Components, Compositionality and Architectures for Networked CPS

John S. Baras
Institute for Systems Research
Department of Electrical and Computer Engineering
Fischell Department of Bioengineering
Applied Mathematics, Statistics and Scientific Computation Program
University of Maryland College Park

London CPS Workshop
October 20, 2012
Notre Dame University

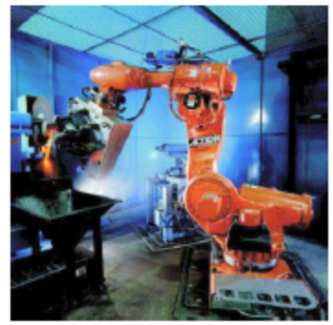
Acknowledgments

- **Joint work with:** Mark Austin, Shah-An Yang, Ion Matei, Kiran Somasundaram, Vahid Tabatabaee, Kaustubh Jain, Senni Perumal, George Theodorakopoulos, Dimitrios Spyropoulos, Yuchen Zhou, Leonard Petnga, Brian Wang, Pedram Hovareshti, Tao Jiang
- **Sponsors:** NSF, AFOSR, ARO, ARL,NSF, DARPA, SRC, NIST, Lockheed Martin, Northrop Grumman, Telcordia

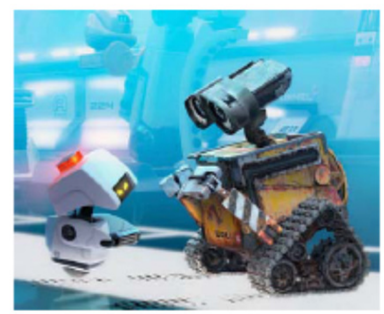
NETWORKED CPS MBSE

Automotive

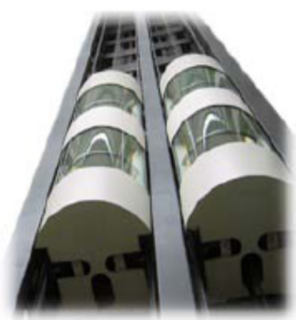
Aeronautics



Industrial automation



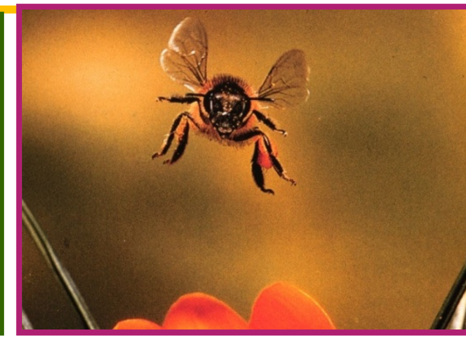
Robotics

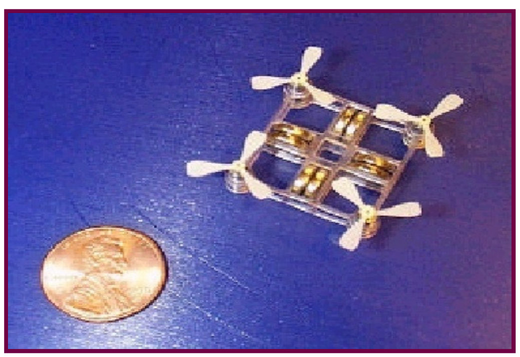


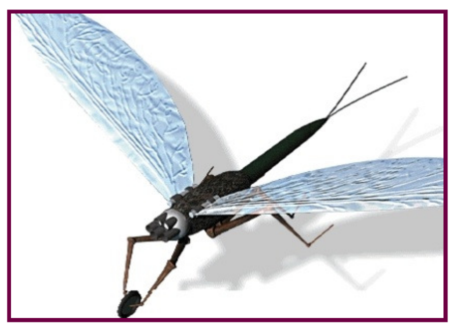
Elevators

Building automation

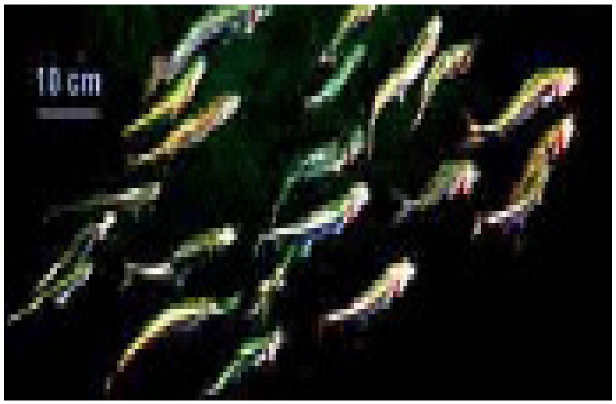
Collaborative Robotic Swarms



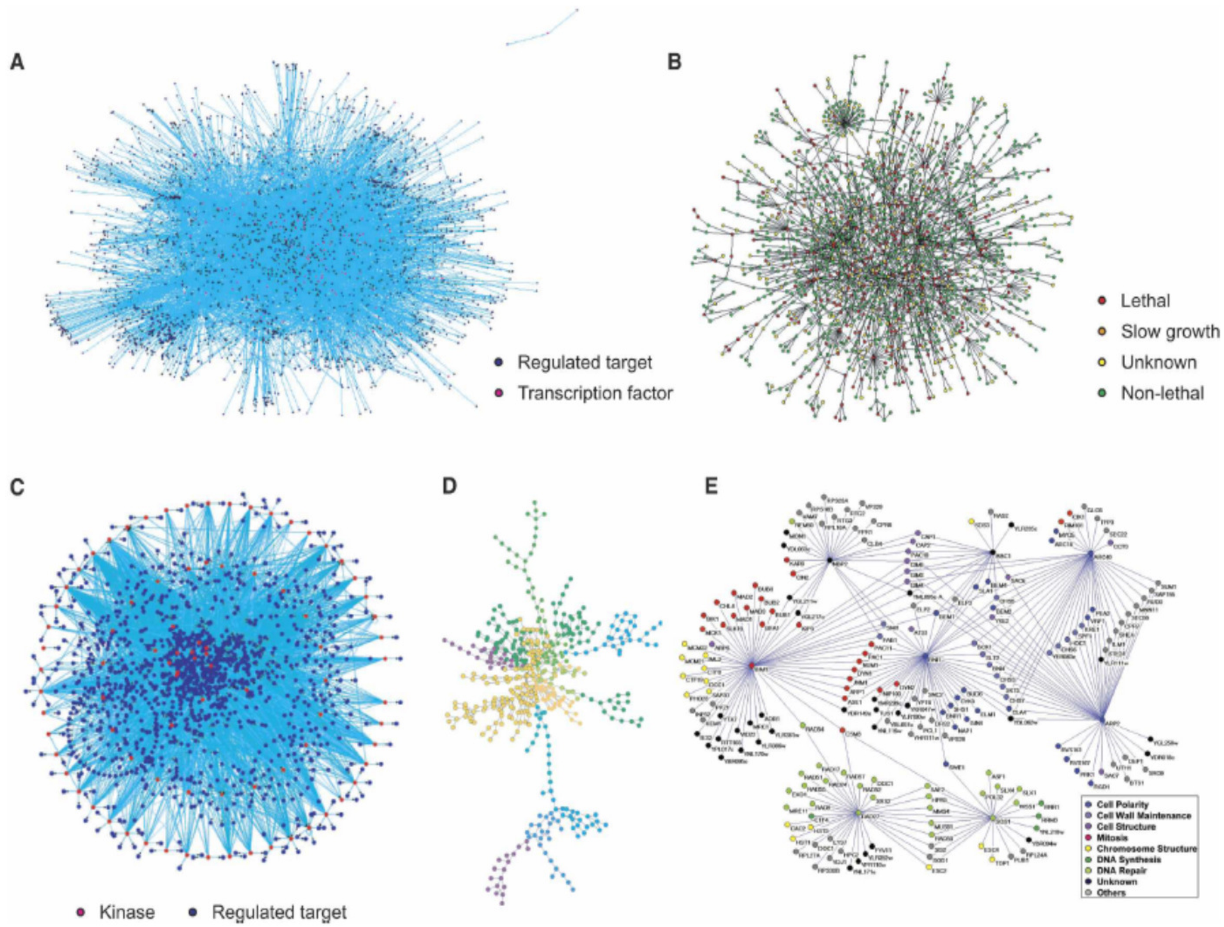




Biological Swarms



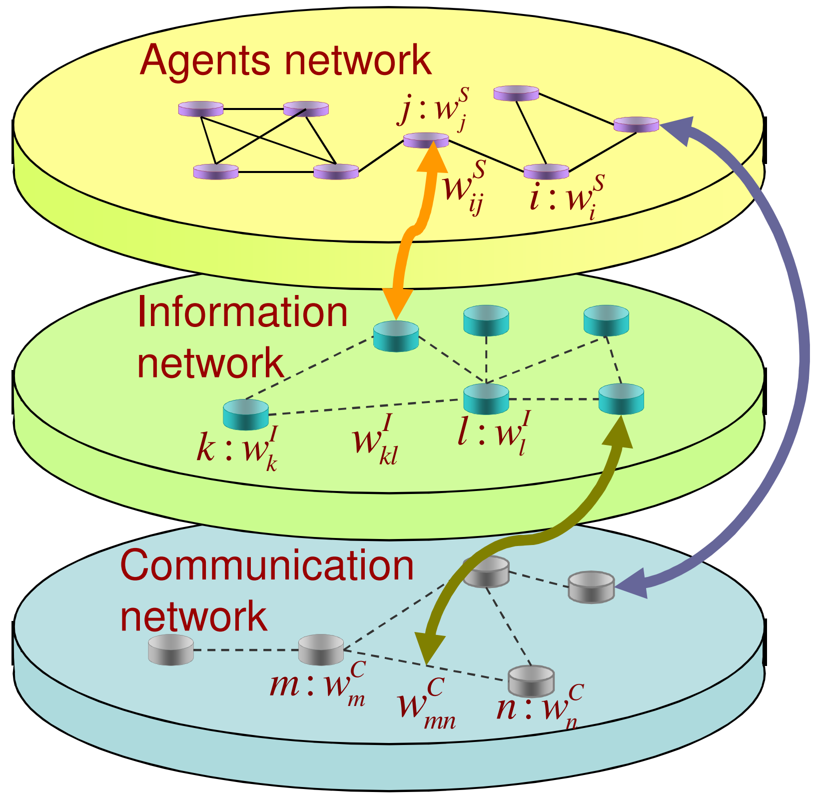
Biological Network Types



Examples of biological networks: [A] Yeast transcription factor-binding network; [B] Yeast protein -protein interaction network; [C] Yeast phosphorylation network ; [D] *E. Coli* metabolic network ; [E] Yeast genetic network ; Nodes colored according to their YPD cellular roles [Zhu et al, 2007]

Multiple Interacting Dynamic Multigraphs

- Multiple Interacting Graphs
  - *Nodes*: agents, individuals, groups, organizations
  - Directed graphs
  - *Links*: ties, relationships
  - Weights on links : value (strength, significance) of tie
  - Weights on nodes : importance of node (agent)
- **Value directed graphs with weighted nodes**
- **Real-life problems: Dynamic, time varying graphs, relations, weights, policies**



Networked System architecture & operation

Three Fundamental Challenges

- **Multiple interacting dynamic multigraphs involved**
  - ***Collaboration multigraph***: who has to collaborate with whom and when.
  - ***Communication multigraph***: who has to communicate with whom and when
- **Effects of connectivity topologies**:
  Find graph topologies with favorable tradeoff between performance improvement (**benefit**) of collaborative behaviors *vs* **cost** of collaboration
  - **Small word graphs** achieve such **tradeoff**
  - **Two level algorithm** to provide efficient communication
- Need for **different probability models** – the classical Kolmogorov model is **not correct**
  - Probability models over logics and timed structures
  - Logic of projections in Hilbert spaces – not the Boolean of subsets

A Network is …

- A collection of nodes, agents, …

  that **collaborate** to accomplish actions, gains, …

  that cannot be accomplished with out such collaboration

- Most significant concept for **dynamic autonomic networks**

The Fundamental Trade-off

- The nodes **gain** from collaborating
- But collaboration has **costs** (e.g. **communications)**
- **Trade-off: gain from collaboration *vs* cost of collaboration**

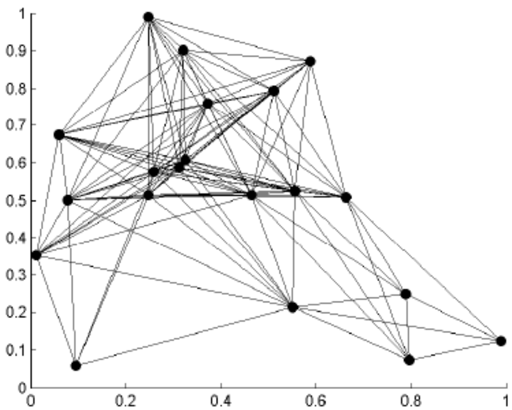
Vector metrics involved typically

→ **Constrained Coalitional Games**

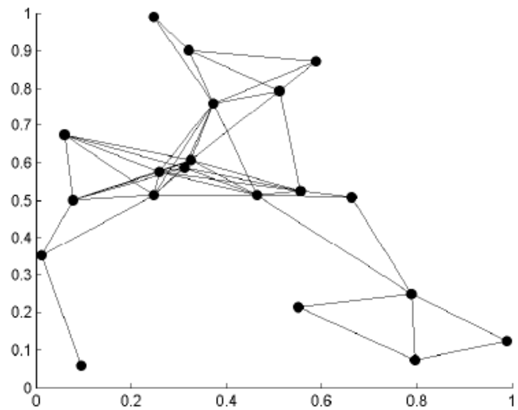
- **Example 1**: **Network Formation** -- Effects on Topology
- **Example 2**: **Collaborative robotics, communications**
- **Example 3**: **Web-based social networks and services**
- **Example 4**: **Groups of cancer tumor or virus cells**

