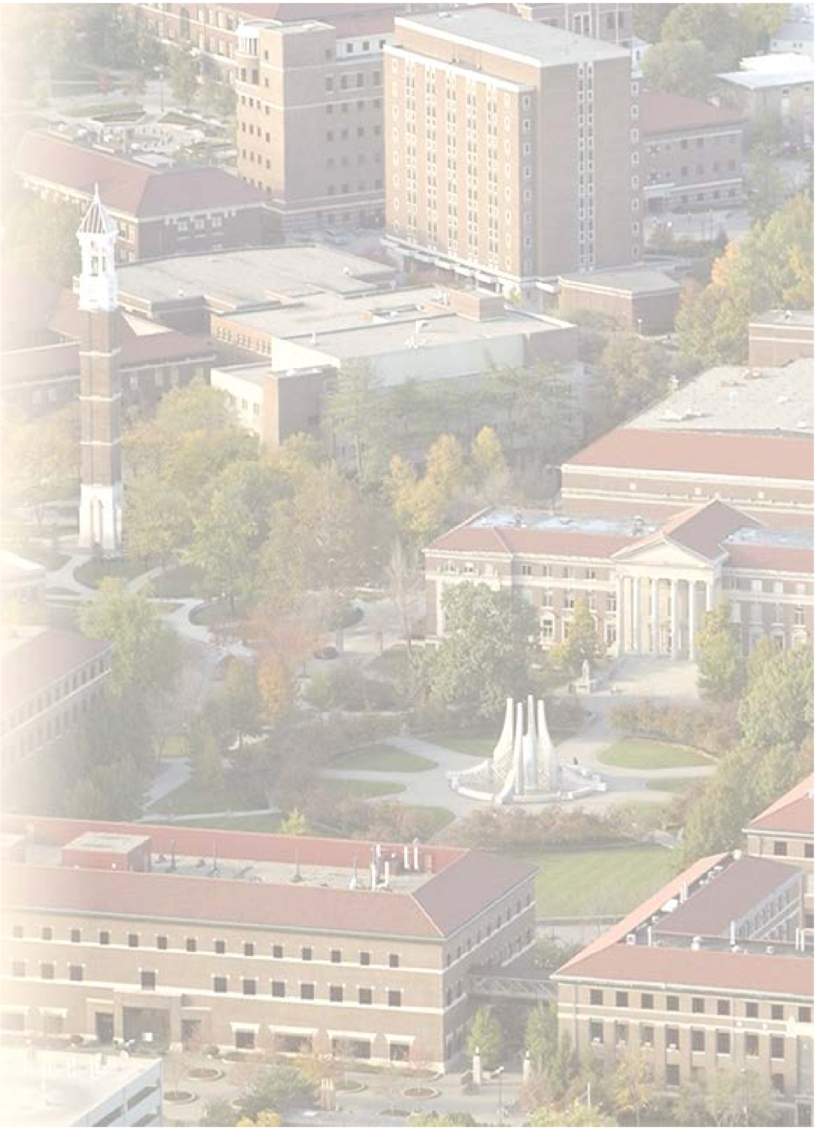
Acquisition Management for System-of-Systems: Requirement Evolution and Acquisition Strategy Planning

NPS Acquisition Research Symposium
May 16th, 2012

Purdue University
Center for Integrated Systems in Aerospace (ISA)

Dr. Daniel DeLaurentis, *Associate Professor*
Simon Han, *Graduate Student*
Zhemei Fang, *Graduate Student*

Contact: *ddelaure@purdue.edu*



Motivation – Interdependency in SoS

- Interdependency between systems is a necessary characteristic to achieve a SoS capability
- Interdependency (expressed in developmental and operational architectures), however, brings possibility of risk, especially from cascading failures.

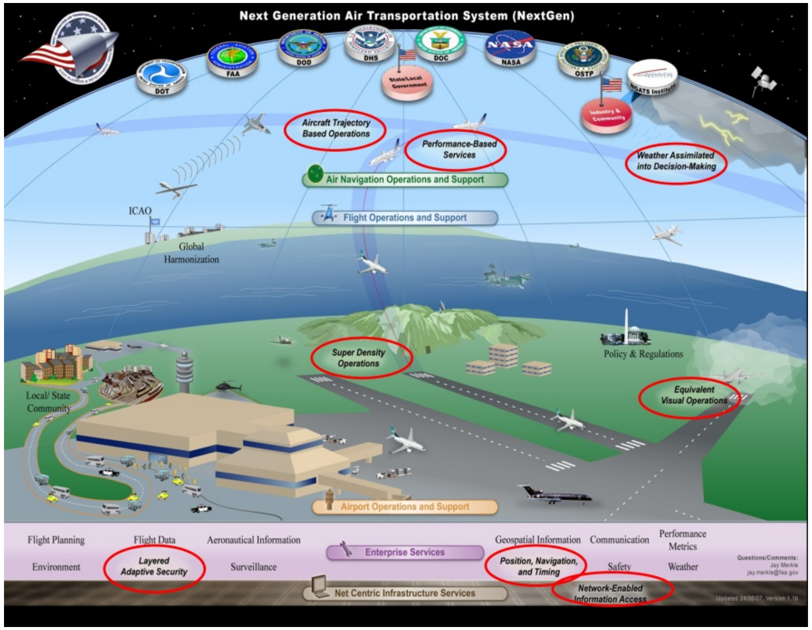


Image from: http://www.nws.noaa.gov/nextgen/ng101.shtml

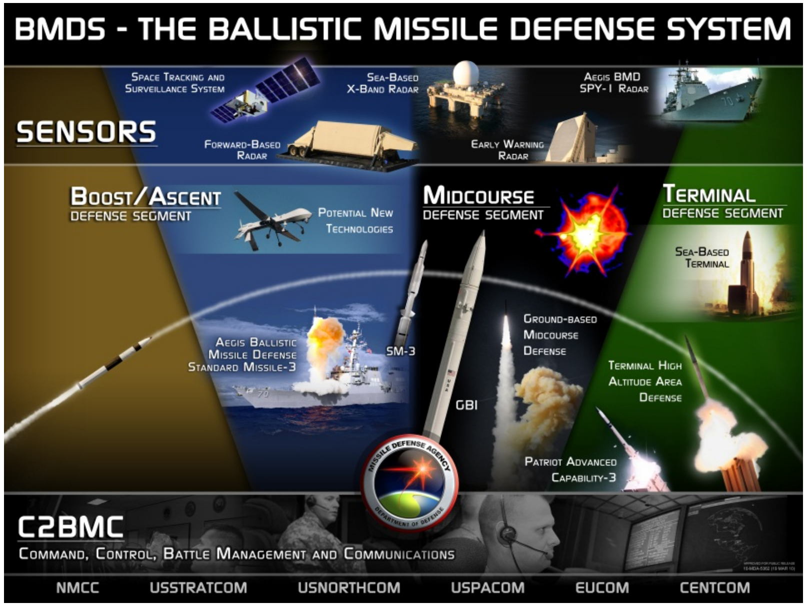
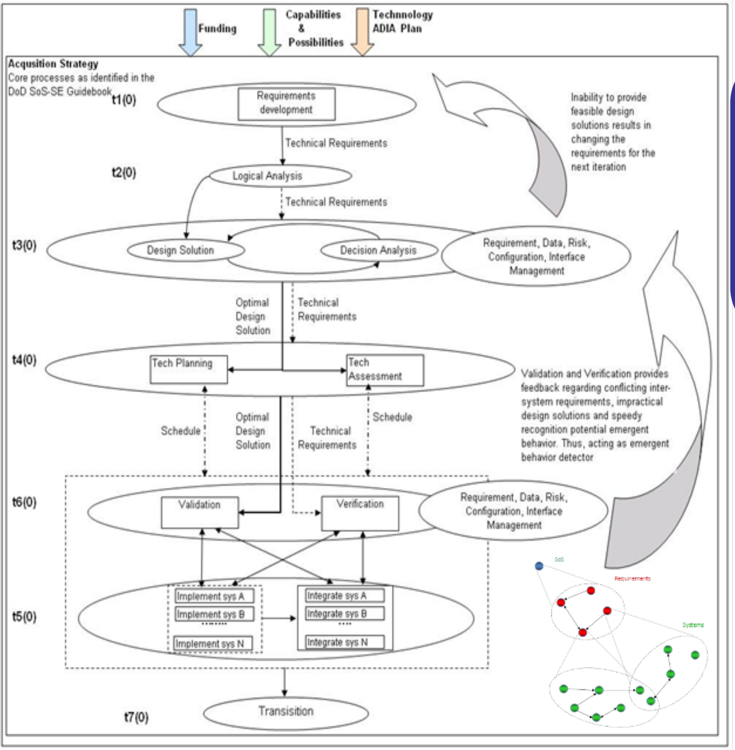


Image from: website of Missile defense agency
http://www.mda.mil/system/system.html

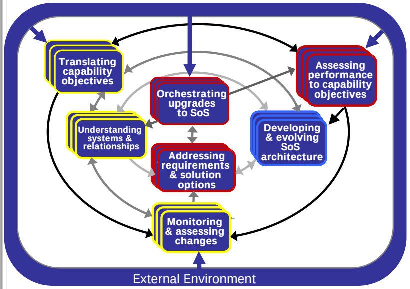
Brief History: Our* Methods Development in this Arena

