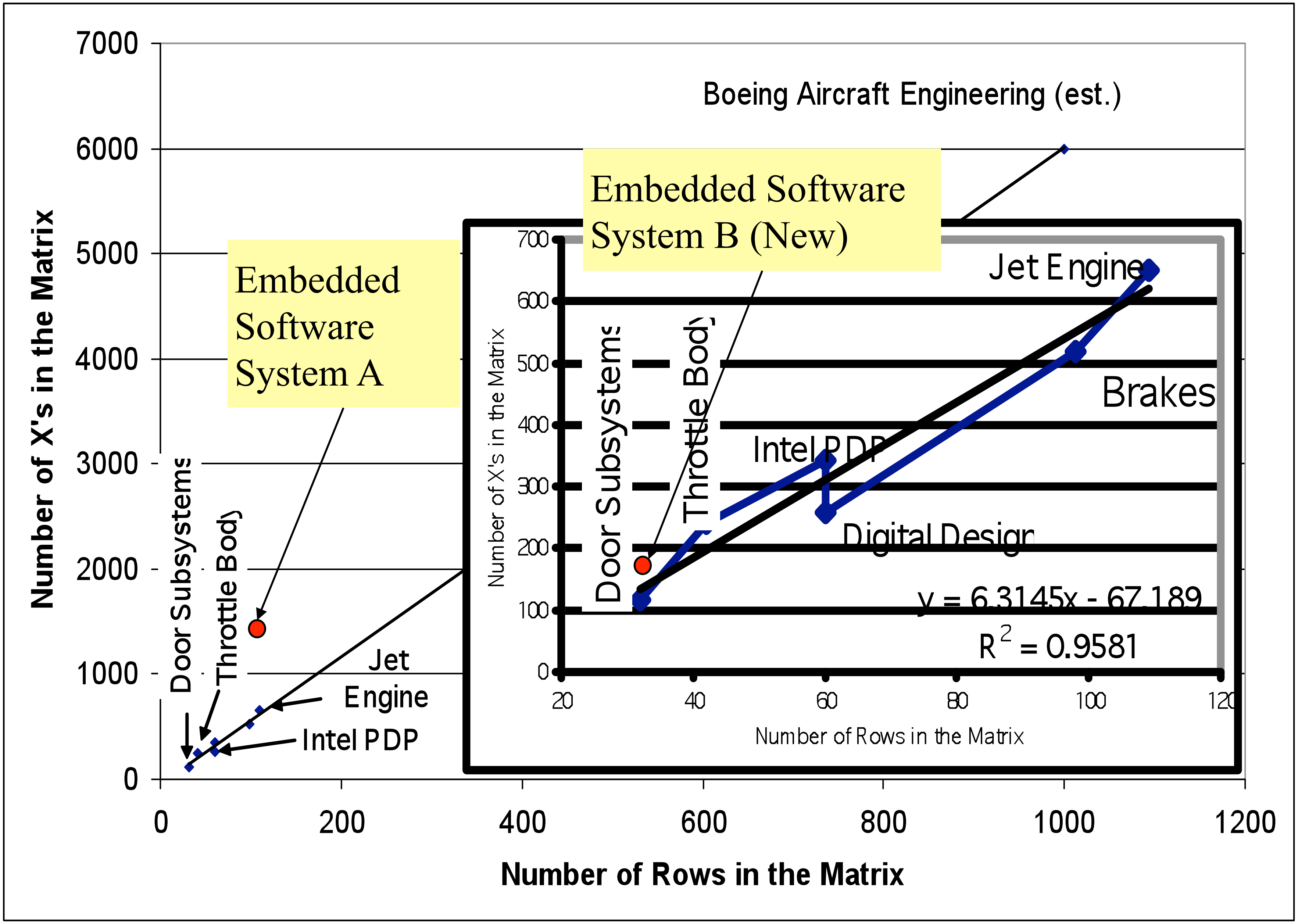
The Control Software System A

- 1 production-level software
- 117 software modules (red dots)
- 1423 binary interactions (black lines)
- 39 such production software releases per year
- <2 weeks per release

