Systemic Security and the Role of Hierarchical Design in Cyber-Physical Systems

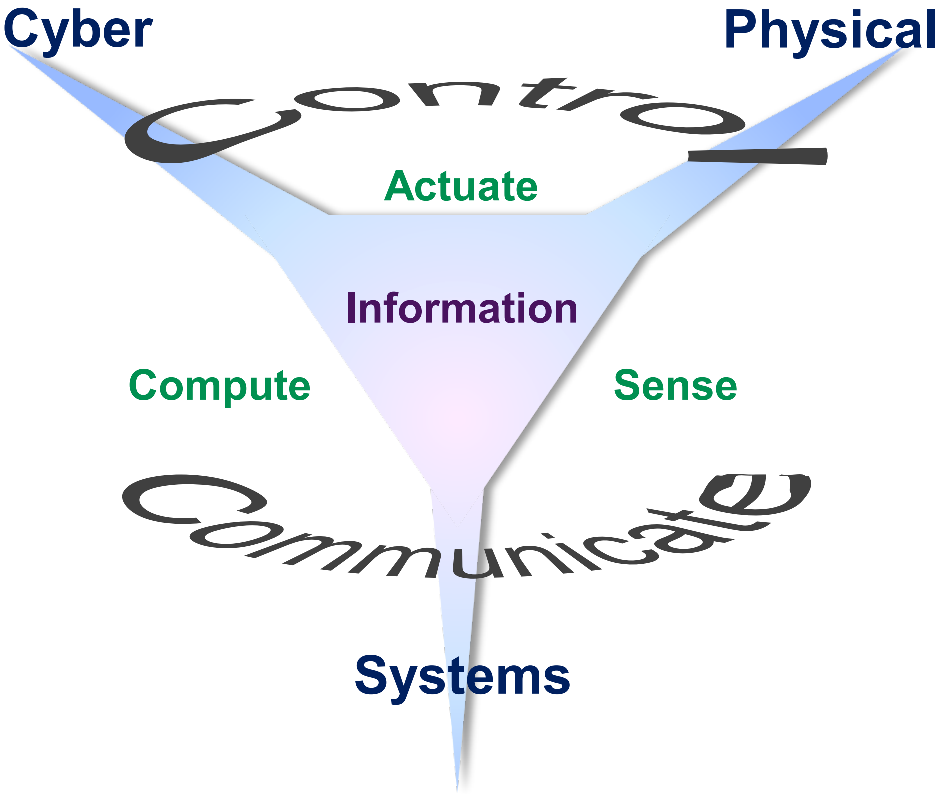
Sponsor: OUSD(R&E) | CCDC

By
Dr. Valerie Sitterle
Mr. Tom McDermott

11th Annual SERC Sponsor Research Review
November 19, 2019
FHI 360 CONFERENCE CENTER
1825 Connecticut Avenue NW, 8th Floor
Washington, DC 20009

www.sercuarc.org

Cyber-Physical Systems



Need

Develop methods to discover and evaluate CPS security vulnerabilities

Purpose

Help evaluate ability of Defense CPS to ***maintain mission-effective capability*** under threat

Help ***design*** an effective control structure that ***reduces adverse events***

Specifically, from a ***Security*** perspective.

Security

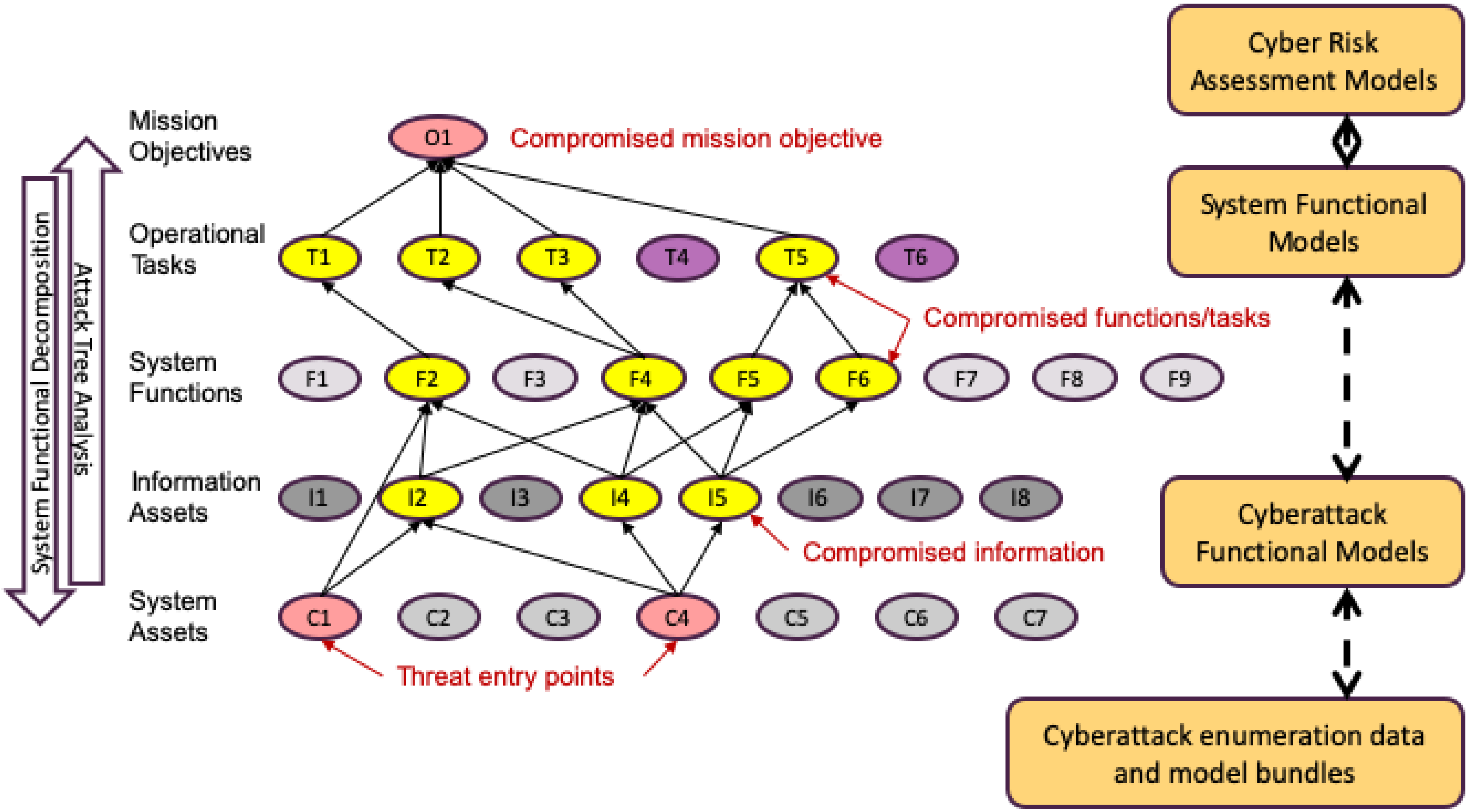
- In broader concept of Resilience, Security concentrates on ***protection from sentient adversary***
- Consider Security a non-functional requirement assessed on how well a given security implementation