** Thanks to NPS ARP and now also DoD SERC*

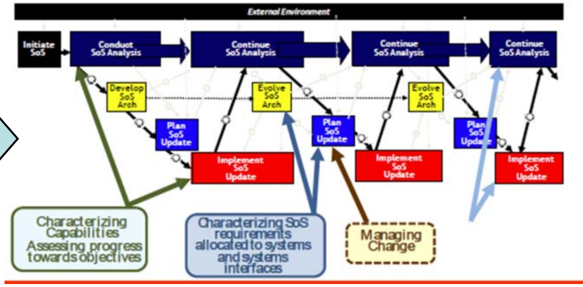
DoD SoSE Guidebook

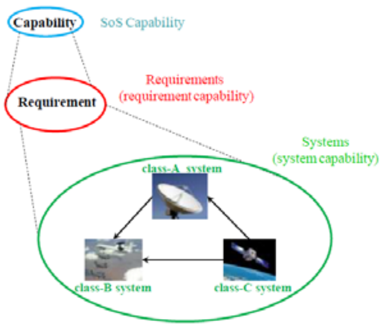


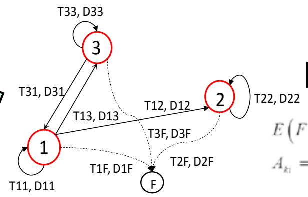
Trapeze



Wave Model



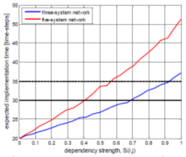




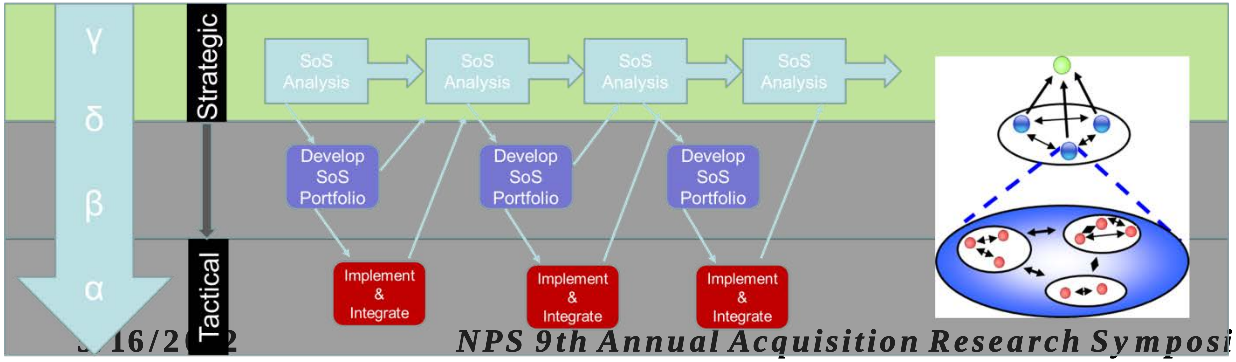
Markov

$$E\left(F \mid x_j(0)\right) = cA(I-A)^{-2} b$$

$$A_{ki} = P\left(x_k(n+1) \mid x_i(n)\right)$$

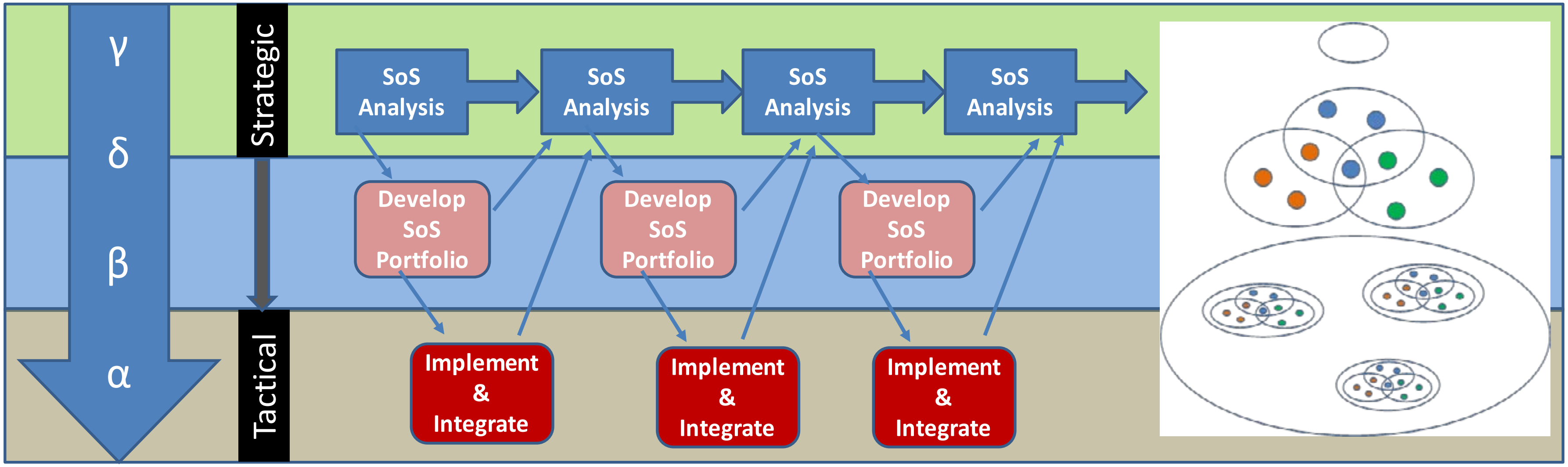


CEM



Methods	Nature	Inputs
CEM	Discrete Event Simulation	Probabilities, Connectivities
Markov	Probabilistic Graphical	Probabilities, Connectivities
Bayesian Network (BN)	Probabilistic Graphical	Conditional Distribution Connectivity
Portfolio Approach	Decision/Analysis based	Capabilities, Requirements, Connection rules

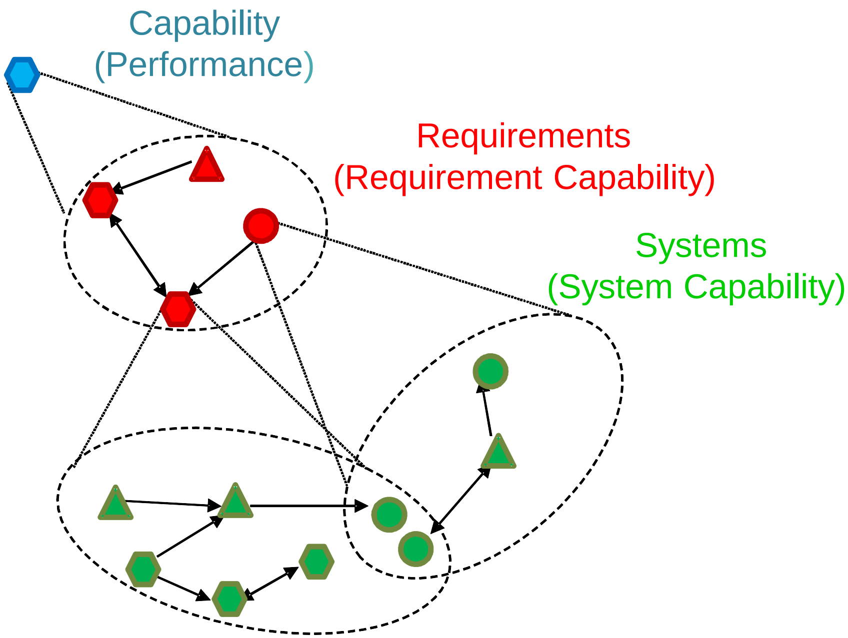
Aim– Tools for Better Acquisition



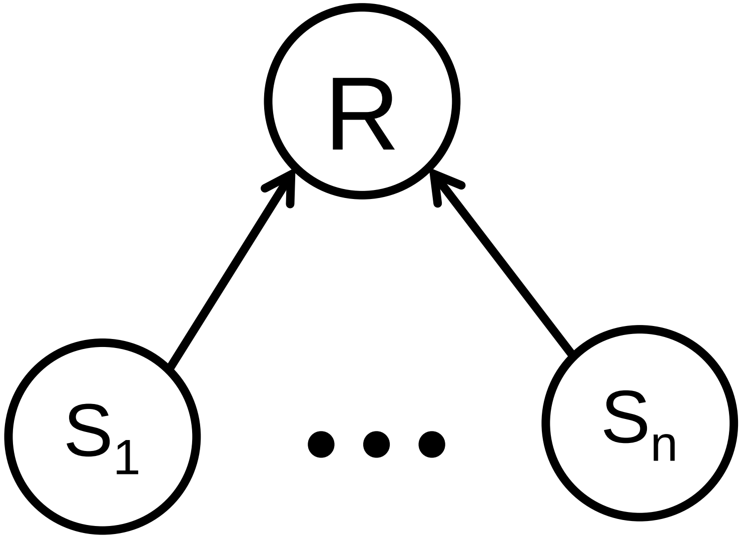
- Aim: Method that supports pre- and post- milestone A, B activities by analyzing the impact of requirement changes and system development failure during generation of a SoS capability.
  - uncertainties in systems and requirement interdependencies
- Evidence: DoD, GAO and others identify requirement evolution & technology uncertainty as critical issue

A hierarchical representation of an SoS

- Interdependent systems are grouped to fulfill a requirement while interdependent requirements are expected to achieve a capability.
- Hierarchy in interdependencies contribute to increasing capability but also may lead to failure through concealed risks



Interdependency analysis via Bayesian Network (BN)



- In analyzing interdependencies, we need:
  - Inherent uncertainty of systems
  - Propagation of uncertainty
- BN is a directed acyclic graph.
- Node *R* has *n* parent nodes.
  - *R*: the requirement
  - S_i: the system that meet this requirement.
  - ***PA***(R_i) denote the set of parents of the node R_i, i.e. $\{S_1 \ldots S_n\}$
- Applying the law of total probability, the probability of achieving a particular requirement, node R_i *is*:

$$p(R_i = 1) = \sum_{\boldsymbol{S}} p(R_i = 1 \mid PA(R_i) = \boldsymbol{S})\, p(PA(R_i) = \boldsymbol{S})$$