• • •

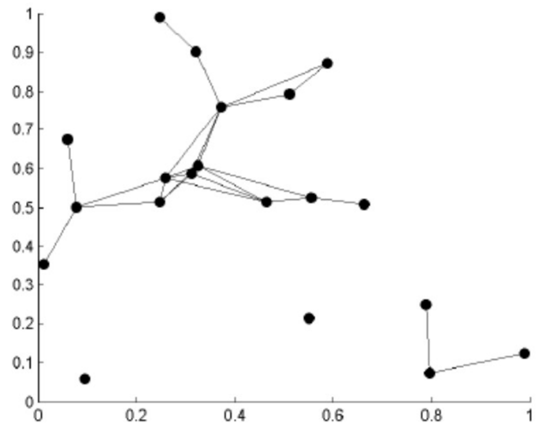
Topologies Formed



(a) P = 0.5 (low cost); complete graph



(b) P = 2 (middle cost); small world topology



(c) P = 4 (high cost); partitioned network

Distributed Algorithms in Networked Systems and Topologies

- Distributed algorithms are essential
  - Agents **communicate with neighbors,** share/process information
  - Agents **perform local** actions
  - **Emergence** of global behaviors
- **Effectiveness** of distributed algorithms
  - The **speed** of convergence
  - **Robustness** to agent/connection failures
  - Energy/ communication **efficiency**
- **Design problem**:

  Find graph topologies with favorable tradeoff between performance improvement (**benefit**) *vs* **cost** of collaboration
- **Example: Small Word graphs** in consensus problems

Consensus problems

- A Simple model:

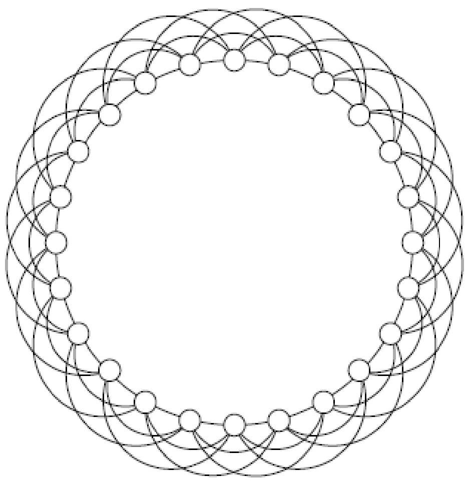
$$\theta_i(t+1) = f_{ii}(t)\theta_i(t) + \sum_{j \in N(i)} f_{ij}(t)\theta_j(t)$$

$$\forall i \in \{1, ..., n\}: \qquad \sum_j f_{ij} = 1$$

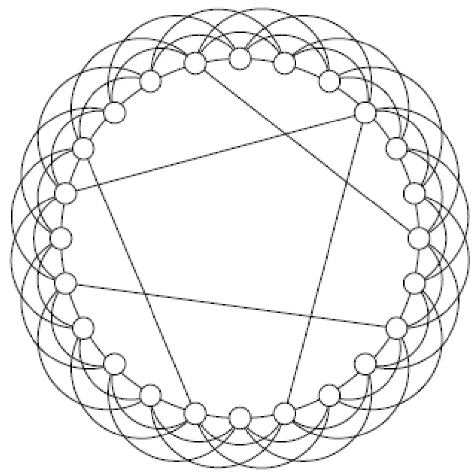
$$\forall i, j \in \{1, 2, ..., n\}: \qquad f_{ij} \geq 0$$

$$\forall i \in \{1, ..., n\}: \qquad f_{ii} \geq \alpha > 0$$

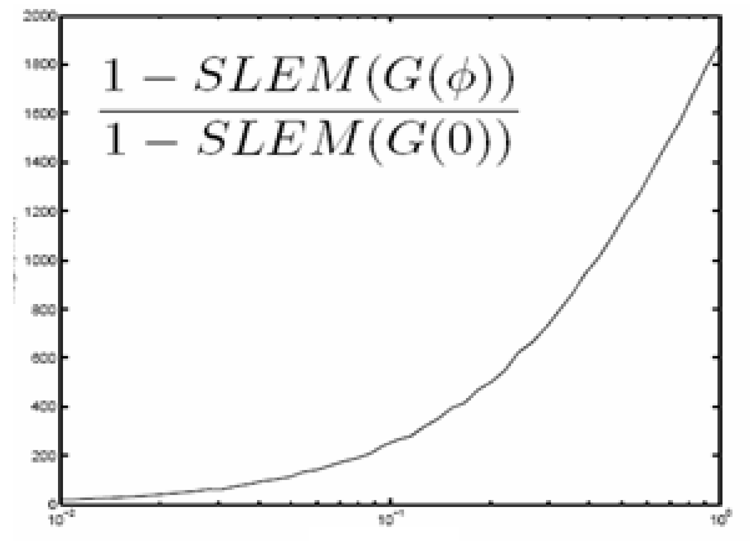
Small World Graphs



Simple Lattice
C(n,k)



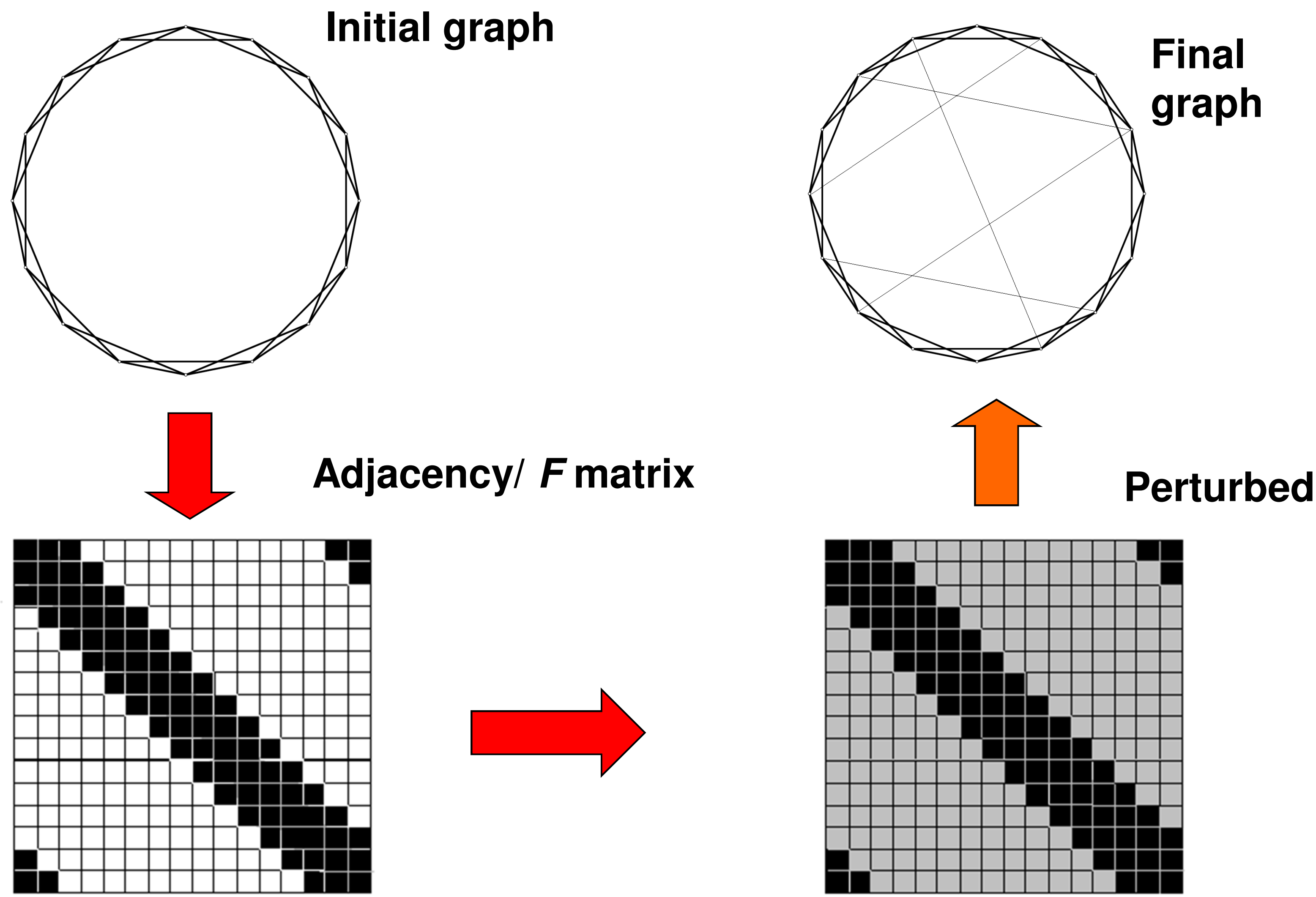
Small world: Slight variation adding $nk\Phi$



\phi
C(1000,5)

Adding a **small portion** of well-chosen links →
significant increase in convergence rate

Mean Field Explanation and Perturbation Approach

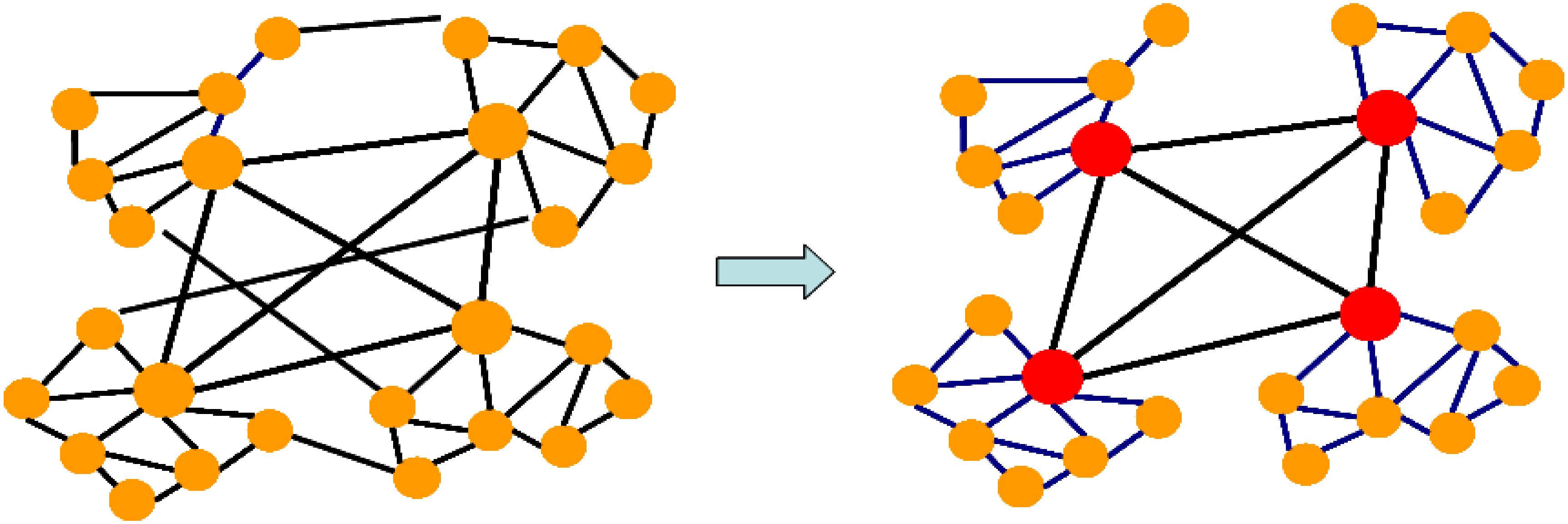


Distributed self - organization

Determine leaders → Find influence → Remove unnecessary links

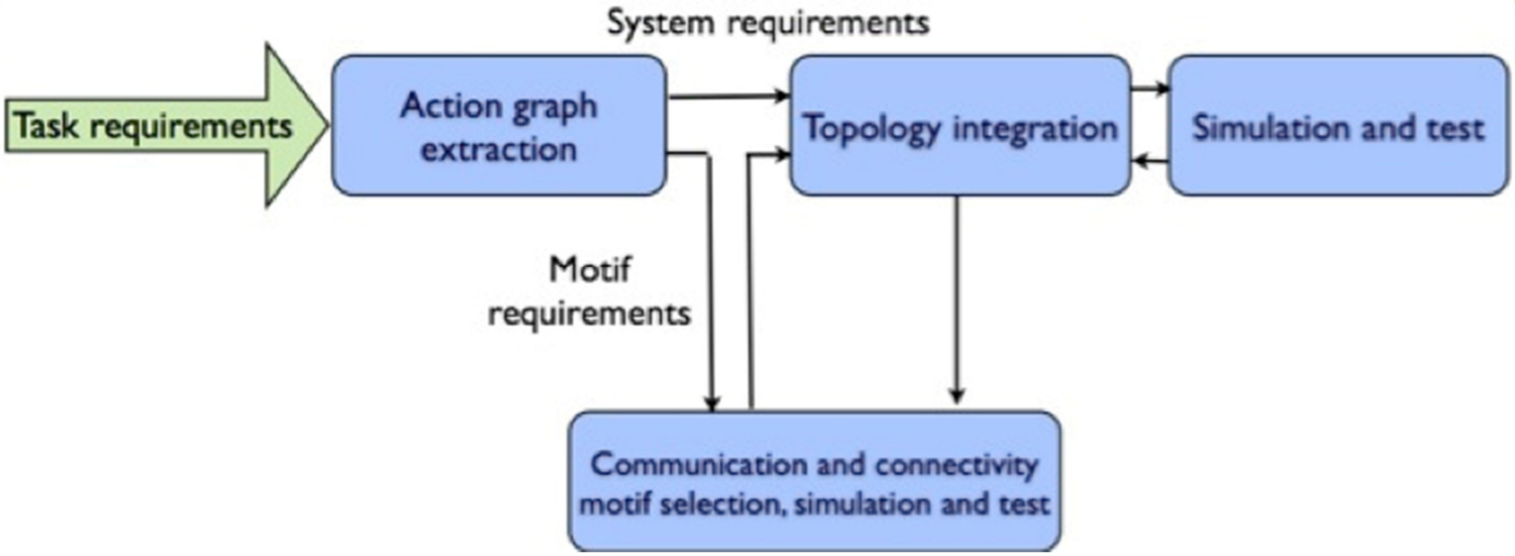
Semi-decentralized

Decentralized: Dirichlet problem on the graph



Goal: design a scheme that gives each node a vector of compact global information