  – ***a design pattern*** –

  protects as intended without adversely impacting capabilities
- Threats are focused on ***function***!

- Most work models impact to function and structure separately
- Need ***functional characterization*** to capture system behavior

Layered View of a System for Analysis



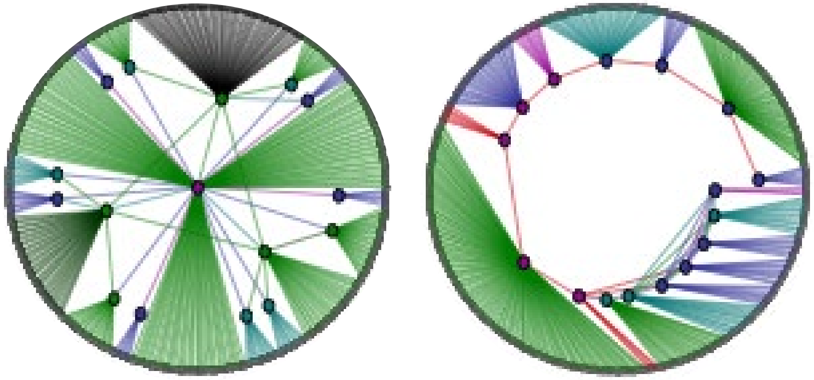
A functional viewpoint is a complement, not a replacement for a structural, component-based view.

Figure adapted from Bodeau, DJ & Graubart, R. Cyber Resiliency Engineering Framework, MITRE Corporation Technical Report MTR-110237, September 2011.

Structure and Function

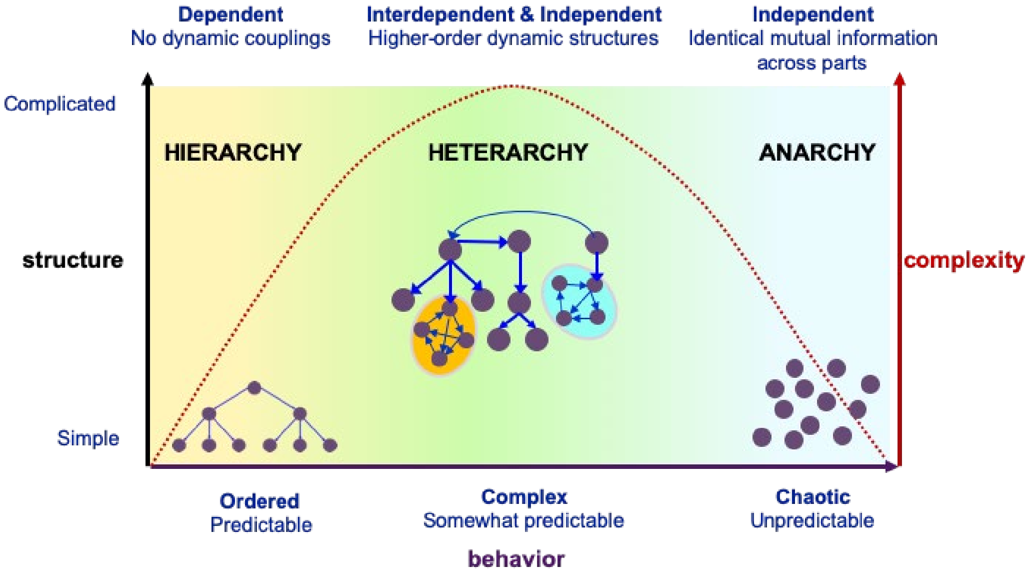
Identical number of nodes, links, and degree distribution.