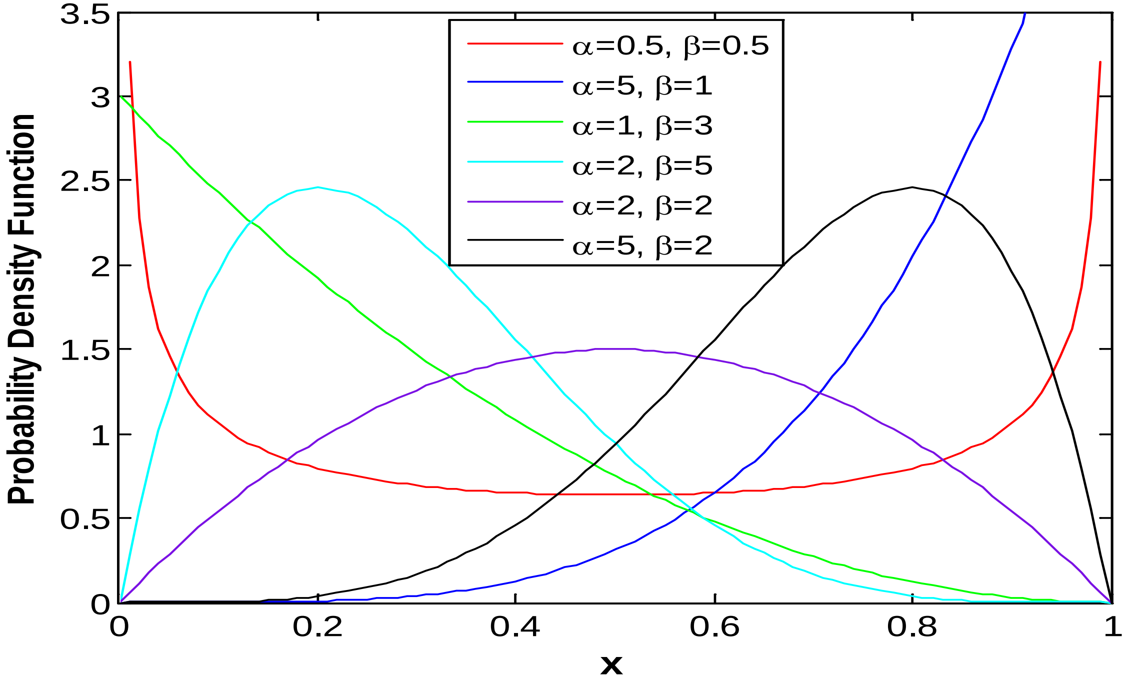
where ***S*** is the set of all the possible combinations of parent node values

- For example, if ***PA****(Ri)* includes two parent nodes S_1 and S_2,

then ***S*** = {(0, 0), (0, 1), (1, 0), (1, 1)}.

Uncertainty with Beta Distribution

- Beta distributions are often used as node failure probability information to address uncertainties.
- Beta distributions are a family of continuous probability distributions defined on the interval between 0 and 1 parameterized by two positive shape parameters (α and β)



Overview of interdependency analysis

- Each system has its own inherent information (e.g., failure rate or reliability).
- Integrated information (e.g., failure rate or reliability) can be obtained by combing inherent information and propagating effects from dependent systems.
- Process:
  1. Estimate propagating effects from dependent systems using a joint probability.
  2. Combine the inherent information and propagating effects.

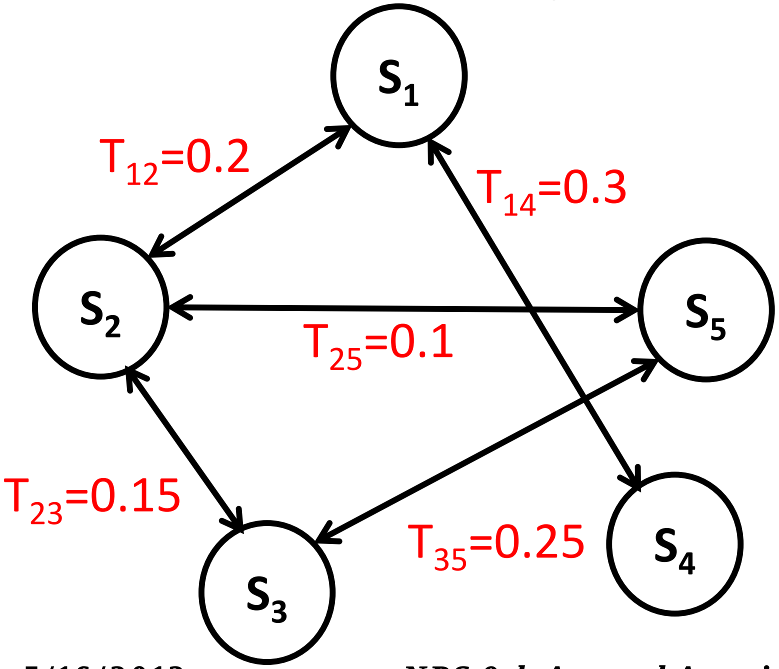
Requirement Evolution

- Both requirement evolution and Technology Readiness Levels (TRLs) can impact component system failure rates.
    - ➢ Requirement evolution refers to changes that take place in a set of system requirement after the initial requirement analysis [1].
    - ➢ TRLs are a systematic metric/measurement, invented by NASA that supports assessments of the maturity of a particular technology and the consistent comparison of maturity between different types of technology [2].
- In this study, we assume TRL to be constant for all component systems.
- Therefore, if a requirement evolves to a higher level, system failure rates will increase always.

[1] Anderson, S., & Felici, M. (2001). Requirements Evolution From Process to Product Oriented Management. *3rd International Conference on Product Focused Software Process Improvement* (pp. 27-41). Kaiserslautern, Germany.
[2] Mankins, J. C. (1995). Technology Readiness Levels: A white paper. *Advance Concepts Office, Office of Space Access and Technology*: NASA.

Synthetic Problem Demonstration

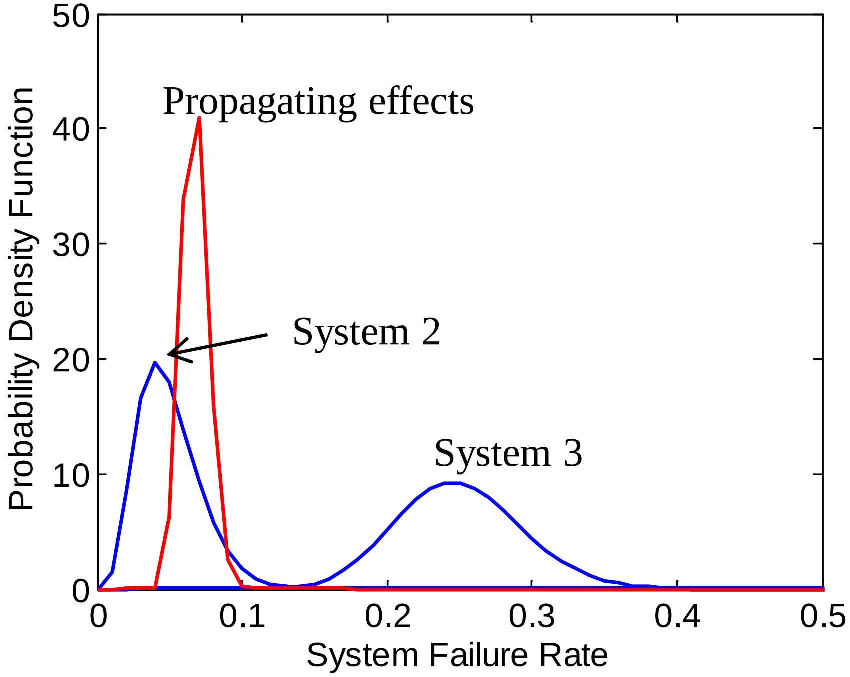
- In a five-system network, the development of system 1, here denoted by S_1, depends on the development of system 2 and 4.
- Table below summarizes inherent failure rates for all systems in terms of beta distributions, means, and standard deviations.
- T values indicate the dependency strength and correspond to the conditional probability of a failure propagating to a dependent system.