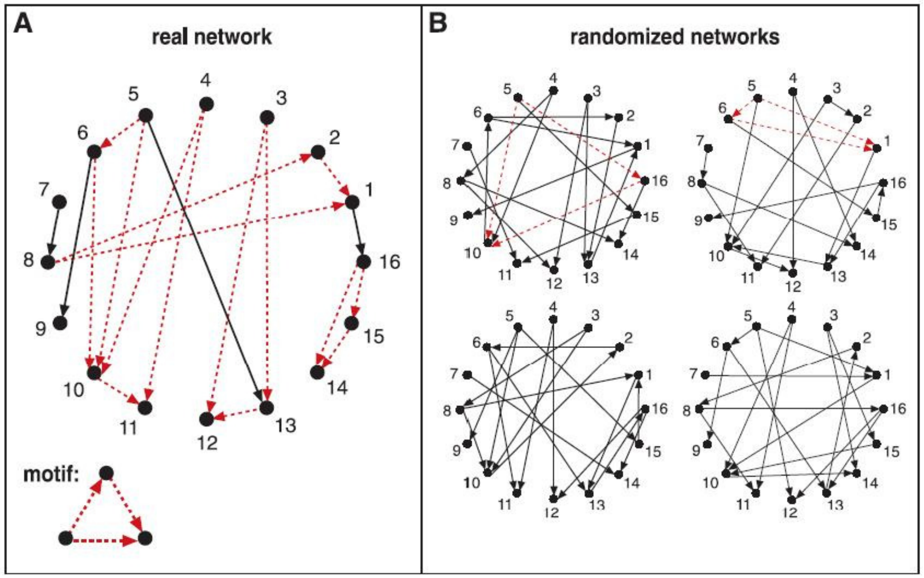
Task-oriented Topology Design



- Based on extracting efficient local graphs
- Dependence on tasks and environment is captured via local “motif” extraction algorithm
- Topology integration is done via considering graph theoretic measures and practical constraints

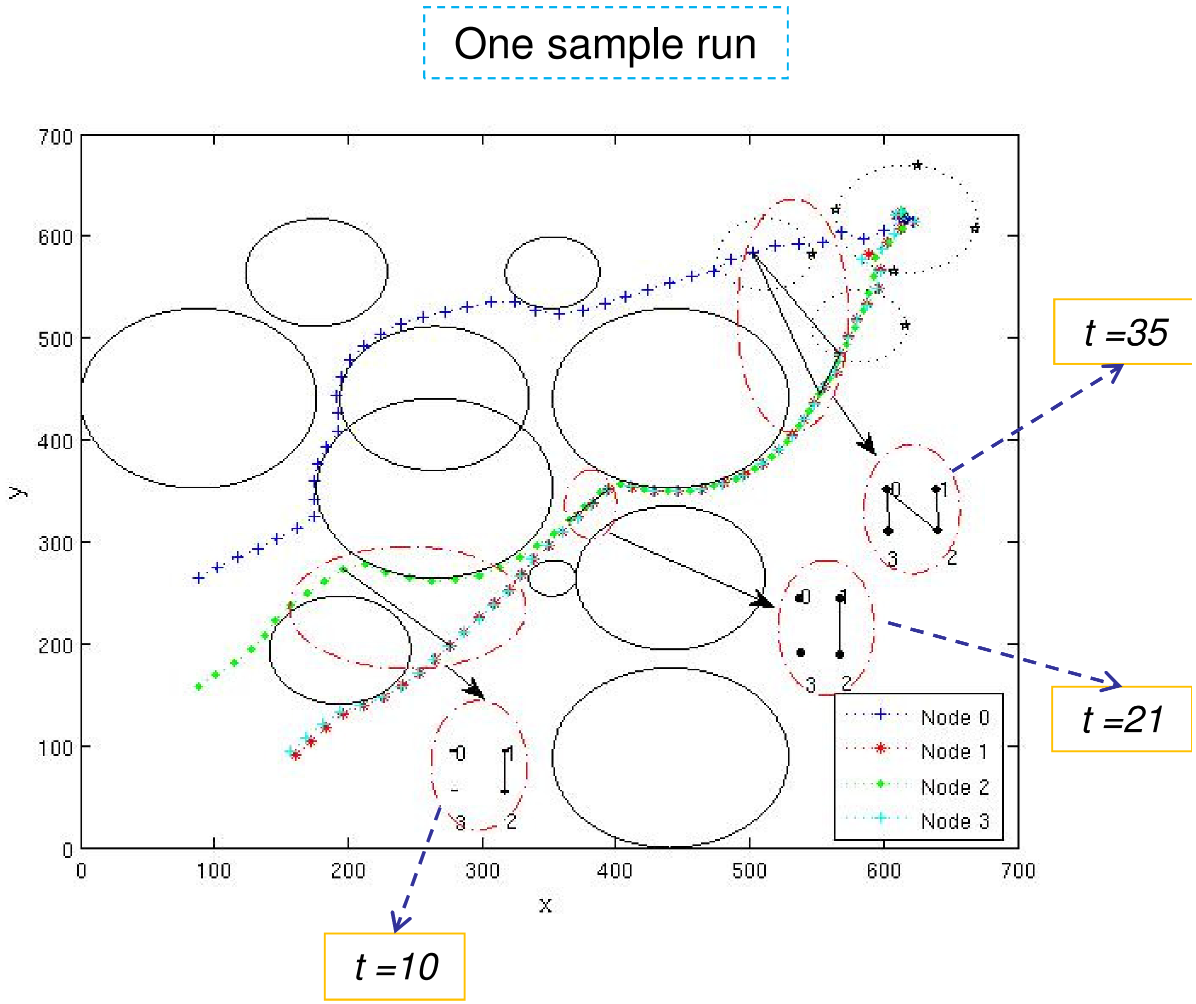
Network Motifs

- Introduced in the context of biological networks
- Subgraphs that appear more than random in a network [Milo et al. 2002, Alon 2007]
- Result of task related global constraints on network structure
- Extension to dynamic networks
- 3 or 4 node subgraphs [Jamakovic et al. 2009]



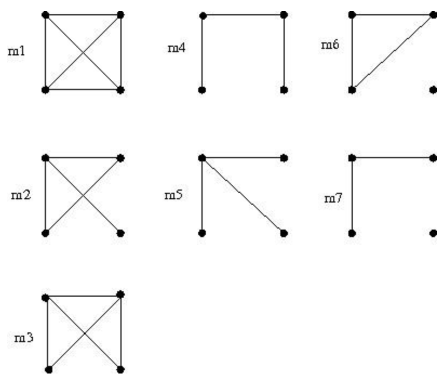
Simulation (1): Communication Graph

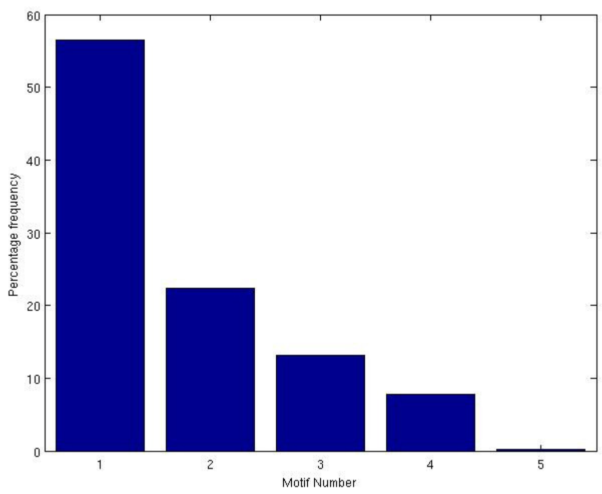
- 4 nodes with a given action graph
- 700mX700m terrain with the target at (670,670)
- 10 uniformly generated obstacles
- 6 moving threats circling around to protect the target
- PHY layer: the Fresnel model
- MAC: IEEE 802.11 CSMA/CA



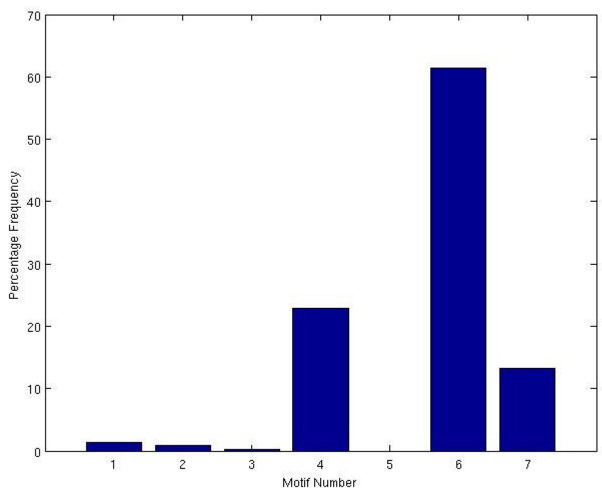
Simulation (2): Percentage of Occurrence

- Series of 100 independent simulations





Connectivity Motifs



Communication Motifs

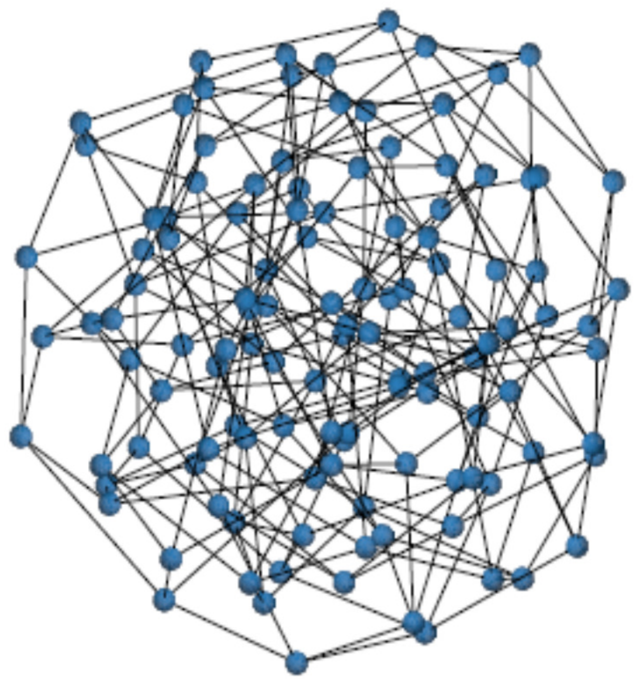
Expander Graphs

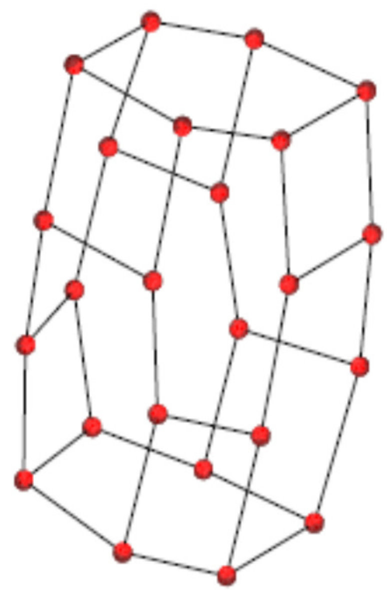
- Fast synchronization of a network of oscillators
- Network where any node is "nearby" any other
- Fast 'diffusion' of information in a network
- Fast convergence of consensus
- Decide connectivity with smallest memory
- Random walks converge rapidly
- Easy to construct, even in a distributed way (ZigZag graph product)
- Graph G, ***Cheeger constant h(G)***
  - All partitions of G to S and S^c,

    $h(G)$=min (#edges connecting S and S^c) /

    (#nodes in smallest of S and S^c)
- (k, N, ε) **expander** : $h(G) > \varepsilon$; **sparse but locally well connected** **(1-*SLEM*(*G*) increases as $h(G)^2$)**

Expander Graphs – Ramanujan Graphs





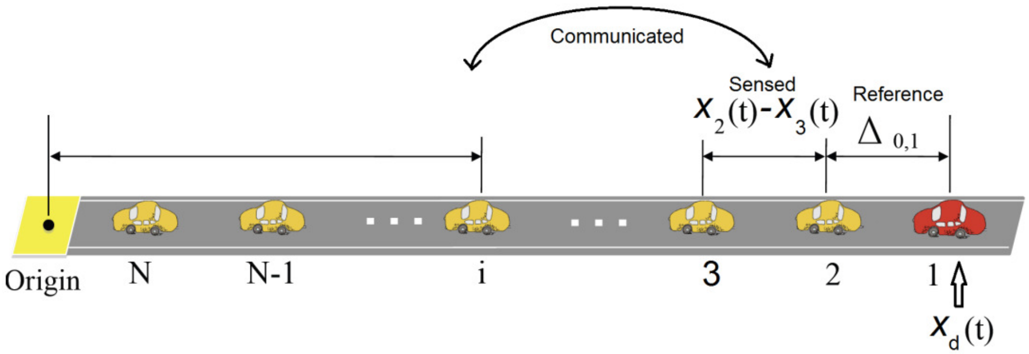
Desirable Network Configurations: Information Patterns for Distributed Control

- Most of the literature in distributed control is devoted to answering the following question- Given a plant, a set of controllers and an information exchange pattern amongst the controllers, when is the optimal controller linear or the synthesis convex?
- Sufficiency conditions like nested information structures and quadratic invariance that give an affirmative answer are known.

We are interested in the following design question-
Given a plant and a set of controllers, design a 'minimal' information exchange pattern that provides desirable control performance.