Image from Li (2005)



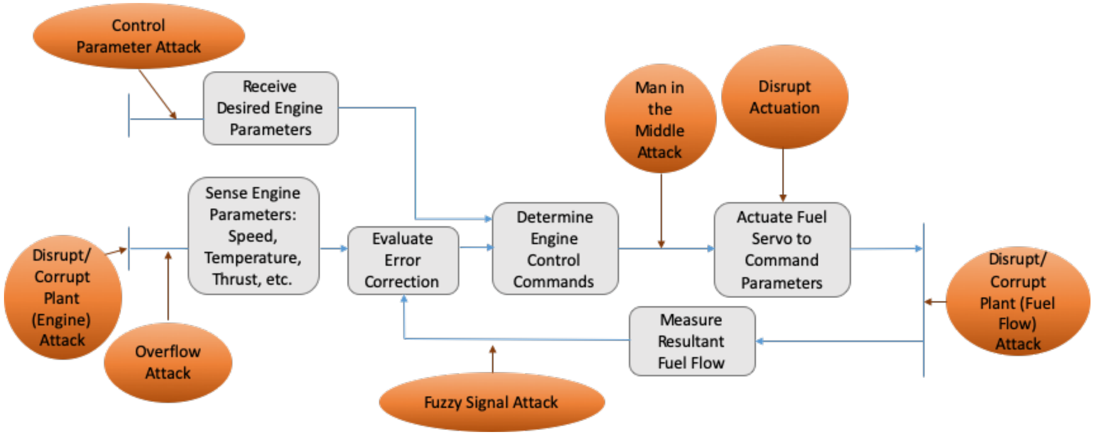
A system's structural characteristics and what processes and behaviors are possible within and as produced by that system are not separable.

- Gap in current MBSE-driven analyses due to heterarchical nature of CPS
  - — Traditional decomposition insufficient
  - — Interdependency makes a threat to a critical system function inseparable from the original system

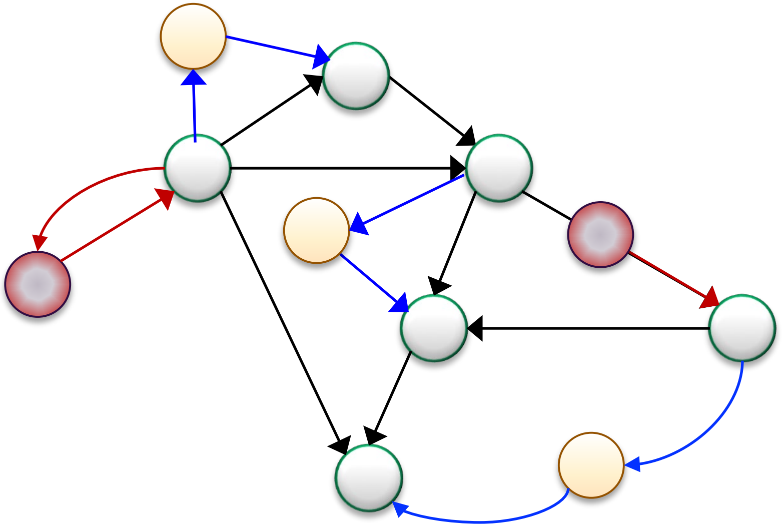


System-Centric vs an Ecosystem

- Functional Perspective
  - What a system ***does***
    - Functions/behaviors/actions
  - ***How*** the system performs purpose
  - How functional behaviors ***interact***


Control Parameter Attack
Receive Desired Engine Parameters
Man in the Middle Attack
Disrupt Actuation
Sense Engine Parameters: Speed, Temperature, Thrust, etc.
Evaluate Error Correction
Determine Engine Control Commands
Actuate Fuel Servo to Command Parameters
Disrupt/ Corrupt Plant (Engine) Attack
Overflow Attack
Measure Resultant Fuel Flow
Disrupt/ Corrupt Plant (Fuel Flow) Attack
Fuzzy Signal Attack

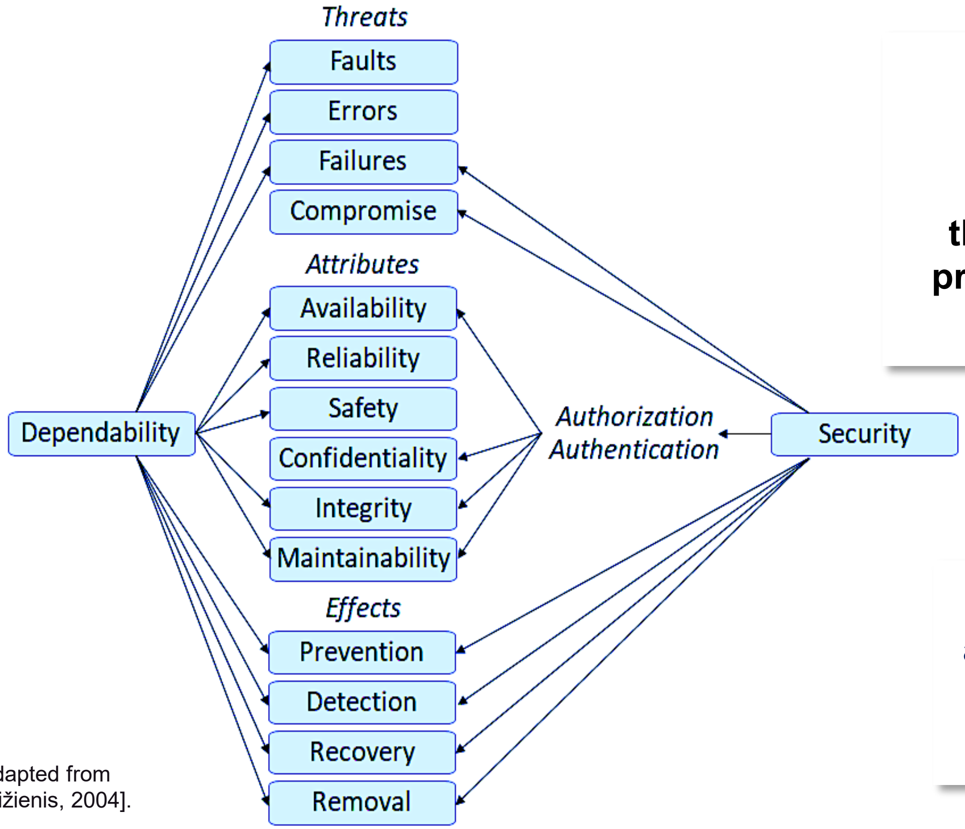
- Functional architecture
  - Topology of functional flow and relationships
  - Reveals how the dynamics associated with these processes and flows propagate through that topology