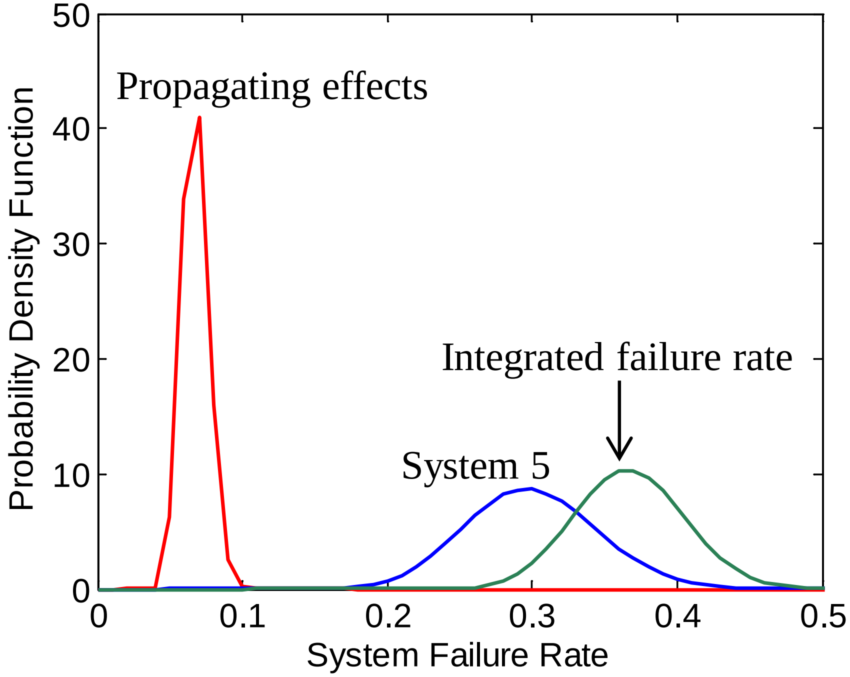


System	Failure Rate Distribution	Mean of Failure Rate	Standard Deviation of Failure Rate
System 1	Beta (20, 80)	0.2	0.04
System 2	Beta (5, 95)	0.05	0.02
System 3	Beta (25, 75)	0.25	0.04
System 4	Beta (10, 90)	0.1	0.03
System 5	Beta (30, 70)	0.3	0.05

Results – Fusion of failure rate information

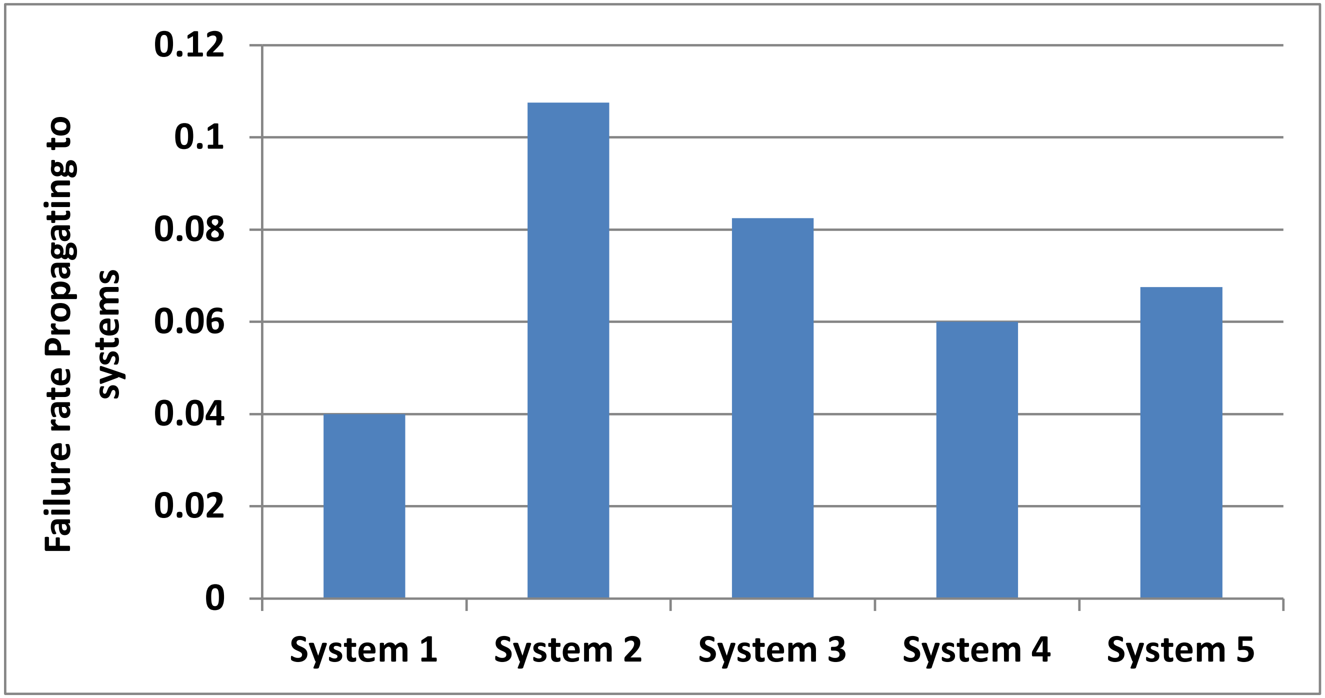
- Figure in left shows the fusion of inherent failure rates for calculate propagating effects.
- Figure in right shows the fusion of propagating effects with inherent failure rate of system 5.





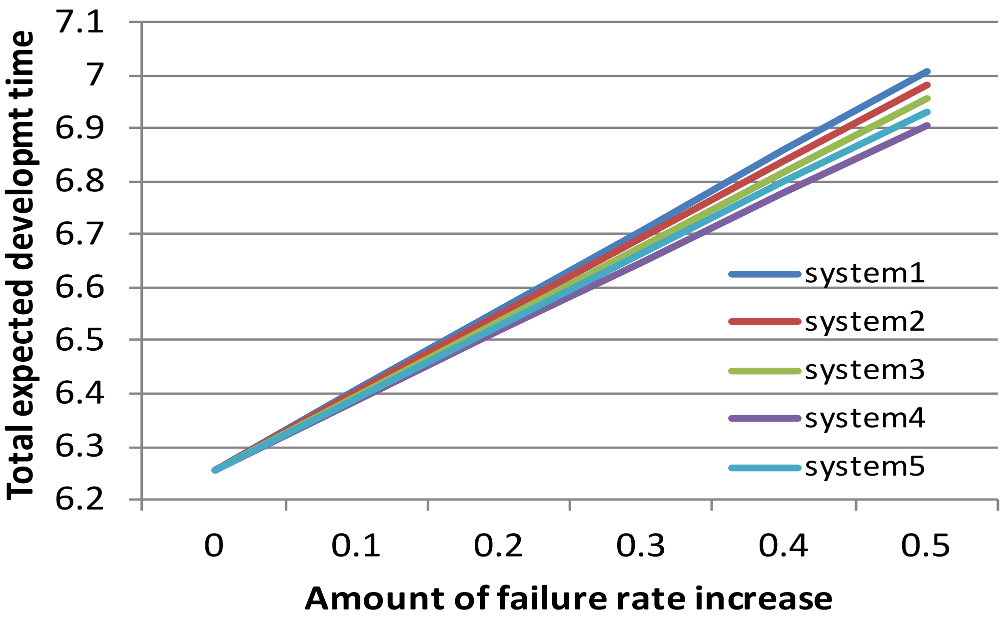
Results – Comparison of propagating effects

- Figure below shows the mean values of propagating effects for all systems.
- System 2 has the highest propagating effects indicating strong dependencies with numerous other systems.
- It also has a higher probability to be disrupted by other system failures during the development process.



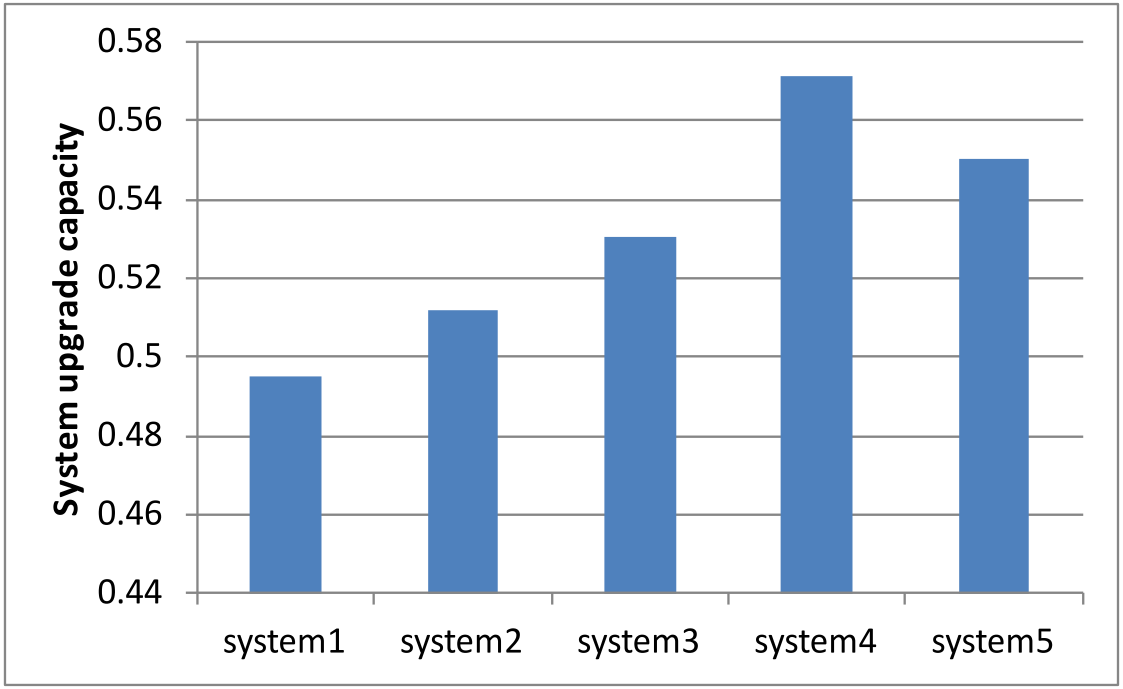
Results – Requirement Evolution Effects

- Figure below shows total expected development time of each system with inherent failure rates increasing from 0 to 0.5.
- System 1 has the steepest slope -- most critical in terms of impact on total expected development time.
- Linearity -- a result of the model assumption of constant interdependent strengths.



Results –Requirement Evolution Effects (Cont.)

- Figure below shows the system upgrade capacity diagram for the synthetic problem.
- System 4’s upgrade capacity is higher than others -- it can be substituted with an alternative system which has higher failure rates.
- System 1 has the lowest upgrade capacity because it was the most critical system.



Implementation Example

- Littoral Combat Ship (LCS) systems is adopted as example for applying the proposed approach.
- LCS systems is designed to counter growing potential threats in the littoral area such as coastal mines, quiet diesel submarines and terrorists on small, fast, armed boats.