Main Obstacles
- Optimizing over information patterns is combinatorially hard → Understand features of the 'right' information pattern
- Given an information pattern, controller synthesis is not necessarily linear/convex → Make context dependent simplifying assumptions

Vehicle Platooning Problem



Vehicles have identical linear dynamics $\ddot{x}_i = u_i$

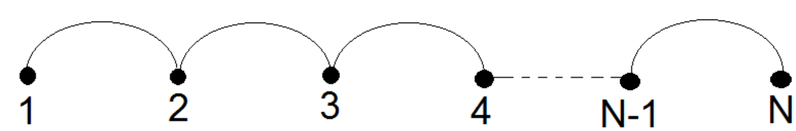
Controller *i* applies linear feedback law based on information available to it:

$$u_i = \frac{1}{deg(i)} \sum_{j \in \mathcal{N}(i)} [-k(x_i - x_j - \Delta_{i,j}) - b(\dot{x}_i - \dot{x}_j)]$$

$$+ \delta(1,i)[-k(x_1 - x_{1,d}) - b(\dot{x}_1 - \dot{x}_{1,d})]$$

Control objective: maintain reference inter-vehicle spacing under the constraint that individual control is function of only the information available to that individual and the lead vehicle alone is provided the desired trajectory information $x_d(t)$

'Local' information patterns i.e. based only on sensed information from predecessor and follower lead to the *information graph* ➡



Expanders as Information Patterns for the Vehicle Platooning Problem*

Theorem: Let λ_{min} be the second smallest eigenvalue of the normalized Laplacian L. Then

$$\frac{\lambda_{min}}{4N} < \gamma_{min} \leq \lambda_{min}$$

- It is immediately clear that the stability margin can be improved to O(1/N) with the expander as the information pattern against the $O(1/N^2)$ that results from local patterns
- Next steps
  - Effects on other metrics of performance like string stability, coherence etc.
  - Demonstrating examples of expanders as information patterns

* Menon A., Baras, J.S. ," Information patterns that improve stability margins in the 1-D vehicle platooning problem: Expander families", 2012 NecSys

"Optimal" Network Topologies

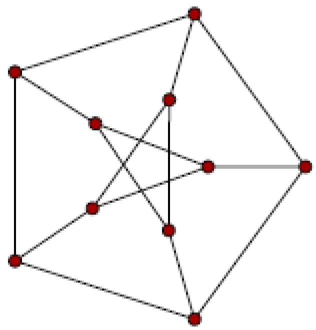
- Want to design a computer network, an infrastructure or communication network, an artificial neural network, etc.
- Restrictions: number of nodes, efficient number of links (e.g. sparsity), optimal topologies for tasks
- Examples: fast synchronization in neural networks, communication networks where nodes are "nearby" each other, optimal dissemination (by dispersion) of packets in a network but avoid creation of 'elephant' hubs

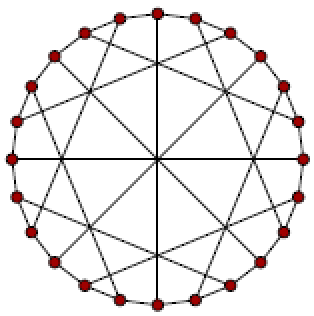
Construction of Networks by Computational Optimization

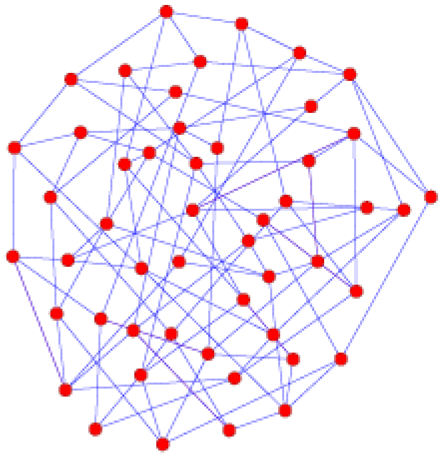
- (Donetti et al 2005) Start with a random network with the desired number of nodes N and average connectivity degree k_{av}, and perform successive rewirings, to get larger and larger spectral gaps
- Employ a simulated annealing or other algorithms
- For small N (less than 30) unique topologies result -- Not so for large N
- Complete analysis as a dynamical system open

Construction of Networks by Computational Optimization

Examples of resulting topologies







Networked Systems – New Probability Models

- *Interaction between information and control*
  - Controllers communication via "signaling strategies"
  - "information neighborhoods" for controllers
  - cost of information versus cost of control
- Despite pioneering work by Witsenhausen and others (formulations and results on the separation of the use of information (estimation) and control),
  - there does not exist todate a satisfactory formulation of the joint "optimization" problem in information flow and control
  - Important to develop theories that treat control strategies and information patterns in a balanced manner

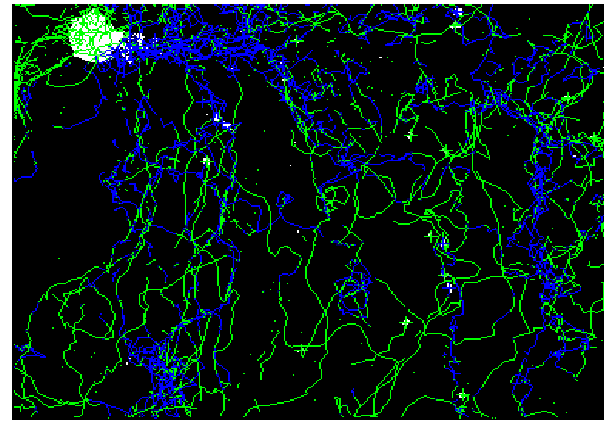
New Probability Models (cont.)

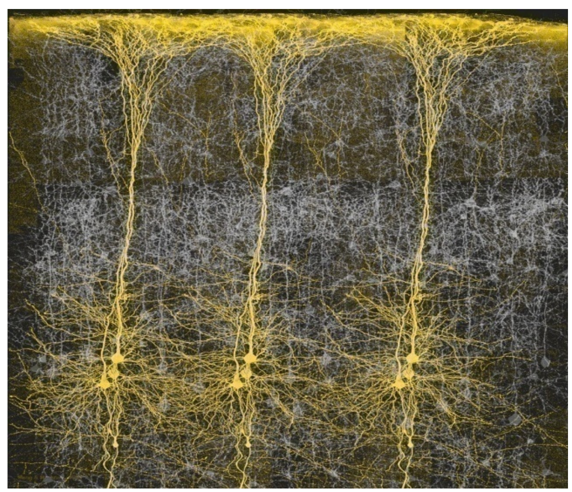
- *Interactions between measurements by different agents and between system dynamics and measurements*
  - Akin to very strong interaction between information and control
  - Often the case where one cannot prove existence of an optimal control law (or design)
- Allow some flexibility over the information pattern
  - What can be said abstractly about the joint selection of information and control patterns?
- No strict preassigned order of action times for the various agents

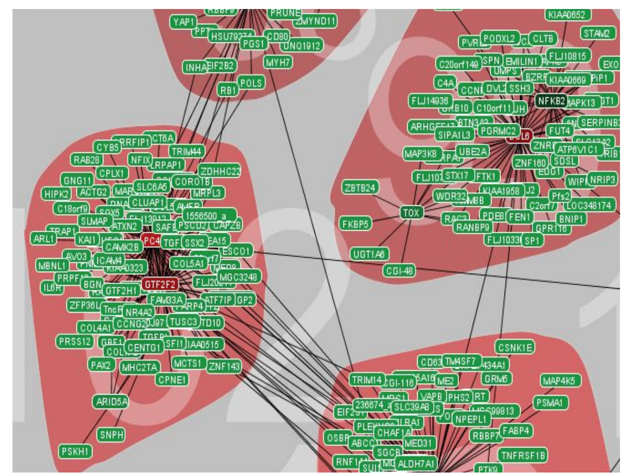
Models with Incompatibility Build-in

- **Active interpretation of operations**: can be thought of as a model for the combined operation of taking a measurement and applying a control law by the agent
- **Passive interpretation of operations**: system's interaction to measurements (used by recent results in information retrieval systems)
- We also get an interpretation of the **conjunction of incompatible events or measurements as "data fusion" or "agreement" between agents**

How Biology Does IT?







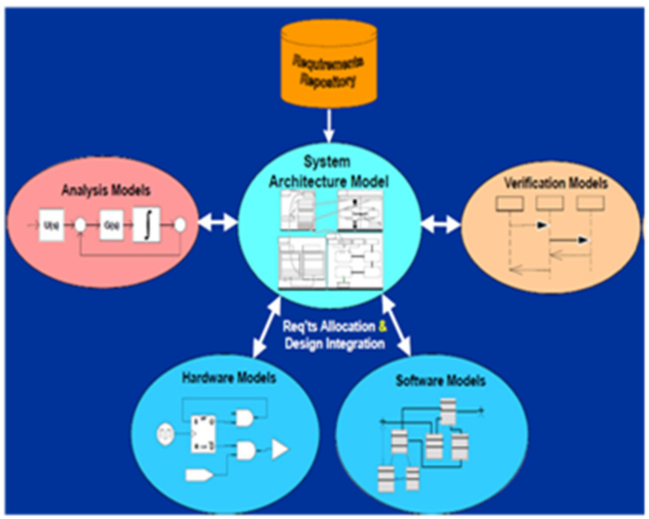
The Institute for Systems Research

Integration of CP and MCO Tradeoff Methods and Tools with SysML– Integrated Models

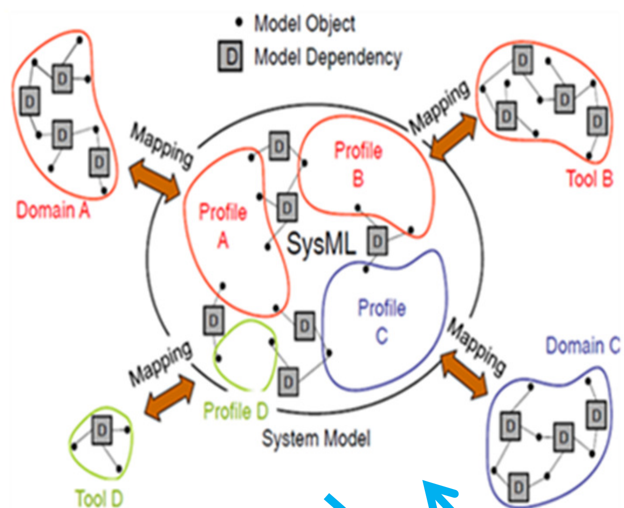
UNIVERSITY OF MARYLAND 18 56

The Challenge & Need:
Develop scalable holistic methods, models and tools for enterprise level system engineering

Multi-domain Model Integration via System Architecture Model (SysML)



System Modeling Transformations



" Master System Model"

Update System Model

Tradeoff parameters

ILOG SOLVER, CPLEX

DB of system components and models

ADD & INTEGRATE

- **Multiple domain modeling tools**
- **Tradeoff Tools (MCO & CP)**
- **Validation / Verification Tools**
- **Databases and Libraries of annotated component models from all disciplines**

BENEFITS

- **Broader Exploration of the design space**
- **Modularity, re-use**
- **Increased flexibility, adaptability, agility**
- **Engineering tools allowing conceptual design, leading to full product models and easy modifications**
- **Automated validation/verification**

APPLICATIONS

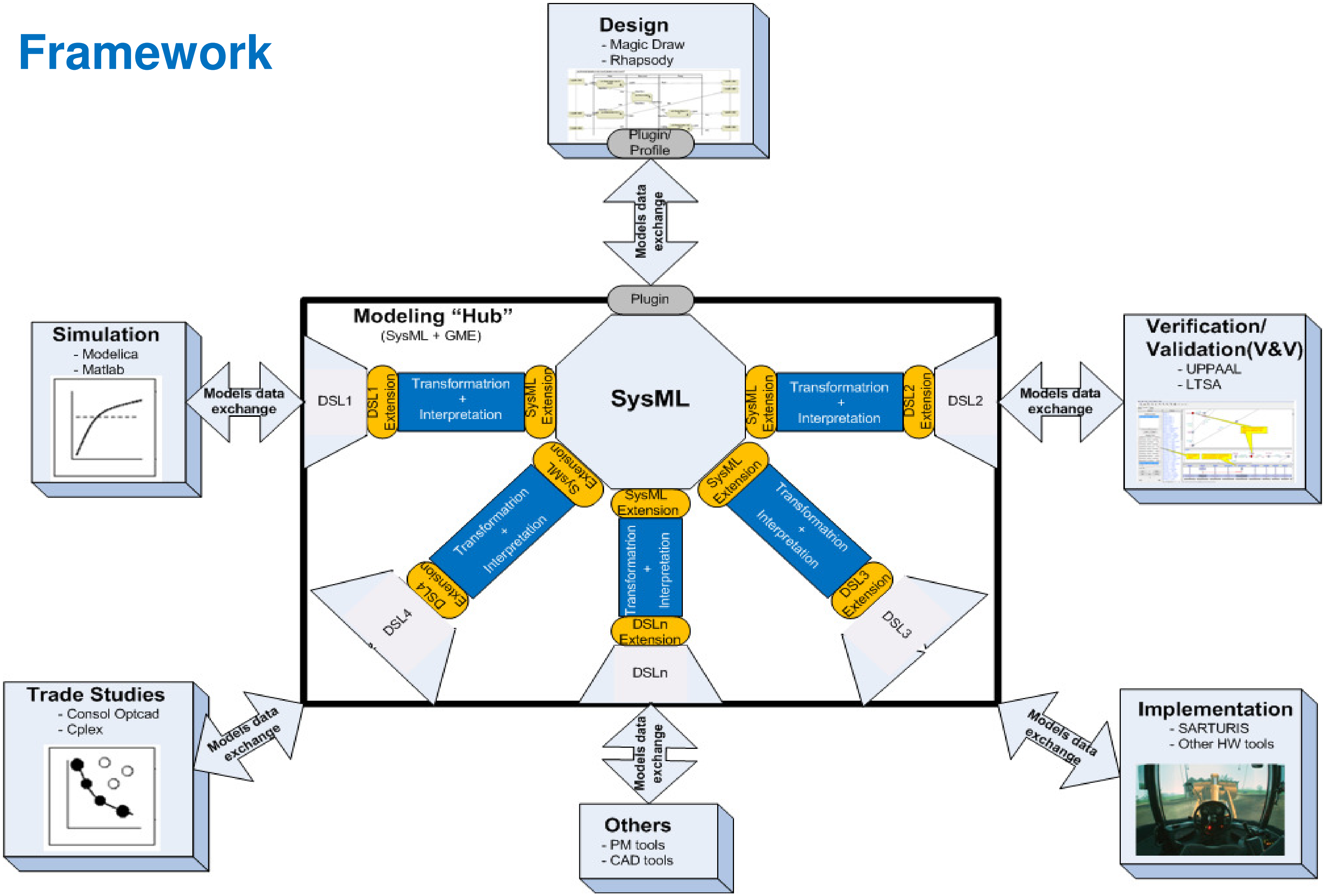
- **CPS broadly**
- **Smart Manufactring**
- **Aircraft /Avionics**
- **Automotive**
- **Energy Effic. Bldgs**
- **Smart Transport.**
- **Smart Grid**
- **MANET and WSN**
- **Collaborating Robots**
- **Security and Trust**