- Current model-based system design paradigms are ***system-centric***
  - ***"Bolt-on"*** technologies change structure = change function

Security, Dependability, and Trust

- In CPS, many essential system properties such as stability, safety, performance are expressed in terms of physical ***behavior***
  - — System security analysis via models that unify functional topological-behavioral dependencies



[Adapted from Avižienis, 2004].

Can we build a model-based process in concert with existing MBSE practices to produce an evidentiary case that a system is trustworthy with respect to the properties its stakeholders legitimately rely upon within acceptable levels of risk?

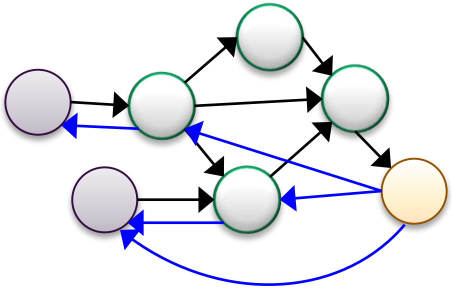
What model-based approaches capture relevant and representative levels of abstraction sufficient to help validate the integrity of the system requirements and the integrity of the design?

RT-204 Research Approach

Presume we have a system model [MBSE]

How do we effectively query that model… to produce relevant system representations (i.e., construct a model transform)?

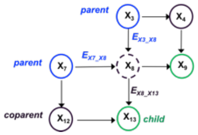
How do we discover what constitutes functions and flows relevant to our analysis?



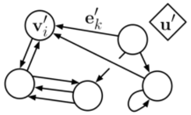
Once we obtain a reduced graph projection of our model, what are efficient approaches to…

Augment with threat vector functional patterns?

Attribute micropatterns for functional state?



Determine if functional state space is preserved?



What did we Learn ?

Transform Single Source-of-Truth Models into a Graph

SysML: XML Metadata Interchange (XMI)

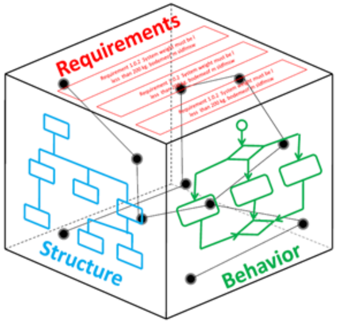
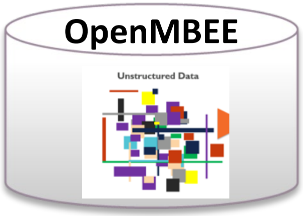
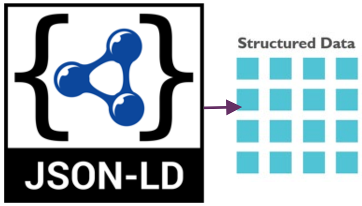


Image from APAN

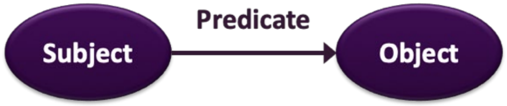
Underlying semantic model in (flat) JSON



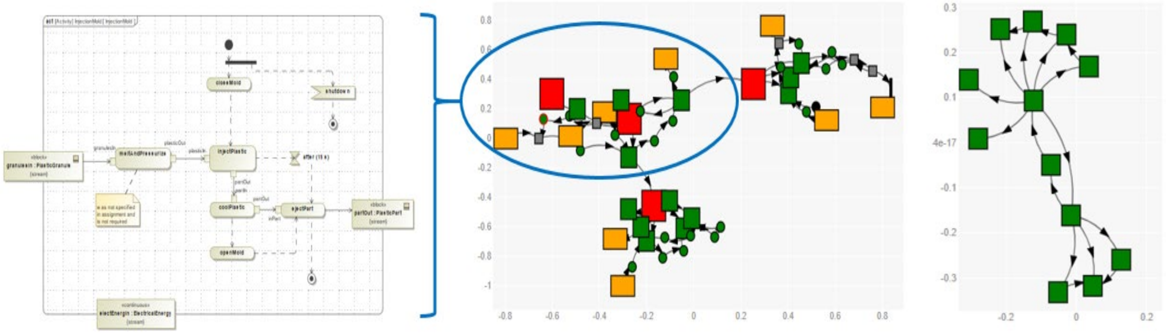
Process OpenMBEE JSON model to produce structured, linked data representation (query-able)



Encoded as set of triples saved into RDF store – Creates an RDF cache of model for semantic querying



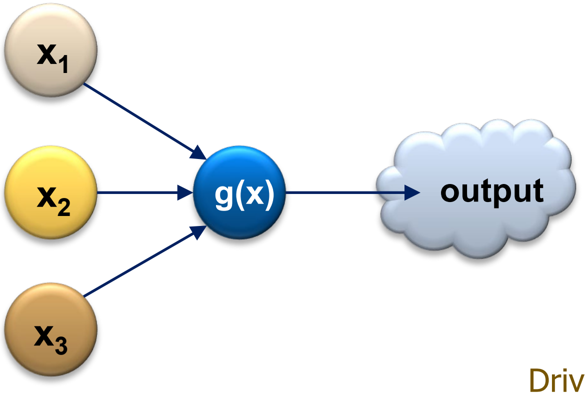
- Focus on extracting formal graph representations of SysML ***Activity Diagrams***
  - Query the RDF graph based on functional patterns (via *object flow*)
  - Compress the resulting graph into an abstract functional representation between *Actions*