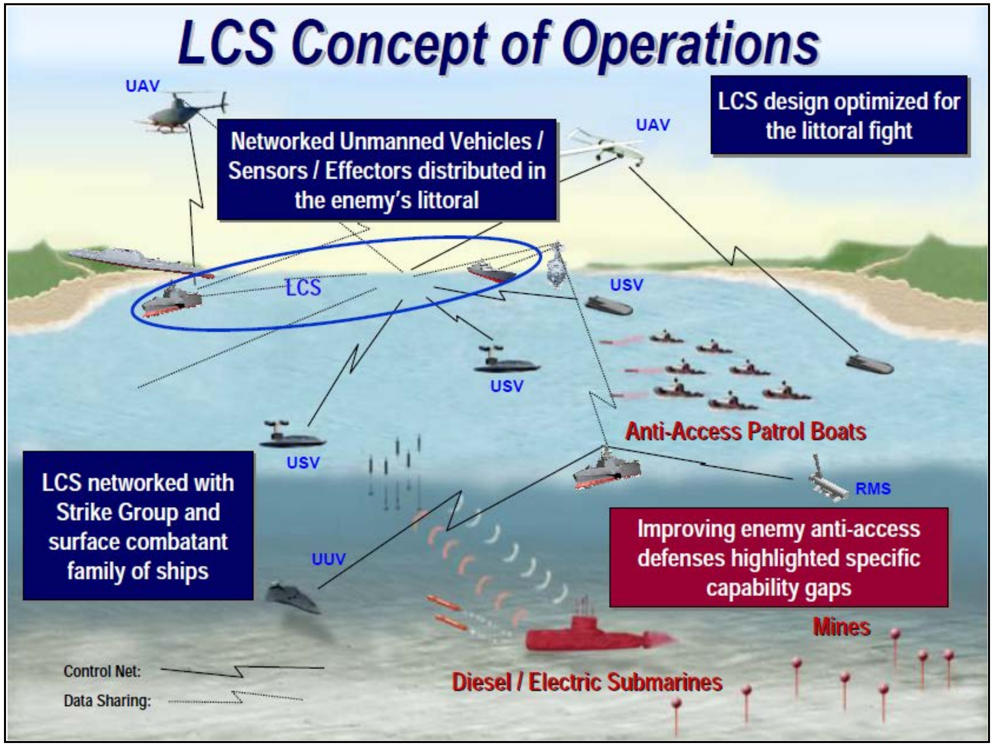
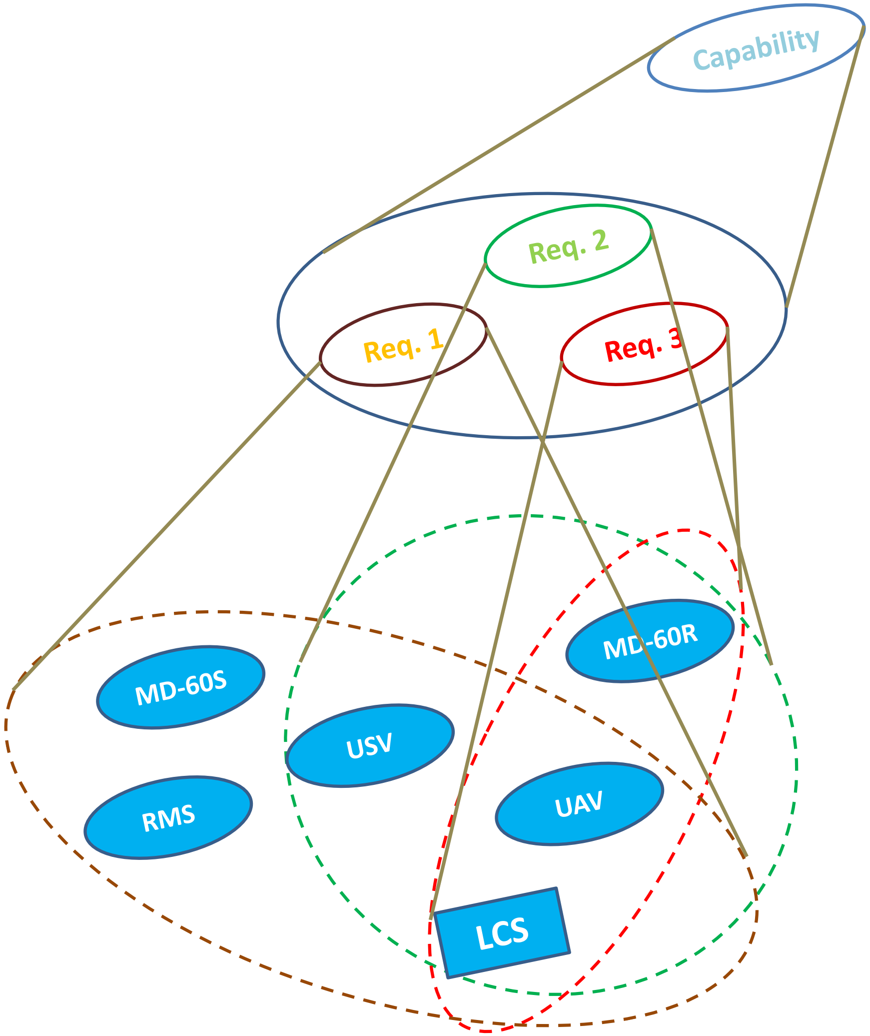
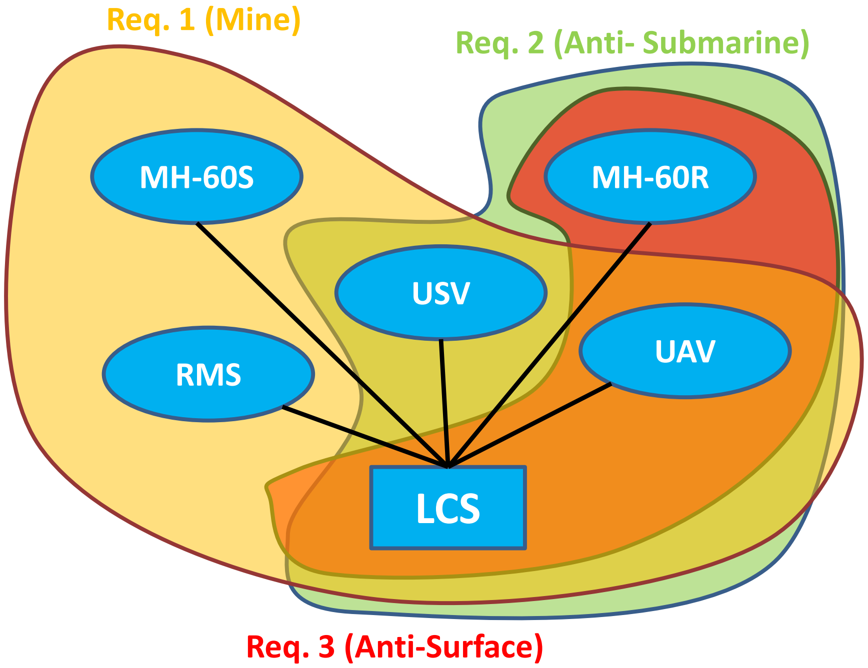


Image from: Presentation slides by RDML Vic Guillory of OPNAV at Mine Warfare Association Conference (titled "Littoral Combat Ship", 08-May-07)

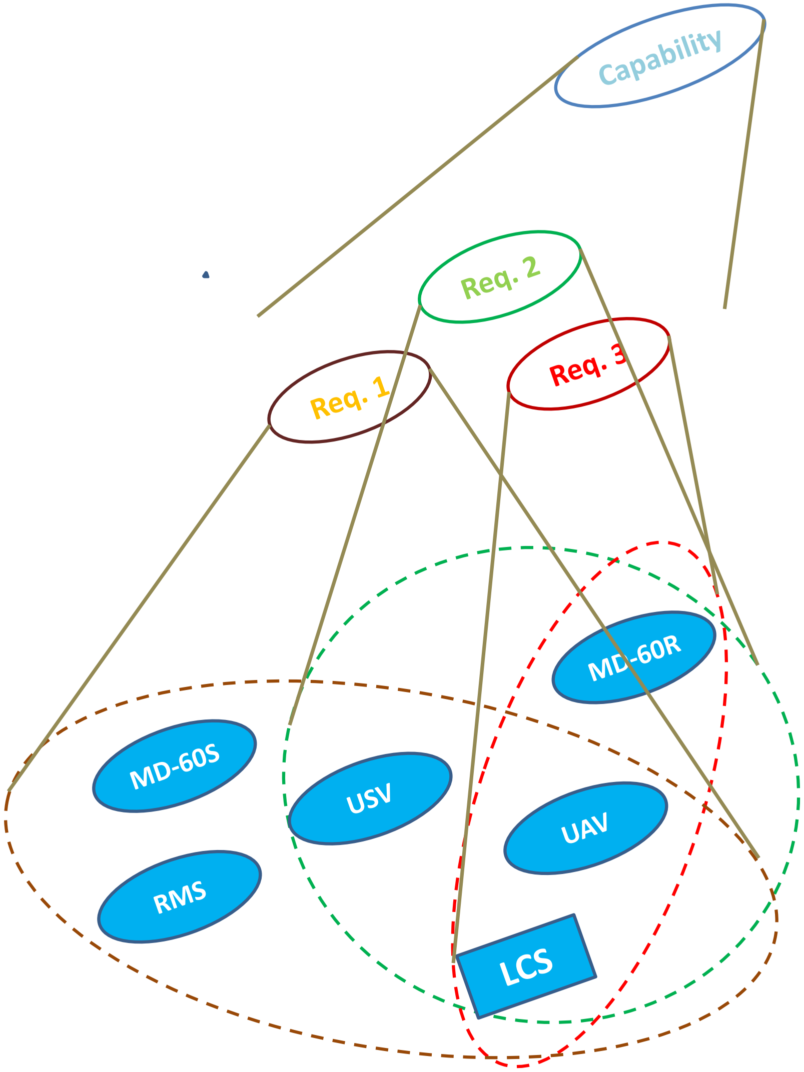
Hierarchical representation of Littoral Combat Ship