MBSE – Integrated Modeling Hub for CPS

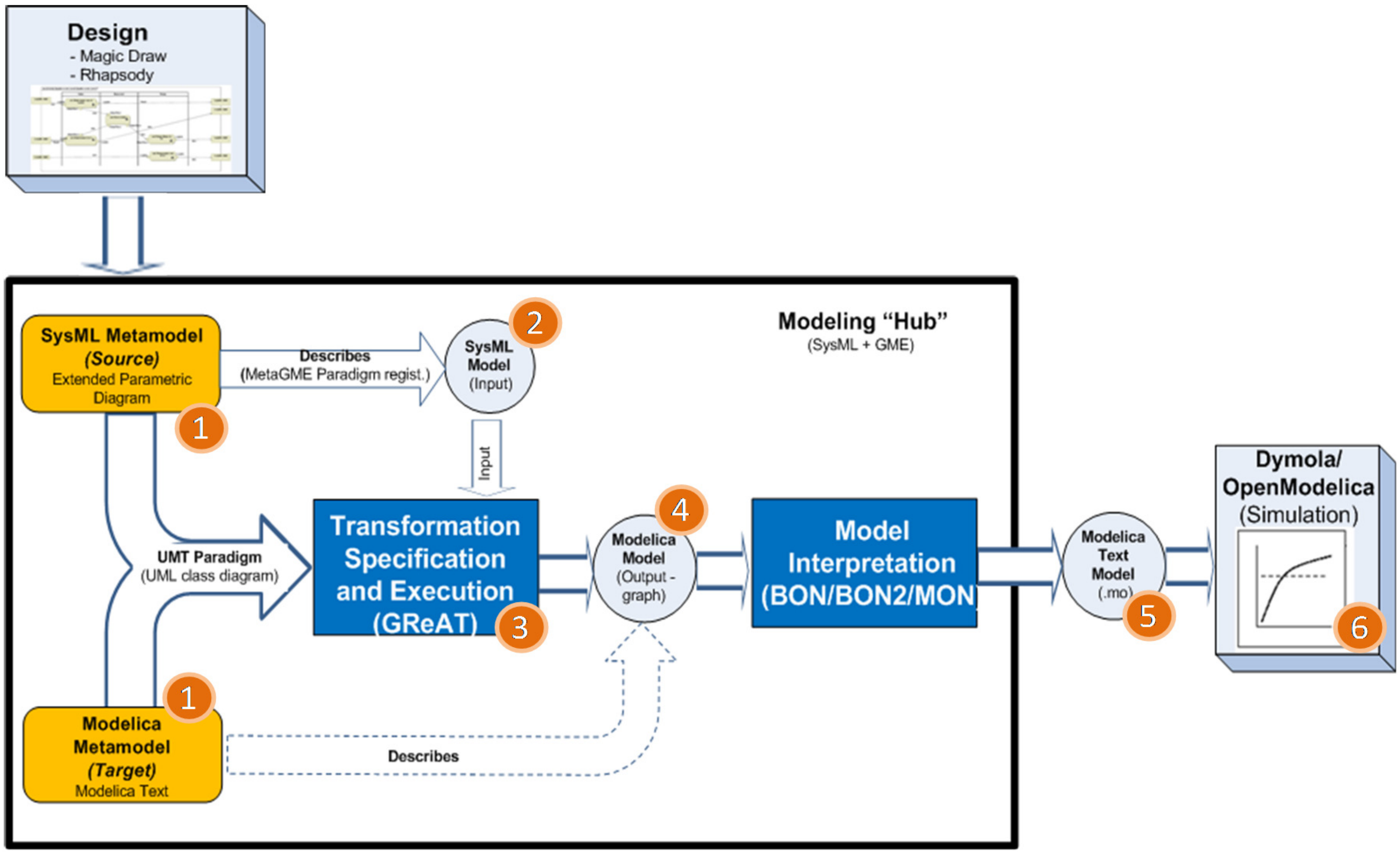
- Integrate systems behavior and structure across domains represented in a ***systems integration modeling hub***.
- Work includes development of ***meta-models*** and ***ontologies*** for the participating domains as well as procedures for synchronizing and transforming data/information across all viewpoints and domains participating in a development
- Support for trade studies and detailed simulations are also needed.
- Key element: find ways for handling discrete changes in systems behavior on the cyber side, coupled to continuous changes in system physics
- Approach is based on ***extensions to regular bond graph technologies*** that will allow the integration of continuous and discrete behaviors

Integrated Modeling Hub (IMH) for CPS

Framework



IMH for CPS



Implementation steps of the CPS IMH

The Institute for Systems Research

IMH : SysML and Modelica Integration

UNIVERSITY OF MARYLAND 1856

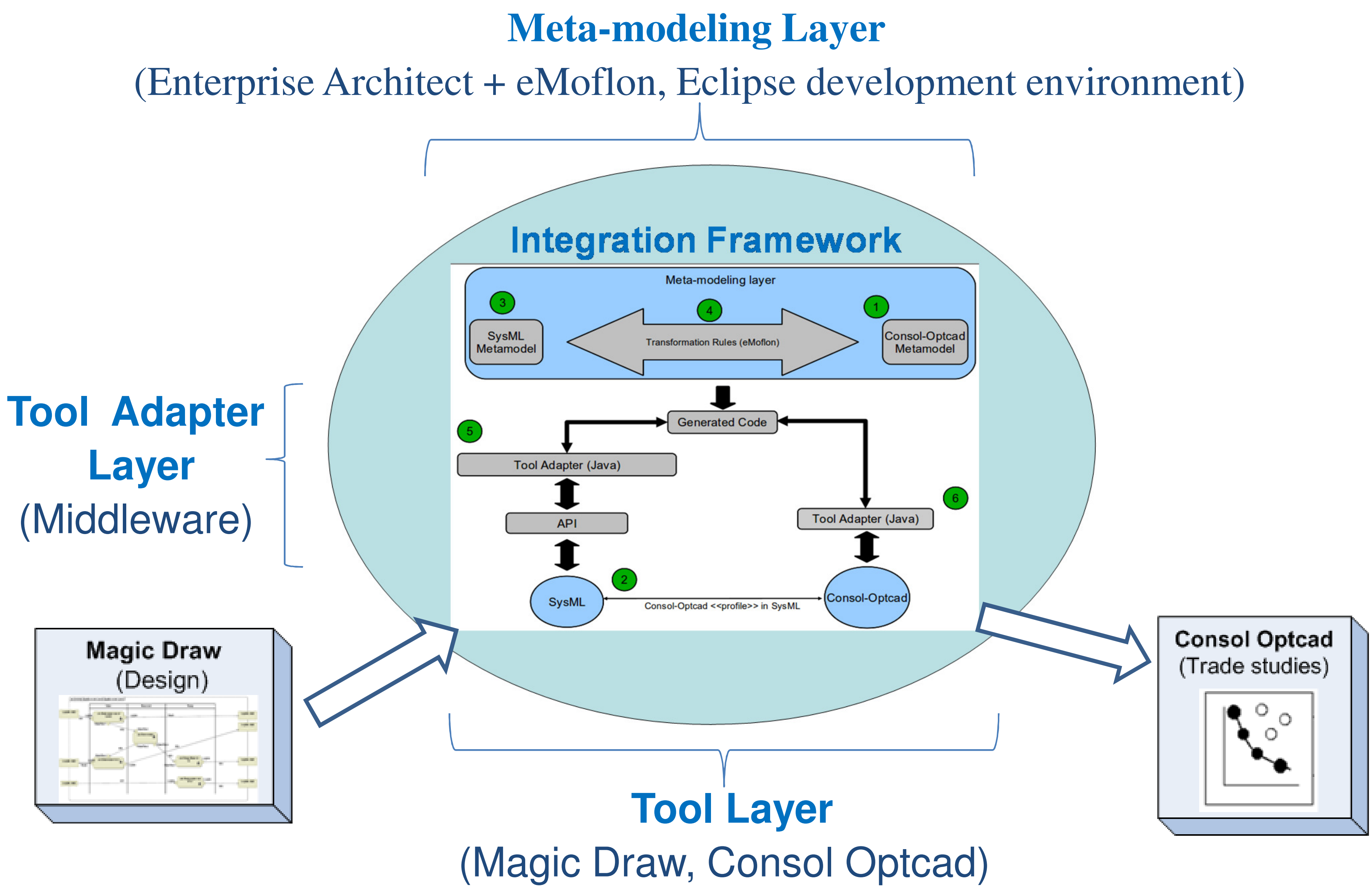
GME toolsuite contributions to the Integration framework

- Infrastructure to **create and add abstract syntax** to concrete syntax used to represent models in DSML and perform semantic mapping among DSML ➔ formulation of mathematical abstractions specifying meaning of models
- Support for the **rapid creation of domain specific modeling**, model analysis and program synthesis environments ➔easy **metamodel** creation and extension specification for integration purposes
- **Transformation framework** (GReAT) built upon the formalism of **graph grammars** (input and output models are considered as graphs); use of Universal Data Model (UDM) framework as underlying data models for programmatic C++ access to transformation artifacts
- Multi-mechanism framework (raw COM, BON, BON2, MON) for creation of **multiple types of components** ➔ code generation of the model artifacts that can be executable in **specified DSL based tool**
- **Weaknesses**: Limited import options for external files (created out of the GME environment); Difficult debugging of errors: multiple/complex interactions among toolsuite components, proprietary constructs/syntax

IMH : SysML and Consol-Optcad Integration

The Institute for Systems Research

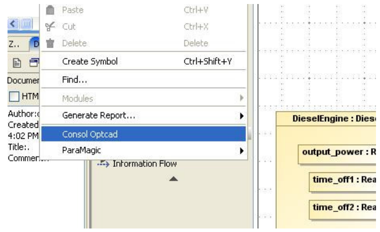
Overview



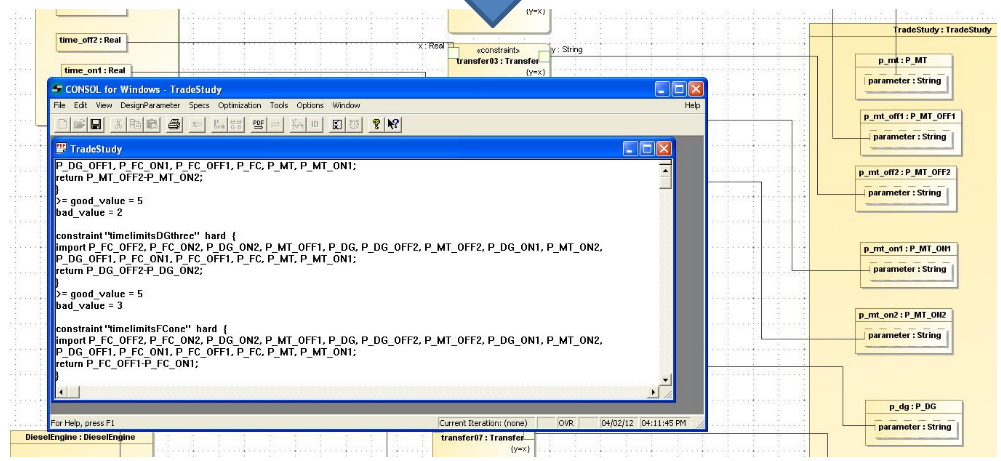
Steps in IMH and Consol-Optcad Integration

The Institute for Systems Research

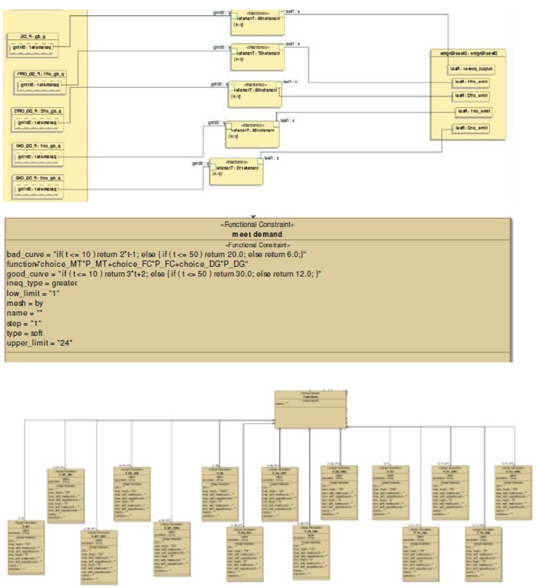
Working Example



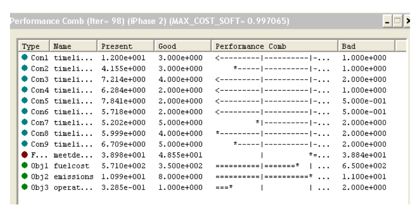
Initiate transformation

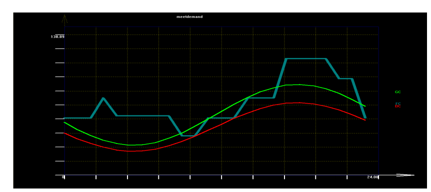


Consol-Optcad environment



Models in SysML





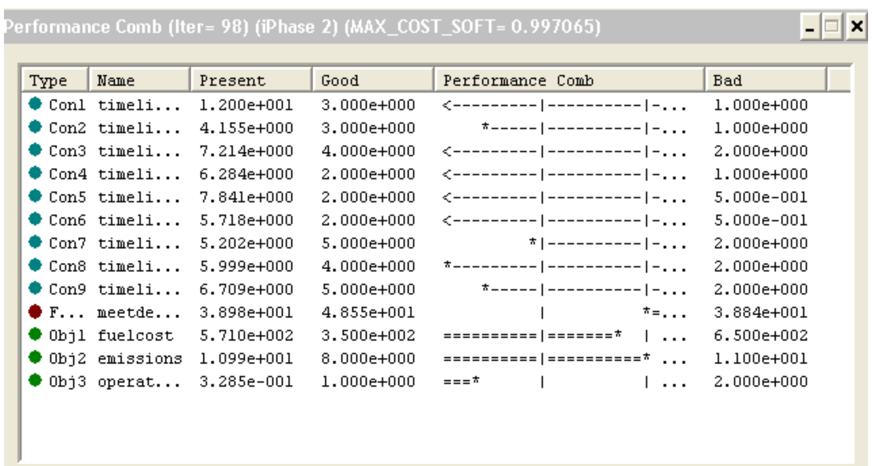
Perform trade-off analysis in Consol-Optcad