Graph Simulation using AI-Derived Concepts

Simulating the evolution of states given an functional model of linked activities:

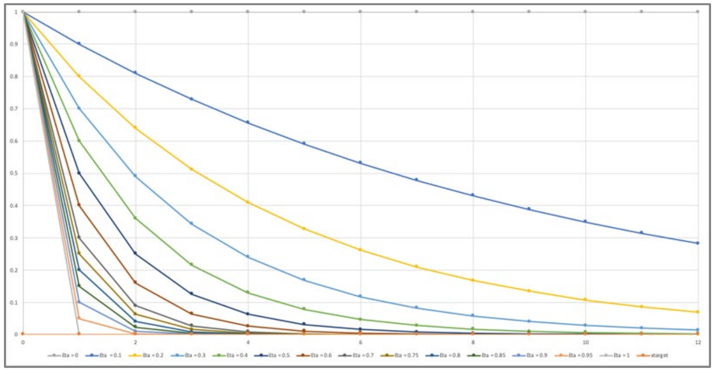
Hybridize the concepts of a Feed-Forward Neural Net with a Markov dependency



$$x_{self}(t) = x_{self}(t\text{-}1) + \eta * g(x_{self}(t\text{-}1), x_{inputs}(t))$$

Shaping parameter to control the rate of change in $x_{self}(t)$

Drives $x_{self}(t)$ toward a target value defined by type of node and input states. Customizable by node type and its intended properties.

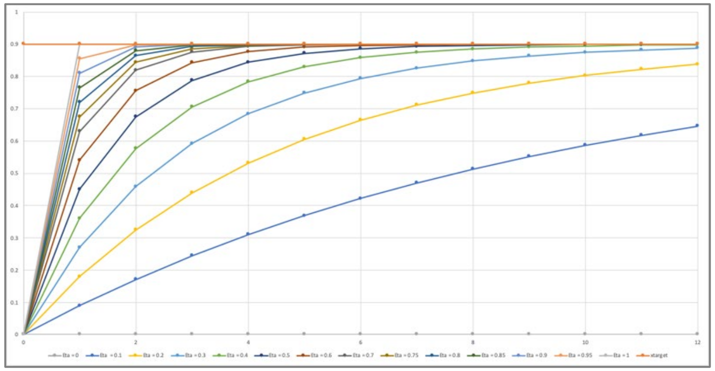


Functional

Degradation

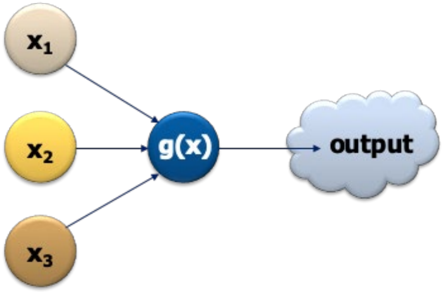
Or

Recovery

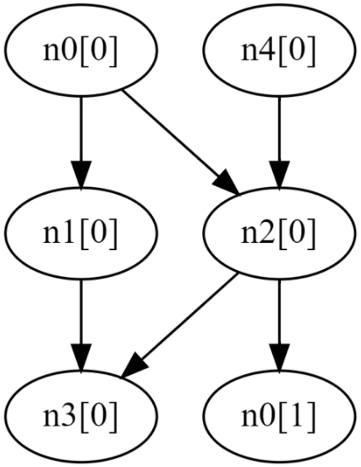


Graph Attribution and Simulation

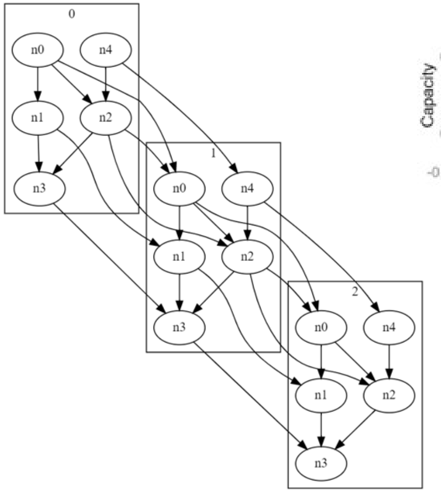
Define auto-attribution of node state functions by type



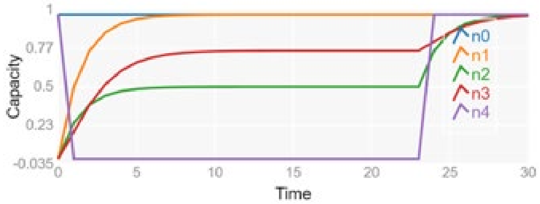
Core of the implementation is the graph template data structure

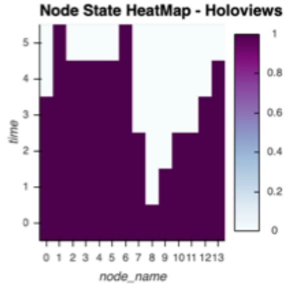


Template expanded for 3 time slices



Evaluating functional state over time



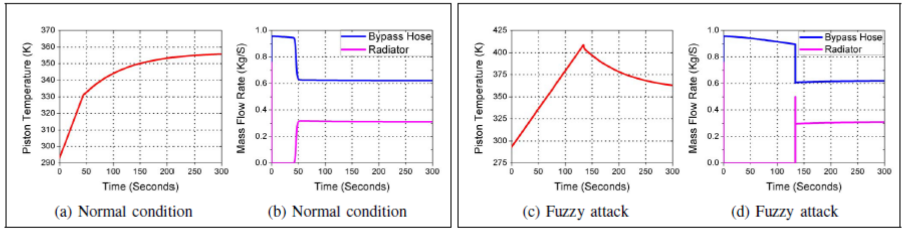


Intra-time and inter-time dependencies captured in the template structure.