MH-60R: Armed Helicopter for Surveillance and Attack
MH-60S: Helicopter for Airborne Mine Counter-Measure
UAV: Unmanned Air Vehicle
USV: Unmanned Surface Vehicle
RMS: Remote Mine Hunting System
Torpedo, **Missile**, and the **LCS**



Interdependency analysis



Fusion all information from requirements

$$p(C=1)=\sum_{\mathbf{R}} p(C=1\mid PA(C)=\mathbf{R})\,p(PA(C)=\mathbf{R})$$

Fusion all information from systems

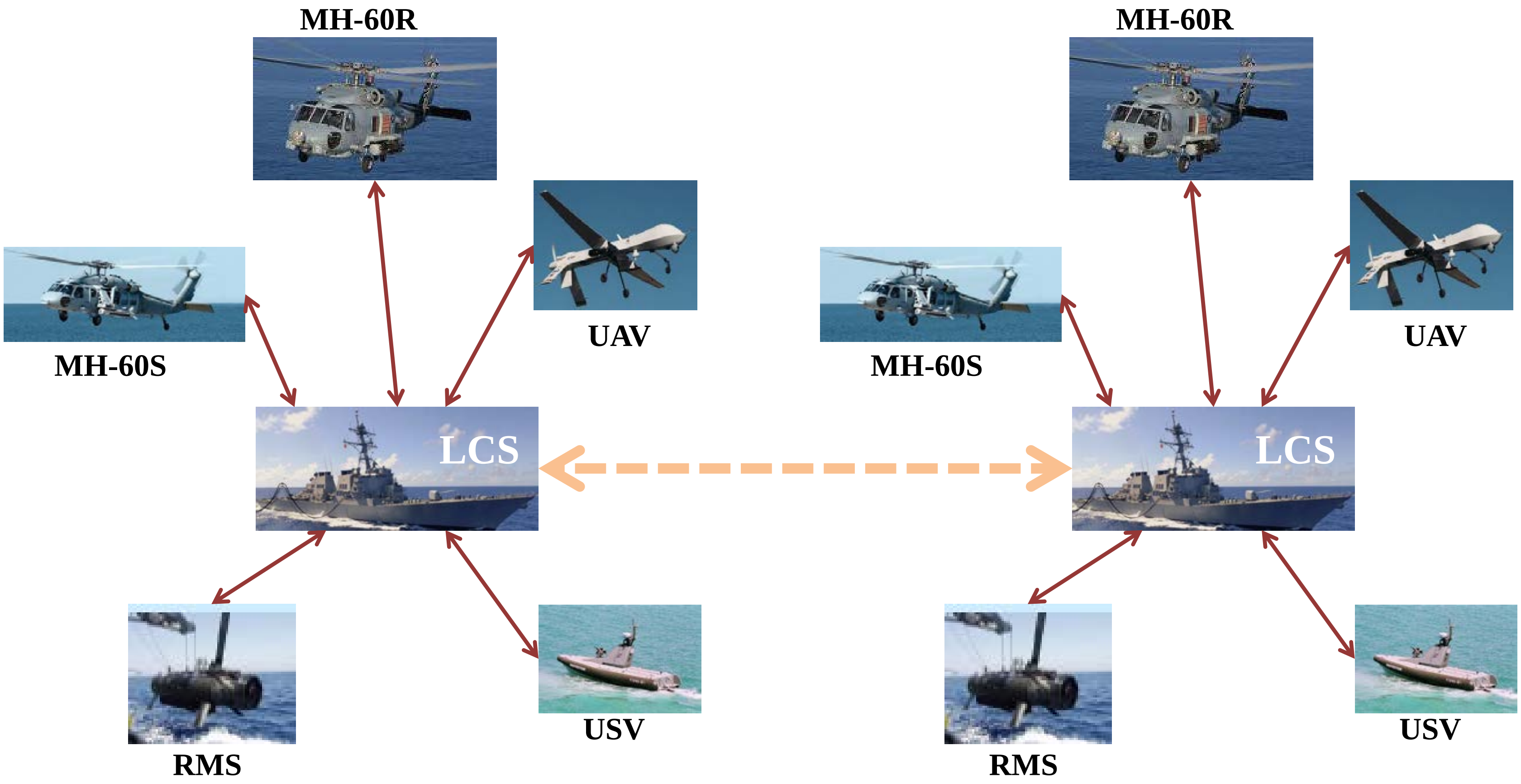
$$p(R_k=1)=\sum_{\mathbf{S}} p(R_k=1\mid PA(R_k)=\mathbf{S})\,p(PA(R_k)=\mathbf{S})$$

System failure rates

$$P(System_i = Failure) = 1-e^{-\lambda_i t}$$

The performance of LCS systems is defined as the probability to complete the mission.

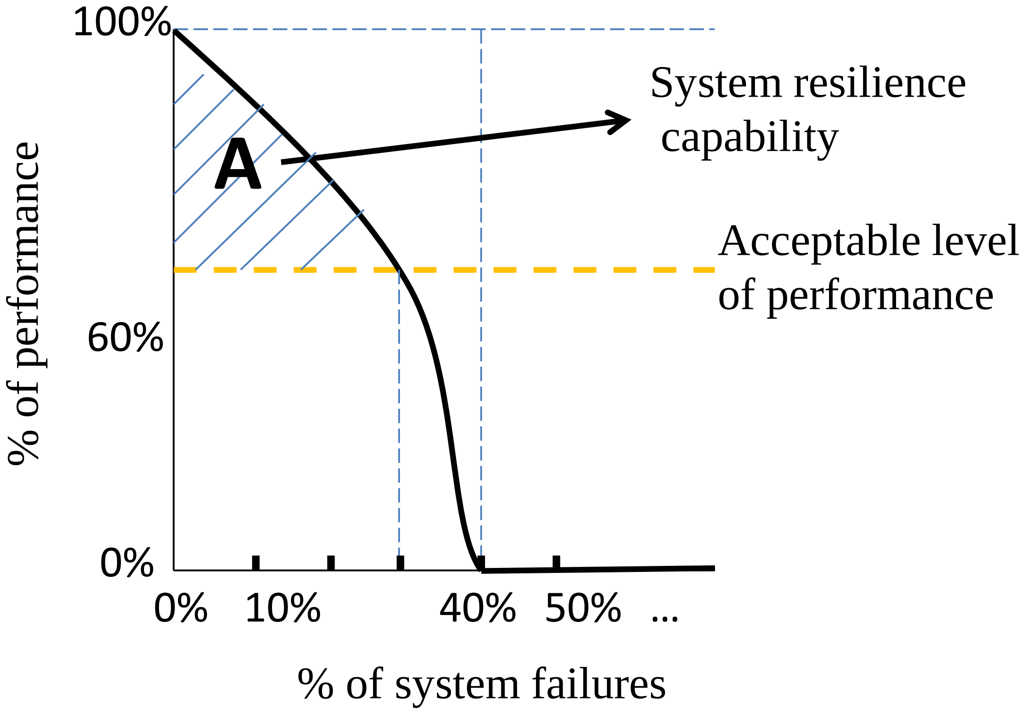
Two different architectures



- architecture 1 has NO communication between LCS command centers
- architecture 2 DOES have communication between LCS command centers

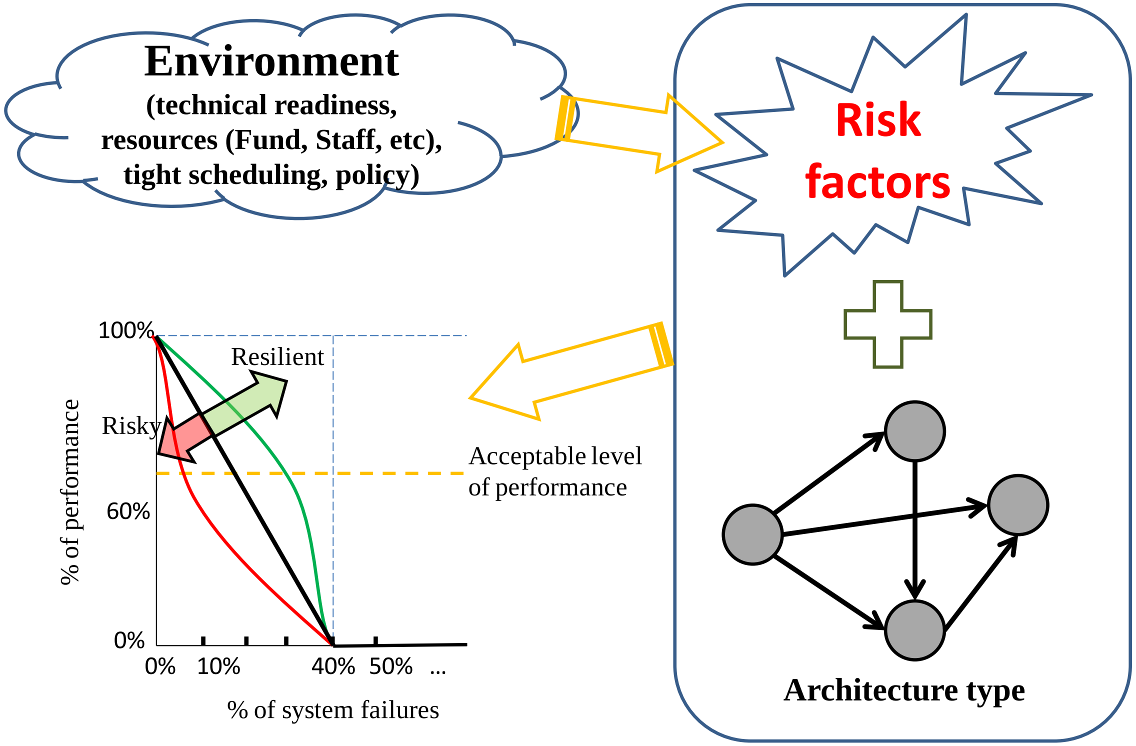
Resilience Metric

- Resilience pattern represents the relationship between performance (or capability) and the level of component failures in the entire domain.
- The yellow horizontal line represents the acceptable level of performance to meet a system requirement.
- The area of A, called system resilience capability, indicates the ability of the system to sustain the performance beyond a desired level in the face of failures.