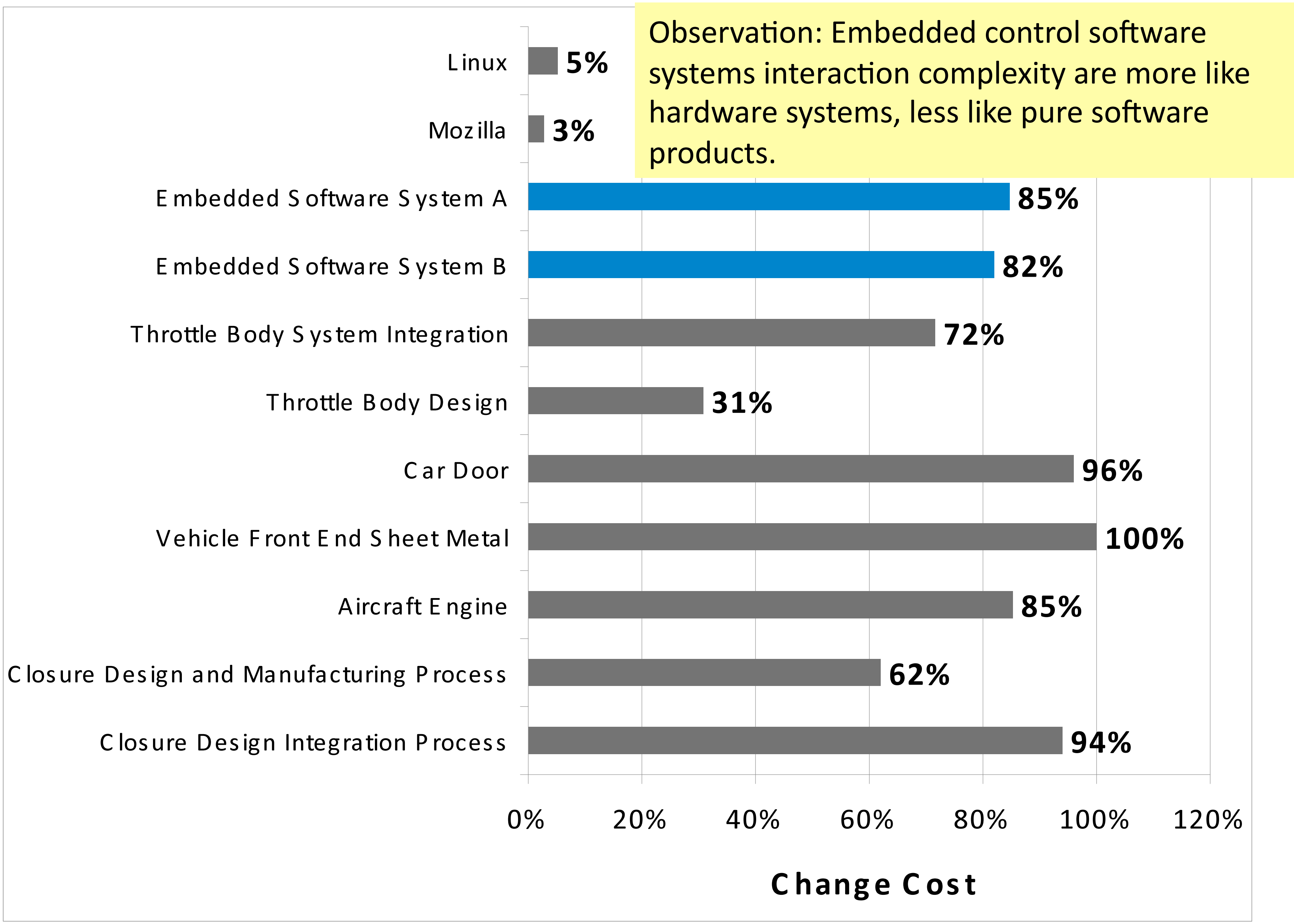


- What can I do about this web of interactions?
- How can I convince management that changes are needed?
- How do I know I actually improved the architecture?

Whitney Index Comparison



Change Cost Comparison



Recommendations to Customers

Yes, architecture improvement is needed!

Short-term Actions:

- For the modules identified by the centrality calculation, reduce the number of interfaces, and standardize the interface parameters.
- Set up an architecture group / change management group to standardize and control the system interfaces.

Mid-term Actions:

- Redesign the boundaries between the existing modules. Change function allocation to minimize the interfaces.
- Redesign the modules involved in long chain of interactions. Try to break up the functions and make the system more modular (Axiomatic Design principle can be helpful).

Long-term Actions:

- Redesign the software architecture to accommodate the new control system technologies.
- Consider the implication of organization design on the system integration efforts.

Conclusions

- This research work identified two metrics that can describe the characteristics of component interactions in a complex system:
  - The Whitney Index (WI)
  - The Change Cost (CC)
- Network centrality analysis can identify elements in the system that contributes to high interaction complexity.
- The case study demonstrated successful application of these metrics in real industry examples.

Future Research Questions

- Need more case study examples of embedded software systems.
- Are embedded software systems fundamentally more complex than IT software systems? Why?
- What design methods can be used to improve embedded software system complexity?
- Do these measures of complexity correlate with:
  - number of errors in system integration?
  - resources needed for system design?
  - number of failures/quality problems during system operation?

References

- Van Eikema Hommes, Q. "System Interaction Complexity Metrics and The Application to Embedded Software Systems." *Research in Engineering Design*, submitted for review in November 2010.
- Van Eikema Hommes, Q. "Comparison and Application Of Metrics That Define The Components Modularity in Complex Products." ASME IDETC 2008 – DTM 49140.

Thank you!

Qi Van Eikema Hommes
qhommes@mit.edu