Dependencies flow in an autoregressive sense whereby the future state of a node depends on the previous value.

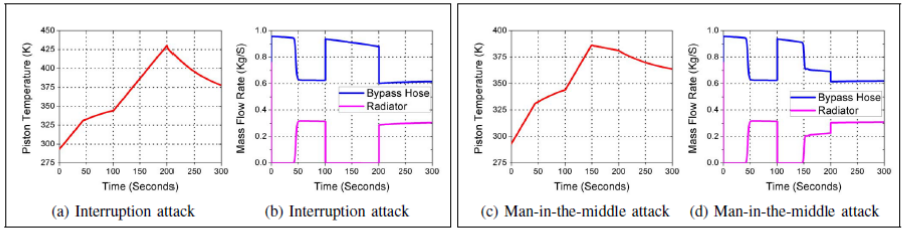
Threat and Security Patterns

- Threat functions will be patterns themselves
- Threats will have differing intent and abilities to achieve that intent
- Future implementation will require timing of simultaneous threats



(a) Normal condition (b) Normal condition (c) Fuzzy attack (d) Fuzzy attack

Fig. 3: Simulation results without attacks and with fuzzy attack



(a) Interruption attack (b) Interruption attack (c) Man-in-the-middle attack (d) Man-in-the-middle attack

Fig. 4: Simulation results with interruption and man-in-the-middle attack

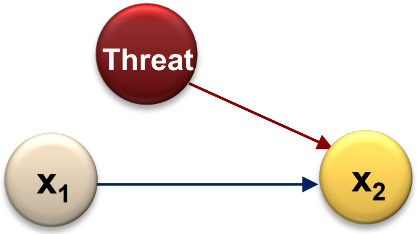
[Rashid, et al (2017)]

Designed system must be augmented with threat

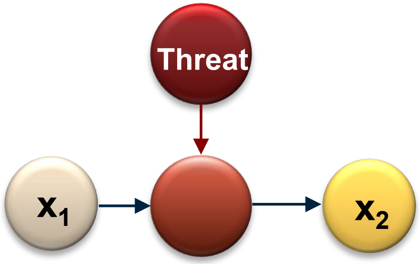
and

security patterns

Functional behavior within context of the **ecosystem**

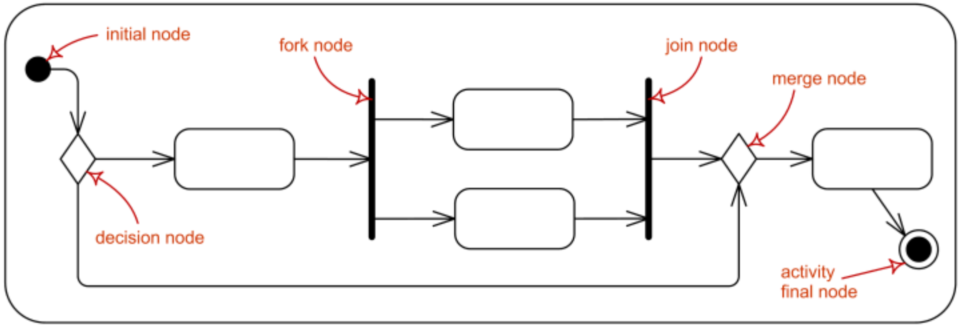


or



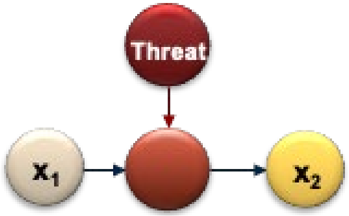
What have we learned so far?

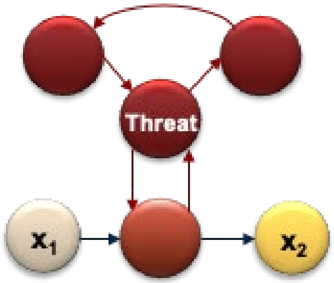
- Need to look more deeply into ***how to model system functionality*** to develop simulations of cyber-physical systems.
  - — Object flow, control flow, other functional MBSE formalisms, etc.
  - — What types of elements, at what level of decomposition, using what consistent ontology, with which types of MBSE implementations (join, merge, etc.)?
  - — How to define extensible architecture and at what level of decomposition?



[Image from https://www.uml-diagrams.org/activity-diagrams-controls.html]

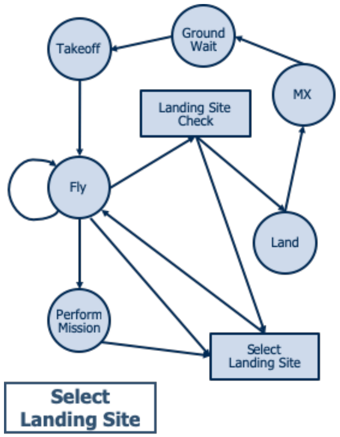
- How can various ***threat types be best expressed as functional patterns*** themselves?
  - — Monumental gap in current understanding and practice
  - — Need to develop a consistent, repeatable way to extract relationships between threat vectors and functional assets common to cyber-physical systems