Two factors affecting resilience patterns

- Architecture type: the fundamental organization of an SoS embodied by its components and their relationships to each other (i.e., interdependencies).
- Risks to SoS: the sources and consequence of component failures.

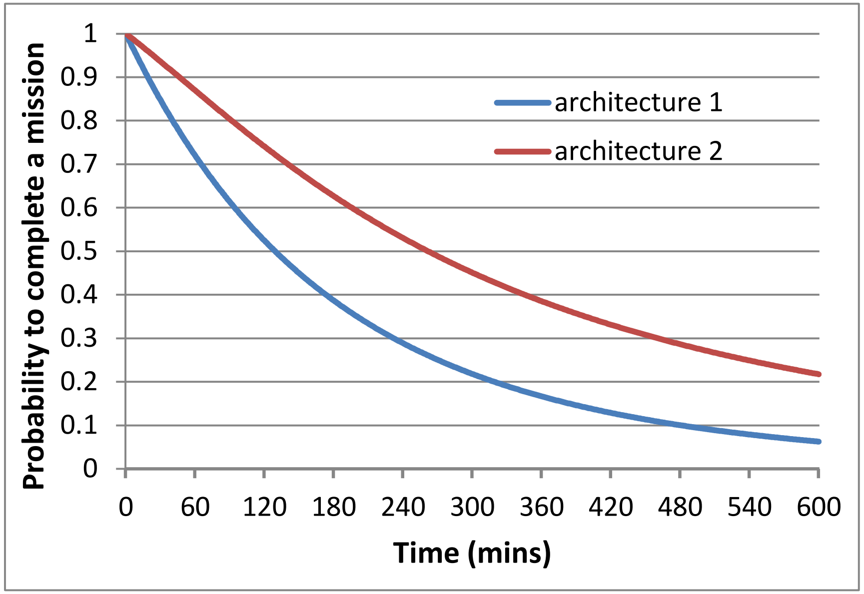


Environment
(technical readiness, resources (Fund, Staff, etc), tight scheduling, policy)

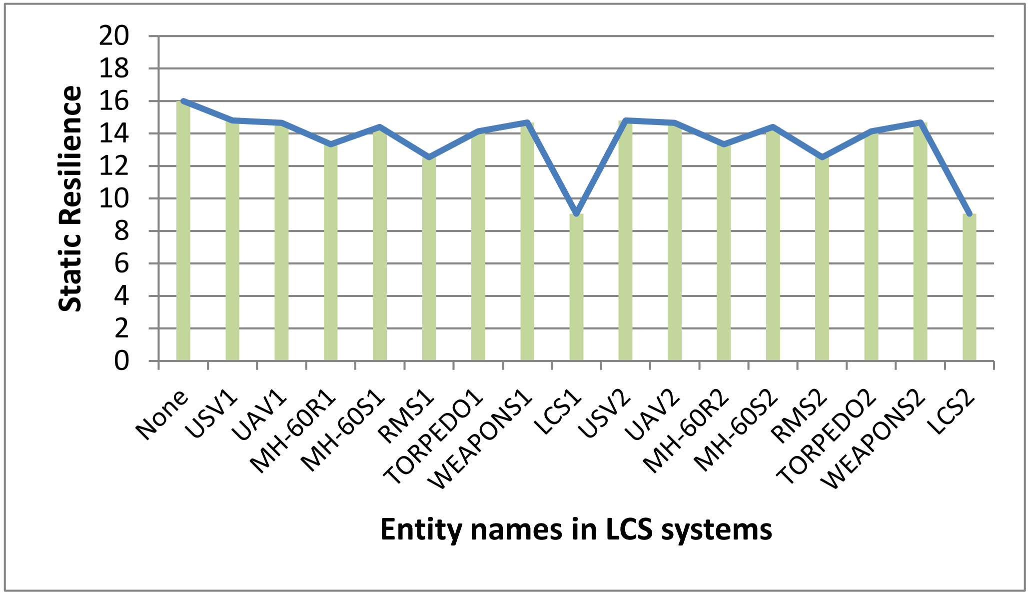
Risk factors

Architecture type

Results of LCS systems case study



Evolution of system performance of both architectures for 600 mins



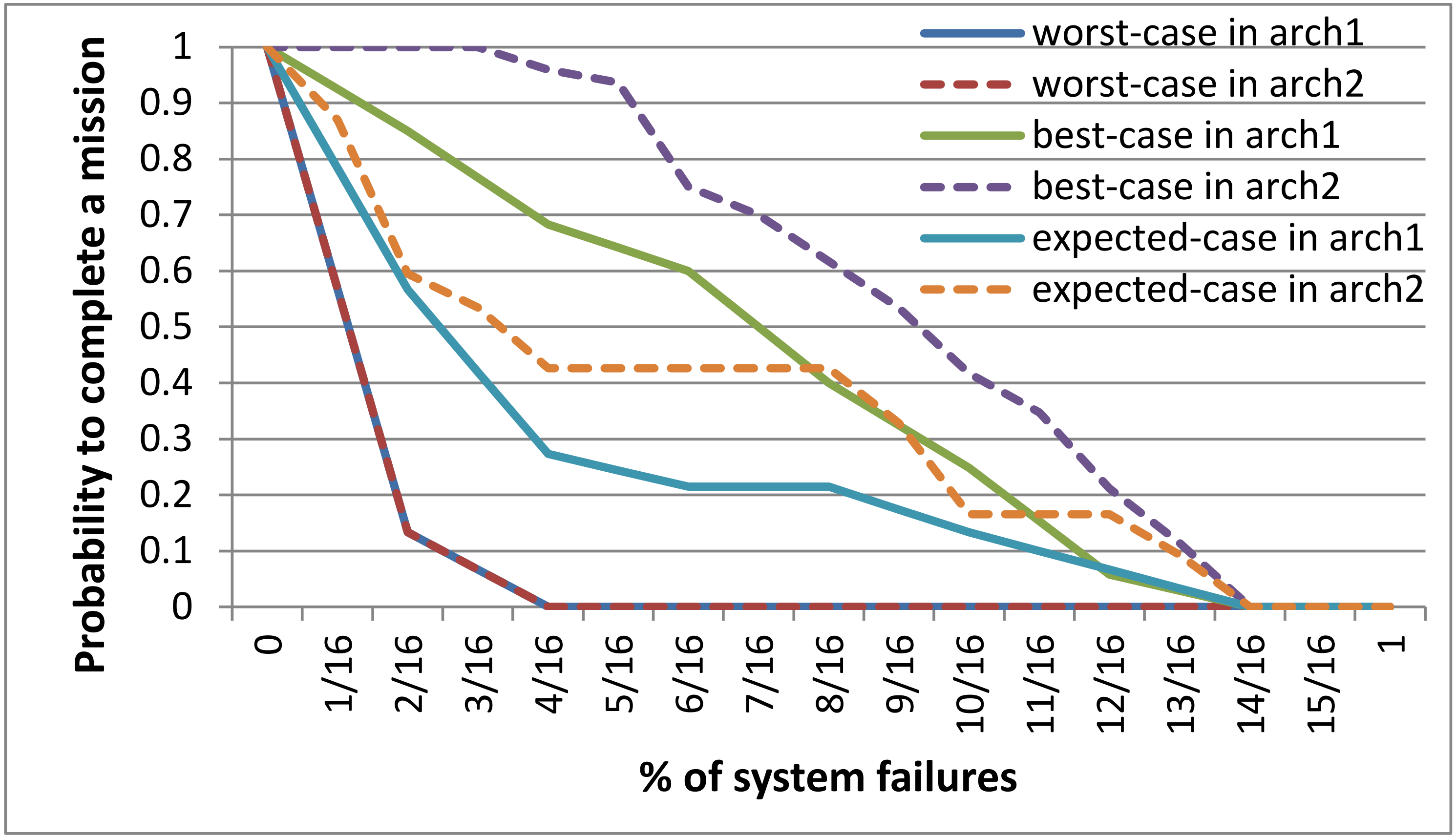
Static resilience with one entity failure in architecture 1

- The exponential distribution is a function of time: the failure rate of a system increases as time elapses.
- Arch2 has the better resilience pattern than Arch1.

- The systems with lower static resilience values are critical systems.
- The right figure indicates that LCS nodes are most critical.

Results of LCS systems case study

- Best-case: when critical systems fail last.
- Worst-case: when critical systems fail first.
- Expected-case: when system failures occur based on systems' own failure rates.



The two major factors, architecture type and failure rate of a system, affect system resilience.