The Institute for Systems Research

IMH : SysML and Consol-Optcad Integration

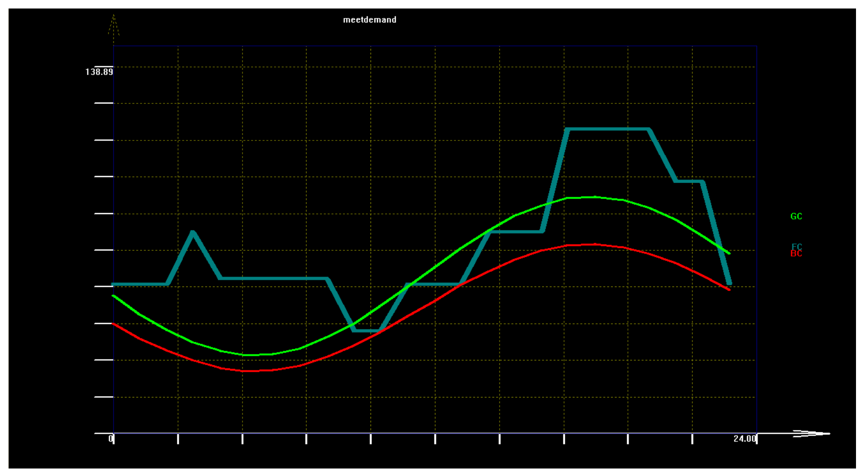
Consol-Optcad

- **Trade-off tool** that performs multi-criteria optimization for continuous variables (FSQP solver) – **Extended to hybrid** (continuous / integer)
- **Functional** as well as non-functional objectives/constraints can be specified
- Designer initially specifies **good** and **bad** values for each objective/constraint based on experience and/or other inputs
- Each objective/constraint value is scaled based on those good/bad values; fact that effectively treats **all objectives/constraints fairly**
- Designer has the flexibility to see results at every iteration (**pcomb**) and allows for **run-time changing** of good/bad values


Performance Comb (Iter= 98) (iPhase 2) (MAX_COST_SOFT= 0.997065)

<table>
<tr><th>Type</th><th>Name</th><th>Present</th><th>Good</th><th>Performance Comb</th><th>Bad</th></tr>
<tr><td>Con1</td><td>timeli...</td><td>1.200e+001</td><td>3.000e+000</td><td><---------|----------|-...</td><td>1.000e+000</td></tr>
<tr><td>Con2</td><td>timeli...</td><td>4.155e+000</td><td>3.000e+000</td><td>*-----|----------|-...</td><td>1.000e+000</td></tr>
<tr><td>Con3</td><td>timeli...</td><td>7.214e+000</td><td>4.000e+000</td><td><---------|----------|-...</td><td>2.000e+000</td></tr>
<tr><td>Con4</td><td>timeli...</td><td>6.284e+000</td><td>2.000e+000</td><td><---------|----------|-...</td><td>1.000e+000</td></tr>
<tr><td>Con5</td><td>timeli...</td><td>7.841e+000</td><td>2.000e+000</td><td><---------|----------|-...</td><td>5.000e-001</td></tr>
<tr><td>Con6</td><td>timeli...</td><td>5.718e+000</td><td>2.000e+000</td><td><---------|----------|-...</td><td>5.000e-001</td></tr>
<tr><td>Con7</td><td>timeli...</td><td>5.202e+000</td><td>5.000e+000</td><td>*|----------|-...</td><td>2.000e+000</td></tr>
<tr><td>Con8</td><td>timeli...</td><td>5.999e+000</td><td>4.000e+000</td><td>*---------|----------|-...</td><td>2.000e+000</td></tr>
<tr><td>Con9</td><td>timeli...</td><td>6.709e+000</td><td>5.000e+000</td><td>*-----|----------|-...</td><td>2.000e+000</td></tr>
<tr><td>F...</td><td>meetde...</td><td>3.898e+001</td><td>4.855e+001</td><td>| *=...</td><td>3.884e+001</td></tr>
<tr><td>Obj1</td><td>fuelcost</td><td>5.710e+002</td><td>3.500e+002</td><td>==========|=======* | ...</td><td>6.500e+002</td></tr>
<tr><td>Obj2</td><td>emissions</td><td>1.099e+001</td><td>8.000e+000</td><td>==========|==========* ...</td><td>1.100e+001</td></tr>
<tr><td>Obj3</td><td>operat...</td><td>3.285e-001</td><td>1.000e+000</td><td>===* | | ...</td><td>2.000e+000</td></tr>
</table>

Pcomb



Example of a functional constraint

System Complexity Analysis and Control

- Basic challenge is to solve a problem described over a network of components where decisions in one component may affect the choices available in another component and there is a global objective that can only be understood by examining the complete space of decisions
- Curse of dimensionality – Use structural decomposition techniques of systems engineering
- It turns out that system complexity is exponential in *treewidth* and *linear* in problem size. The intuition behind this result is that problems on graphs are difficult to solve due to the presence of loops. Removing the loops by multiplexing variables (aggregating them into objects) can lead to the tree decompositions of graph problems. Once the problem is in the form of a tree, then summary propagation is a viable technique for solving the problems.
- Have successfully applied to sensor networks and quadrorotors

Wedging Systems

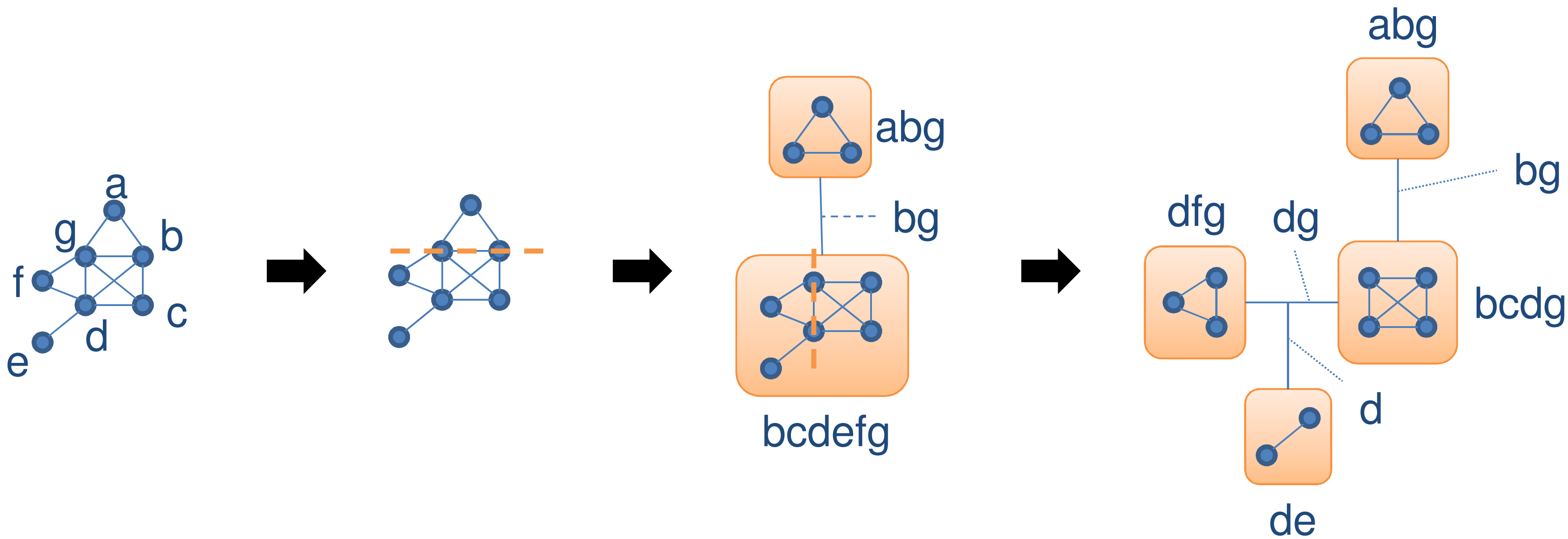
- System represented by an undirected graph *G* = <*V*,*E*>.
  - Nodes, *V*, correspond to variables.
  - A formula $f(x_1, \ldots, x_n) = C$ induces edges (x_i, x_j) $\forall i \neq j \in [1, n]$.
  - Edge, $(x, y) \in E$, means that variables x, y are in mathematical relation.

- Rules of system partitioning.

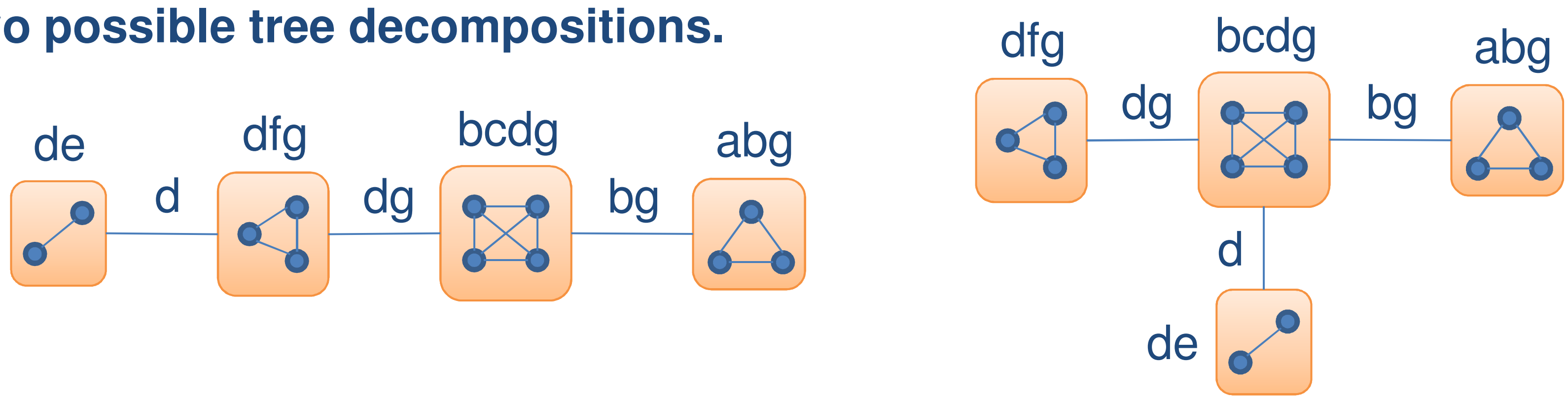
1. Choose a subset of nodes that ***completely separate*** the graph into subgraphs.
2. Separation produces an ***interface relation*** that contains all the nodes in the separator.
   - By adding links, brings resulting subsystems closer to *inseparability*.

- Due to recursive partitioning this **decomposition results in *trees***.

Tree Decomposition by Creating Interfaces (Chordal Example)



Two possible tree decompositions.



Tradeoff Analysis using Summary Propagation

Solution consists of a partially ordered set of local computations

Sample the parameter space of the Perch Block and determine which points are feasible.

Propagate shared variables (drop PerchTime by projection).

Evaluate the Weight Block by sampling its parameter space and taking intersections additionally with the propagated data.

Sample the parameter space of the Range Block and determine which points are feasible.

Propagate the shared variables (drop FlightCurrent by projection).

Evaluate the Metrics Block by sampling its parameter space and taking intersections additionally with propagated data.

Propagate the shared variables (drop Weight by projection).