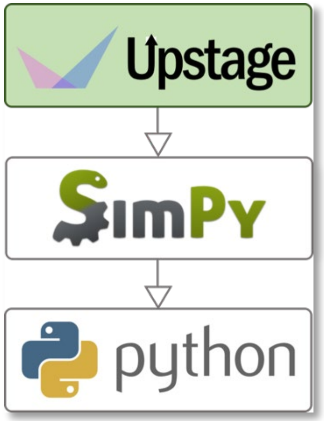




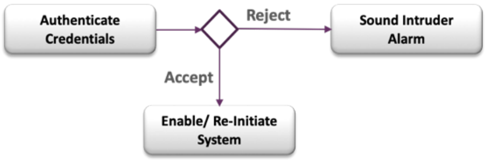
Next Steps: Functional Analysis

- Mature ***implementation of functional abstraction***
  - — Graph node/edge insertion for threats and security patterns
  - — Node type differentiation (e.g., Sentry, Redundancy, etc.)
  - — Addressing Scalability and Timing -> **UPSTAGE**
    - ○ Functional graph + Discrete Event Simulation
  - — Defining and building from a library

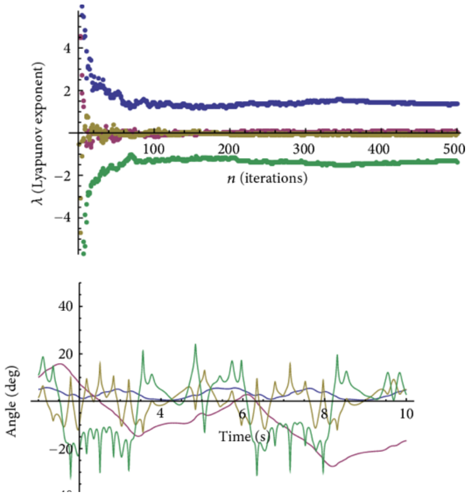




  - — Decision node abstractions



- ***Develop a preliminary set of metrics*** and/or methods for analyzing the outputs of a dynamic simulation of CPS when represented as a dynamic state graph
  - — Most graph metrics designed for static concepts → Combine graph metrics/ concepts and time series analysis
    - ○ Stable, vary significantly, gradually trend toward capability, restored capability, gradual failure, rapid failure?

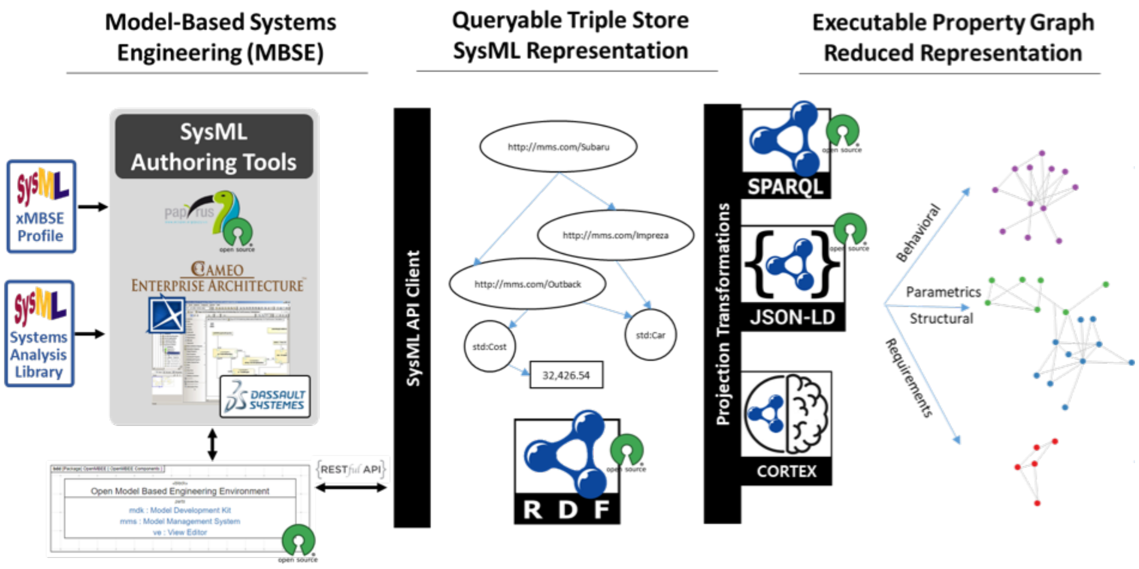


[Example Time Series Data for Illustration Purposes from Yunping et al (2013)]

Next Steps: Overall Implementation

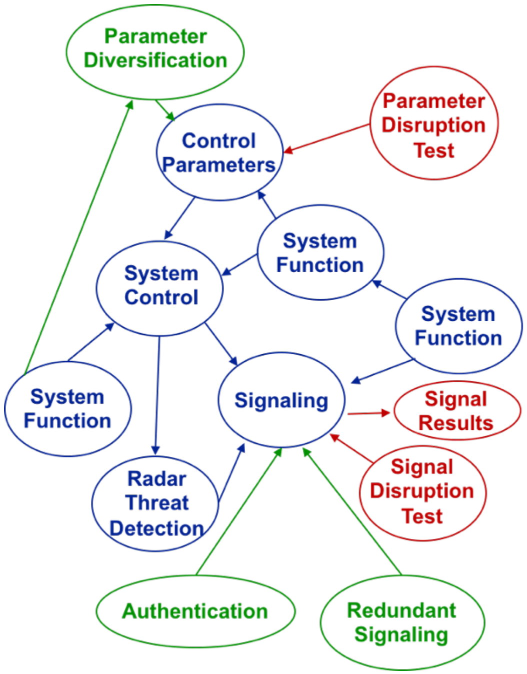
- Determine an efficient approach, ***synergistic with the state of development and compatibility (where it exists) across current MBSE tools*** whereby SEs can efficiently and effectively:
  - — Define the necessary CPS functionality at a relevant level of abstraction, and
  - — Analyze system outside of MBSE tools to produce meaningful evolutions of state space dynamics