Strengths and weaknesses of the BN

- Strengths
  - Can handle directed nets of interdependencies
  - Can easily represent the system structure and interactions
  - Can be automatically updated with new observations
  - Can be used constructively in sensitivity mode when very few or no data available, with some prior knowledge
- Weaknesses
  - Cannot handle cycles in network.
  - Require rich collection of data (failure and performance) to obtain conditional probability distribution connectivity and node information

Conclusion

- A Bayesian Network approach is adopted to analyze interdependencies by measuring component failure rates and total expected development time for a whole SoS.
- Integration of inherent failure rates and propagating failure rates is calculated to more completely represent the true risk and more faithfully determine the critical components.
- Upgrade capacity for each system can be obtained for decision makers when requirements evolve.

Future Work

- A Bayesian Network approach can only use 0 or 1 to represent two discrete states, like ‘working’ or ‘failure’; thus continuous variables such as development percentage cannot be expressed directly
- Collection of input failure rates -- may be inferred from analysis of historical data on TRL levels as they related to eventual development success.

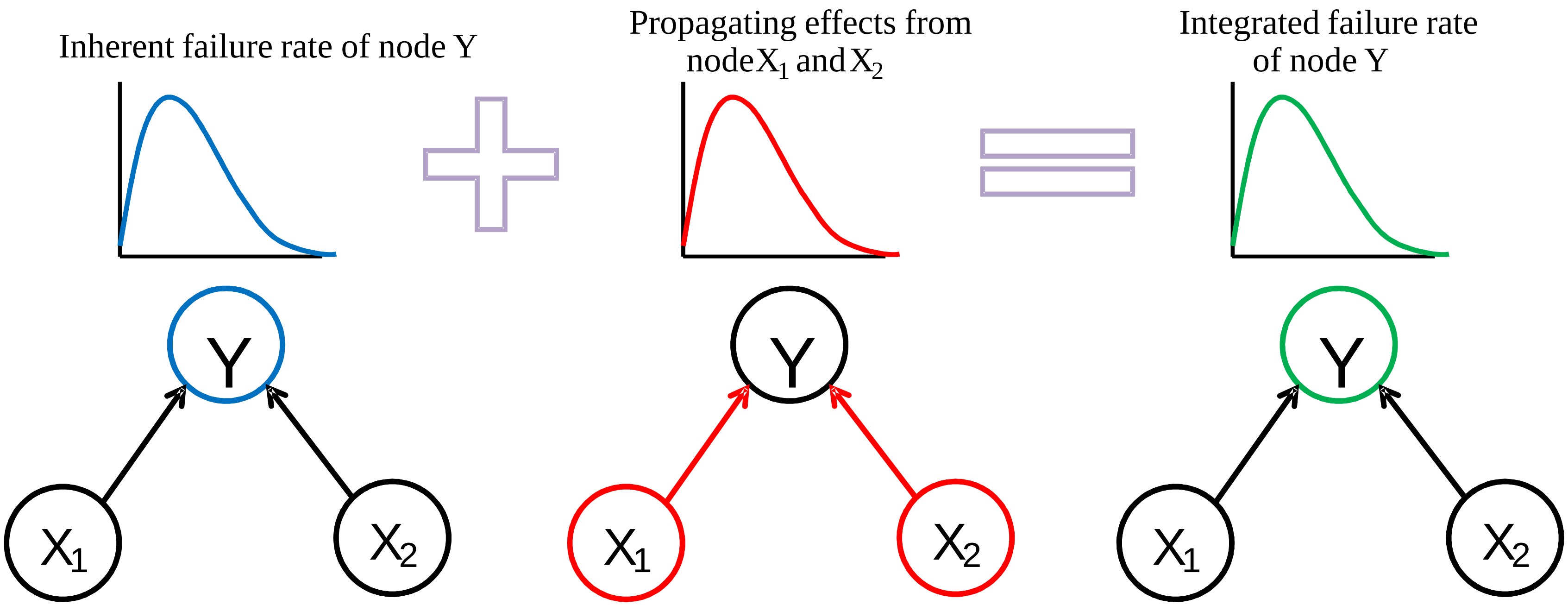
Thank you

Questions ?

BACKUP SLIDES

Overview of interdependency analysis

- Each system has its own inherent information (e.g., failure rate or reliability).
- Integrated information (e.g., failure rate or reliability) can be obtained by combing inherent information and propagating effects from dependent systems.



Inherent failure rate of node Y

Propagating effects from node X_1 and X_2

Integrated failure rate of node Y

Previous Work

- **2008 --** Computational Exploratory Model (CEM) based on the 16 basic technical management and technical system-engineering processes.
- **2009 --** Improvements to allow for modeling of scenarios that illustrate the underlying dynamics that produce schedule delays and cost overruns.
- **2010 --** Addition of system development-risk detail that enables the analysis of the impact of system maturity on the development process when the higher-order effects of interdependencies are captured.
- **2011 --** A capability module based on Markov analysis to the CEM that aggregate the network interdependency characteristics and compare alternatives with respect to the time required to arrest the propagation of development delays in a network.

Case study – Littoral Combat Ship (LCS) (Cont.)

- The failure rates are also given to all entities based on mission time limit of that entities using the exponential distribution.

$$p(X = 0) = 1 - e^{-\lambda T} \qquad \lambda = \frac{1}{mission \;\; time \;\; limit}$$

- where T is the system exposure time and λ is the system's failure rate which can be written as a function of the mean time between failures (mission time limit).
- Expected reliability of LCS systems to finish the mission is calculated as the performance using Bayesian Network with two different architectures: unavailable (arch1)/available (arch2) communication between command centers of two Littoral Combat Ships.

Static resilience of a complex system

- Two metrics are defined to quantify system resilience in the multi-level system tester module.
- The static resilience can be thoughts of as a ‘specific’ performance measure – for a given number of component failures, how much performance is maintained.
- Static resilience is defined as the ratio of percentage of performance of a complex system (e.g., SoS) to percentage of component failures

$$\text{Static Resilience} = \frac{\%\text{ of performance (or capability)of a system}}{\%\text{ of component failures}}$$

Metric: System Resilience

- The general definition of resilience is the capacity of a system to survive, adapt and grow in the face of change and uncertainty[1].
- The concept of resilience has been used to analyze systems and solve problems in fields such as ecology, psychology, computer science, material science, and disaster management[2,3,4,5,6].
- See back-up slide for definition in different domains

[1] Fiksel, Joseph, “Sustainability and resilience: toward a systems approach,” Sustainability: Science, Practice, & Policy, Vol. 2, No. 2, pp. 1-8, 2006

[2] Folke, C., Carpenter, S., Walker, B., Scheffer, M., Elmqvist, T., Gunderson, L., Holling, C.S., "Regime Shifts, Resilience, and Biodiversity in Ecosystem Management", Annual Review of Ecology, Evolution, and Systematics Vol. 35: 557–581, 2004

[3] Masten, A. S, "Ordinary Magic: Lessons from research on resilience in human development" (PDF). Education Canada 49 (3): 28–32, 2009

[4] Mohammad, A. J., Hutchison, D., Sterbenz, James P.G. "Poster: Towards Quantifying Metrics for Resilient and Survivable Networks", in 14th IEEE International Conference on Network Protocols (ICNP 2006), Santa Barbara, California, USA, November 2006

[5] From: Avallone, Eugene A., et al. eds. Marks' Standard Handbook for Mechanical Engineers, 11th ed., 2007

[6] Shinozuka, M., Chang, S.E., “Evaluating the Disaster Resilience of Power Networks and Grids,” Springer-Verlag, edited by Okuyama, Y., Chang, S.E., 2004

Definitions of resilience in different domains

Domains	**The definition of resilience**
Ecology	The capacity of an ecosystem to respond to a perturbation or disturbance by resisting damage and recovering quickly. Such perturbations and disturbances can include stochastic events such as fires, flooding, windstorms, insect population explosions, and human activities such as deforestation and the introduction of exotic plant or animal species.
Psychology	The idea of an individual's tendency to cope with stress and adversity. This coping may result in the individual "bouncing back" to a previous state of normal functioning, or using the experience of exposure to adversity to produce a "steeling effect" and function better than expected.
Computer science	The ability to provide and maintain an acceptable level of service in the face of faults and challenges to normal operation.
Material science	The property of a material to absorb energy when it is deformed elastically and then, upon unloading to have this energy recovered. In other words, it is the maximum energy per unit volume that can be elastically stored.
Disaster application	The ability of countries, communities and households to manage change, by maintaining or transforming living standards in the face of shocks or stresses - such as earthquakes, drought or violent conflict – without compromising their long-term prospects.

Presentation Outline

- Motivation and Research Objectives
- Previous Work
- Analytical Approach
  - A hierarchical representation of System of Systems (SoS)
  - An interdependency analysis of an SoS using a Bayesian Network with beta distribution
- Synthetic Problem Demonstration
- Conclusion & Future Work

Research Objectives

- This paper aims to provide a methodology that supports pre- and post-milestone B activities by analyzing the impact of requirement changes and system development failure during generation of a system-of-systems capability.
- In other words, this study is focused on tools suitable to analyze uncertainties in systems and the interdependencies between systems and possibly evolving requirements.

An interdependency analysis using a BN

- In analyzing interdependencies, we need to handle:
  - ➢ Inherent uncertainty of systems
  - ➢ Propagation of uncertainty
- A Bayesian Network (BN) can handle these.
- A Bayesian Network (BN) is a directed acyclic graph.
  - ➢ Nodes are the random variables and edges correspond to direct influence of one node on another.
  - ➢ Each variable in the BN model is associated with a conditional probability distribution.
- A BN facilitates the formation of a joint probability:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid P_a(X_i))$$

where $Pa(X_i)$ are the parents of X_i.