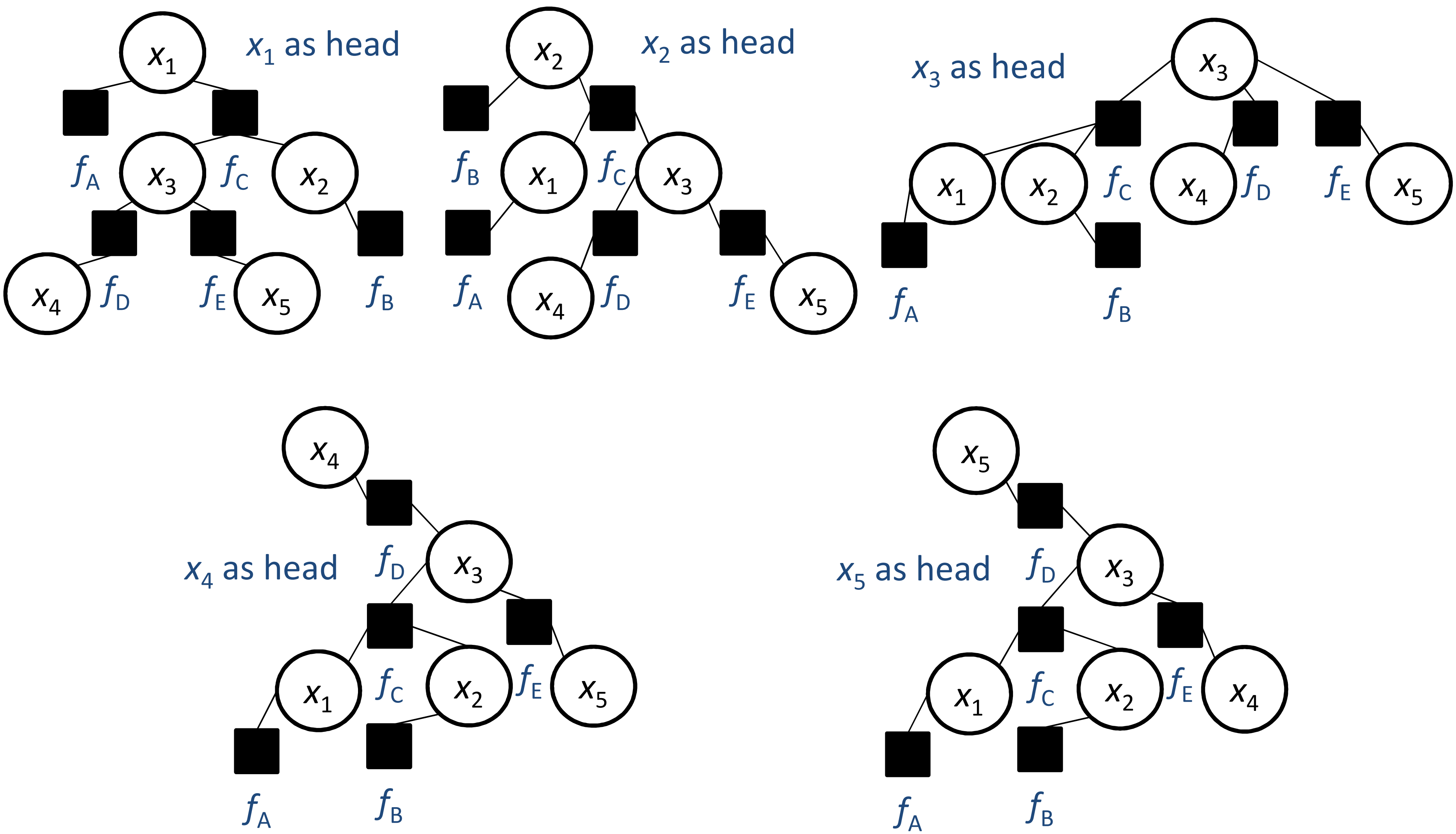
- Builds tables of feasible values for each of blocks
- Uses (weighted) ***natural-semijoin*** on tables to propagate information
- Applies (aggregated) ***projection*** on tables to hide unnecessary information

System Complexity Analysis and Control

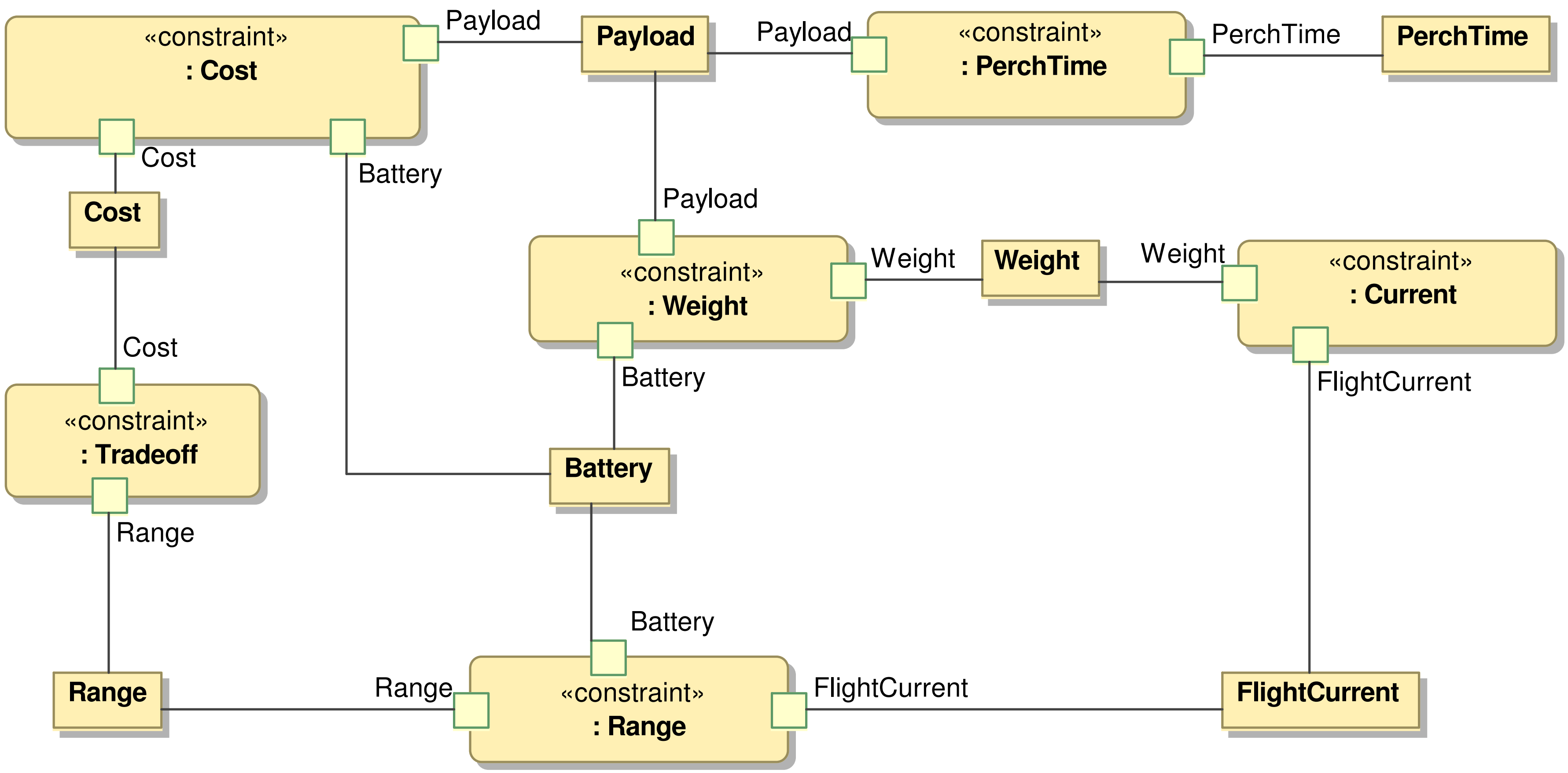
Tradeoff Queries

- The query itself influences the shape of the resulting graph
- A query that is not local can create links between non-local variables
- The resulting graph and *analysis complexity is dependent on the query*

Query Induced Hierarchies



Quadrotor Parametric Diagram

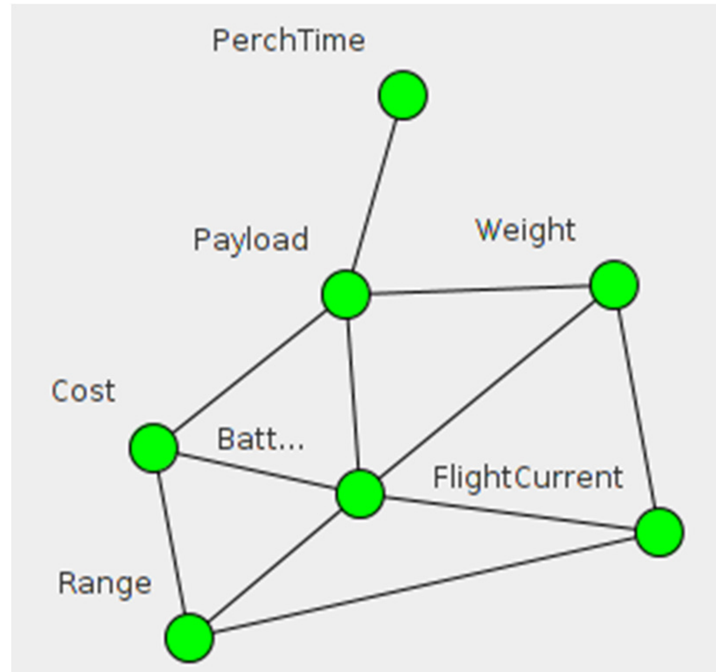


Quadrotor Analysis

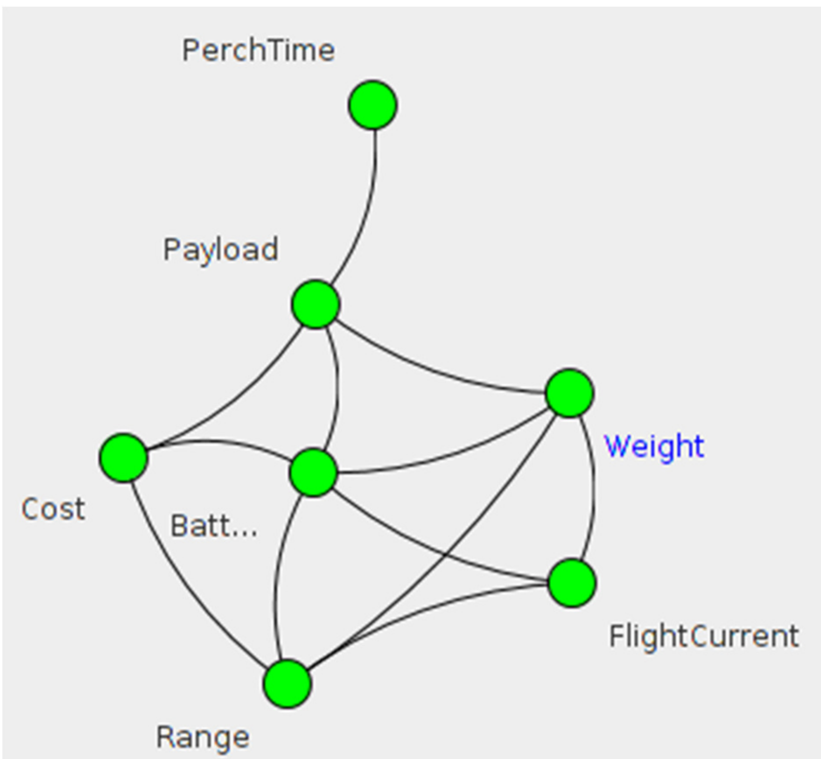
The

NO	Func Name	
1	Cost	Cost,Battery,Payload
2	Tradeoff	Cost,Range
3	Range	Battery,Range,FlightCurrent
4	Weight	Weight,Battery,Payload
5	PerchTime	Payload,PerchTime
6	Current	FlightCurrent,Weight

Tool input from parametric diagram.

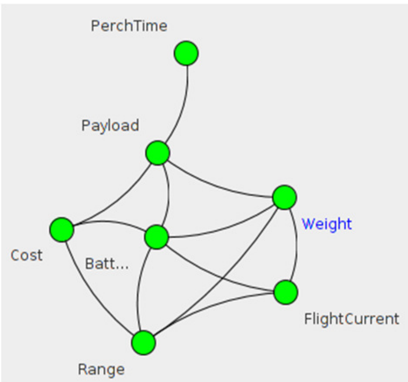


Initial graph.

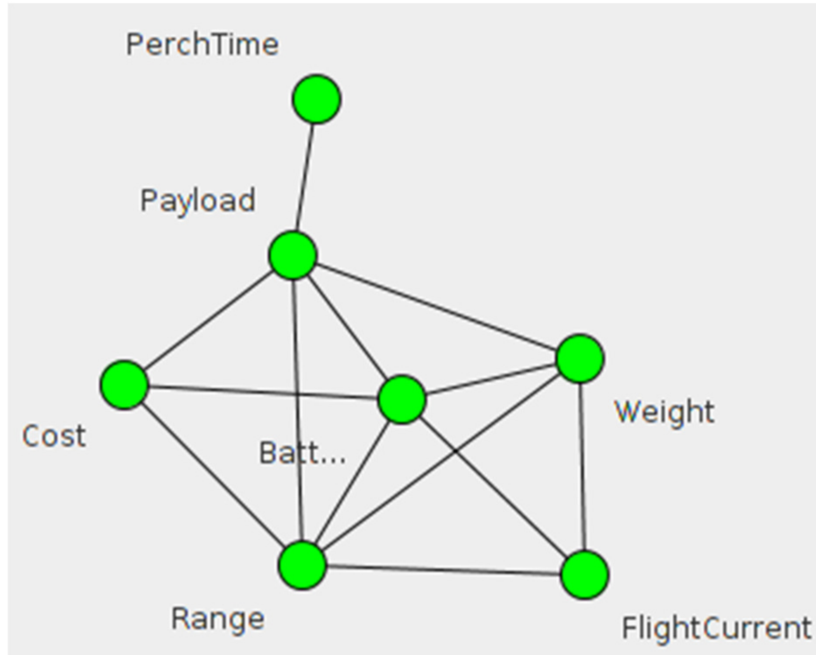


Weight to range fillin created.

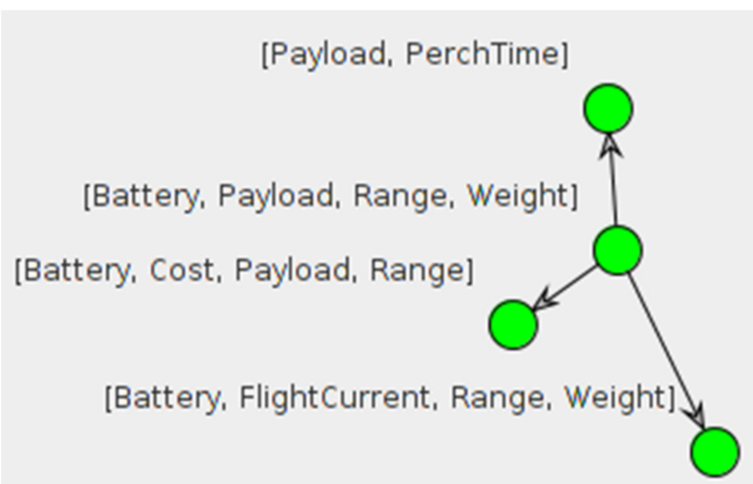
Quadrotor Analysis (cont.)



Weight to range fillin created.