  - — Take a ***neutral starting point*** for bi-directional conversion between formal MBSE architectures and dynamic simulation using other open-source technologies
  - — Suggest a ***roadmap*** for a viable, efficient path forward



Best Practices and Systemic Security

- What is different?
  - — A ***functional view*** of a system/threat/security ***ecosystem***
- Architectures as true analytical tools, not just templates
- Enables traceable analysis of functional dynamics:
  - — Functional failure
  - — Compromise
  - — Corruption
  - — Where protection is most critical
  - — Impact on intended function and preservation of function
- Aim to answer if approach can produce a path forward to realize a ***test framework for assurance***



The key is to create "safe" designs, not respond simply to known threats.

Acknowledgements

Georgia Tech Research Institute

- Ms. Erika Brimhall
- Dr. Dane Freeman
- Dr. Zach Welz
- Mr. Nicholas Bollweg

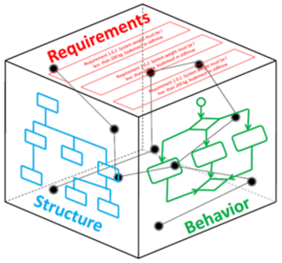
SYSTEMS ENGINEERING RESEARCH CENTER

- Mr. Tom McDermott
- Ms. Megan Clifford

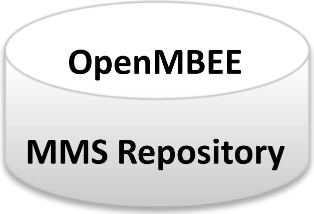
Portions of this material are based upon work supported, in whole or in part, by the United States DoD through the Systems Engineering Research Center (SERC) under Contract HQ0034-13-D-0004. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology. The views and conclusions are those of the individual authors and participants, and should not be interpreted as necessarily representing official policies, either expressed or implied, of the DoD, any specific US Government agency, or the US Government in general.

RT204: Overall Flow

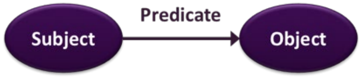
Formally expressed system model



OpenMBEE linked data architecture



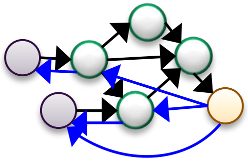
Query-able graph-structured data model



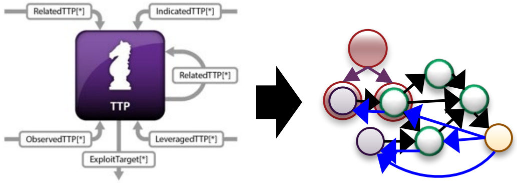
Function and data flow query/ discovery

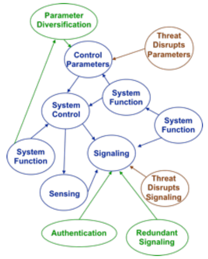
SPARQL

Reduced Graph functional architecture representation

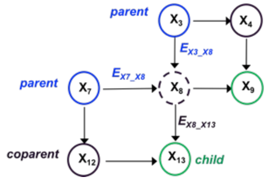


Threat vector functional patterns relevant to system



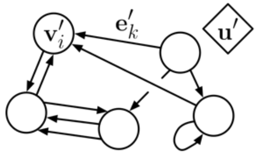


Reduced ecosystem graph



Attribute with dynamic state abstractions

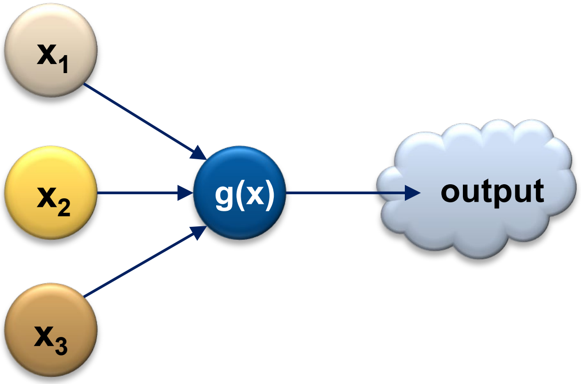
Is functional state of system preserved?



Graph Simulation using AI-Derived Concepts

Simulating the evolution of states given an functional model of linked activities:

Hybridize the concepts of a Feed-Forward Neural Net with a Markov dependency



$$x_{self}(t) = x_{self}(t-1) + sgn(\Delta) * \eta * |\Delta|$$

$x_{self}(t)$: Updated value of functional state of current node, [0, 1]

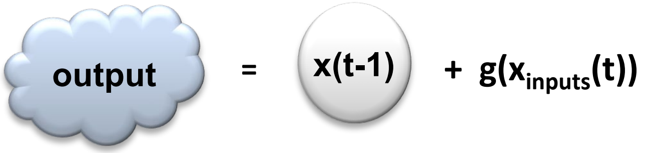
$x_{self}(t-1)$: Initial value of functional state of current node, [0, 1]

Δ: The difference in the current node state and the target state, ($x_{self}(t-1)$ - x_{target})

$sgn(\Delta)$: The sign, or signum, function of Δ.

η: Eta, the shaping parameter controlling the rate of change in $x_{self}(t)$

$|\Delta|$: The absolute value of Δ (also expressible as ($\Delta * sgn(\Delta)$).



output
=
x(t-1)
+ $g(x_{inputs}(t))$