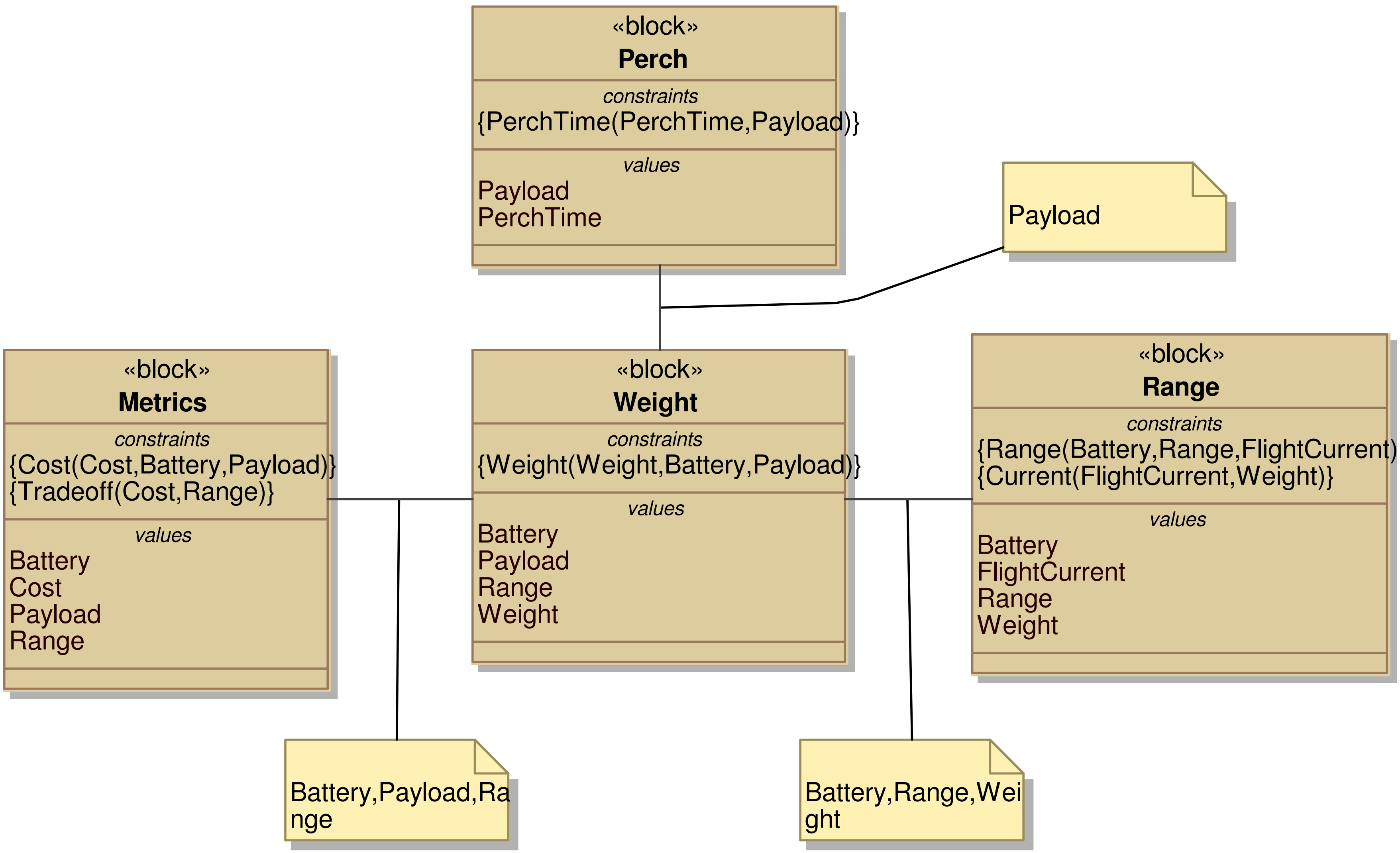
Payload to range fillin created. Graph is now chordal.



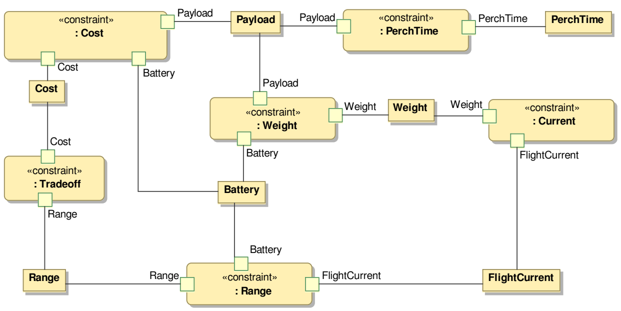
Join tree created.

The tool implemented currently uses **elimination order** rather than **separators** to perform analysis. They are mathematically equivalent. An implementation using separators is underway.

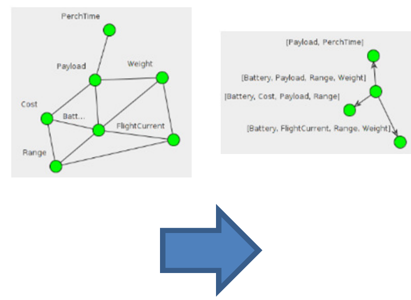
Quadrotor Factor Join Tree

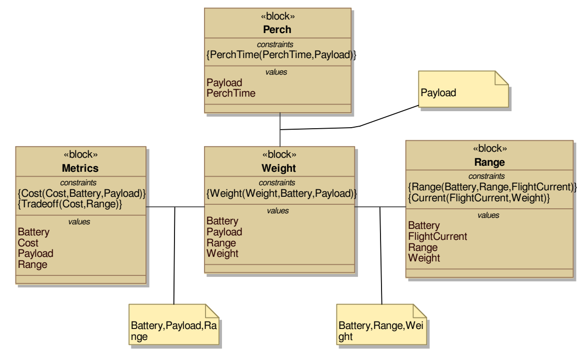


Tool for Structural Analysis



System Parametric Diagram





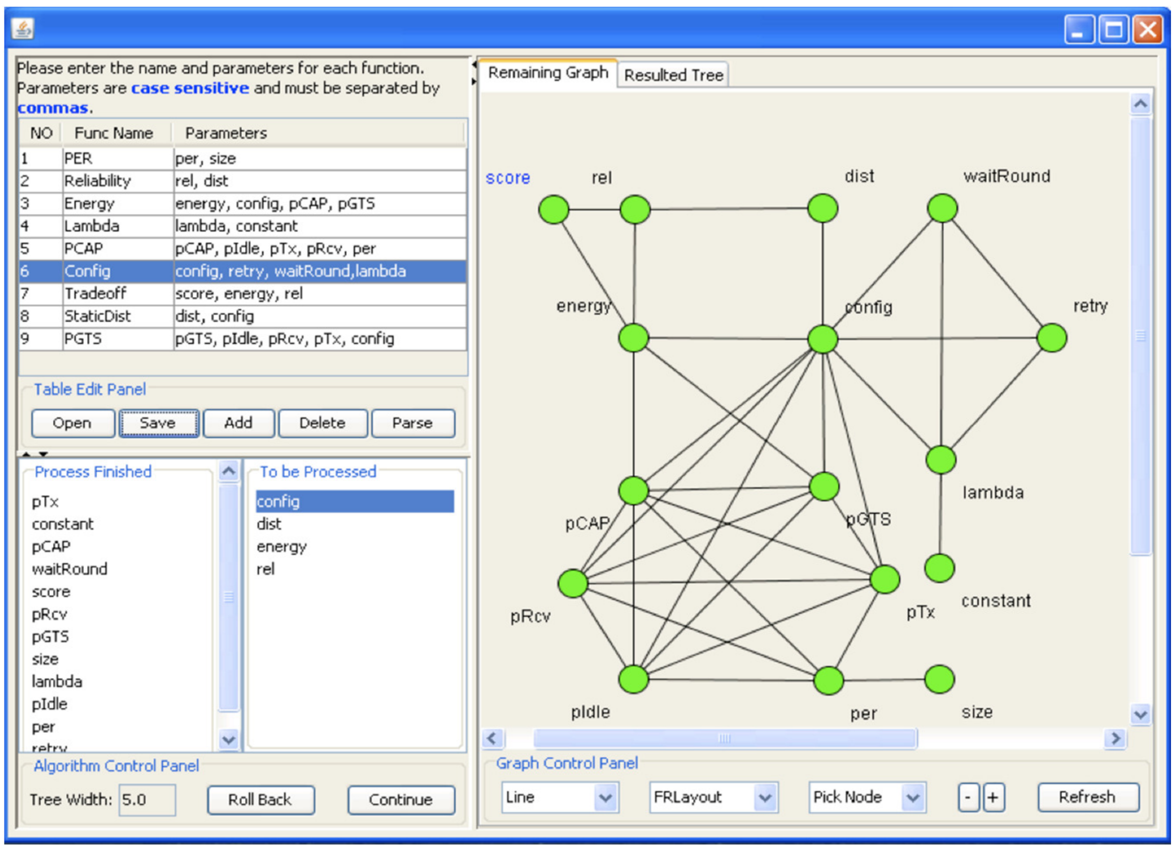
System Block Diagram

Interactive transformation tool

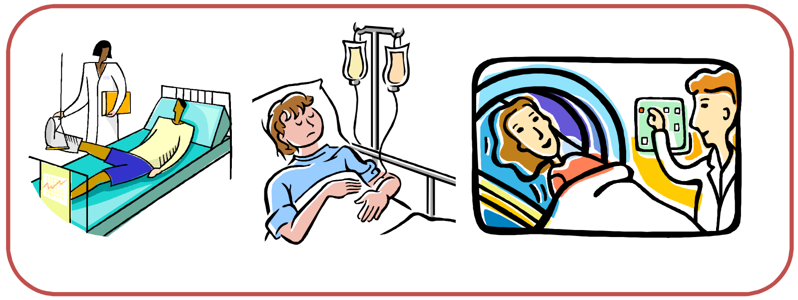
- Implemented a Java based interactive tool to assist with the transformation of a relational system description (as expressed using parametric diagrams).
- Tool uses a simple random tree search algorithm to elaborate user selections towards optimal treewidth configurations.

Tool for Wireless Sensor Nets

- We considered the trade-off analysis between energy efficiency and transmission reliability in wireless sensor networks, where the IEEE 802.15.4 standard is applied as the media access control protocol. For simplicity, we only provide high-level abstract functions here, emphasizing the abstract relationships between the parameters in each function.



Model-Based Systems Engineering for ITU Management



Healthcare operations

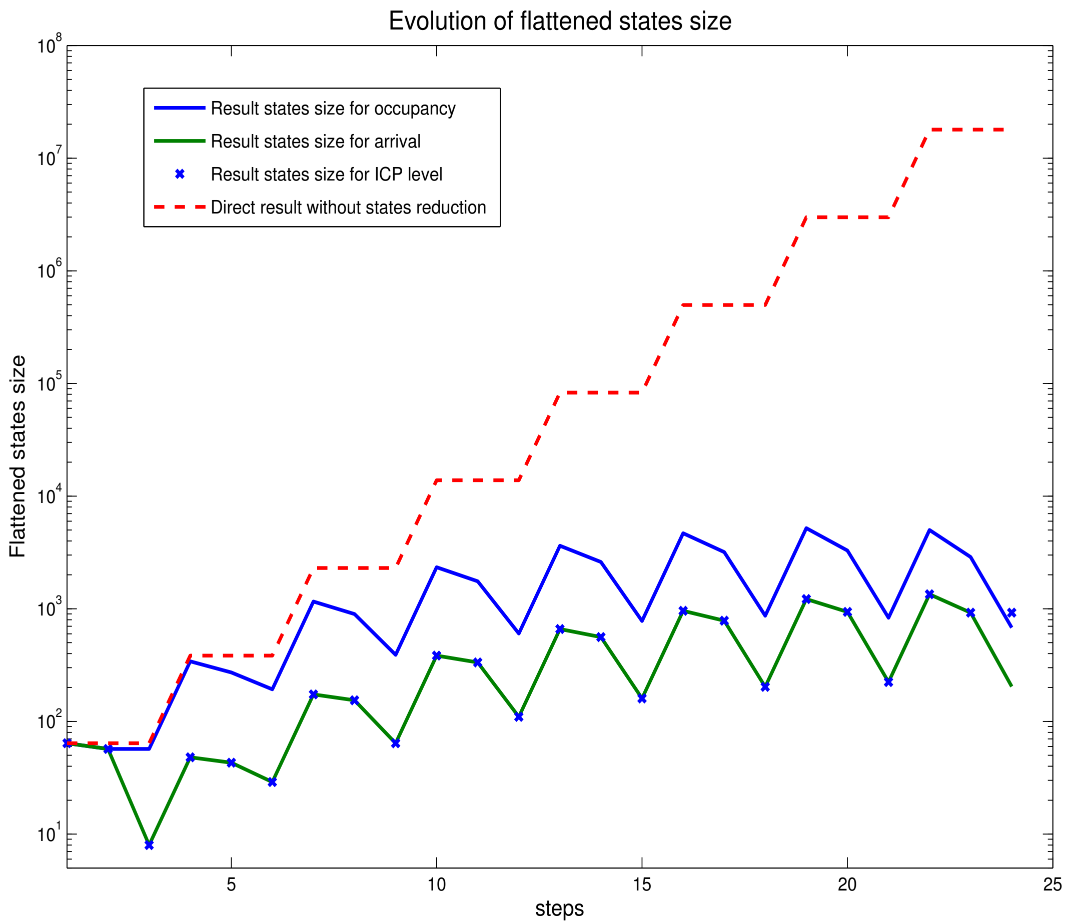
Monitor performance,
generate ideas,
implement changes

Build models,
analyze operations,
predict changes

ICU is a Cyber Physical System

- Cyber physical systems are defined as those with *tight coupling between the computational and physical aspects*.
- Allowing here that physical is not restricted in definition to systems that are literally physical, but anything having to do with some real world non-software system.
- We propose that deeper integration between the sensors, instruments and the enabled computation in the practice of medicine will allow for better care for patients at reduced cost.
- In healthcare, the "physics" are dominated by the uncertainty in patient outcomes. This means that the integration implies computation over time evolving uncertainty.
- We propose using dynamic Bayesian networks (DBNs) as a model for capturing this uncertainty.

State Reduction Achieved



Number of states as a fcn of number of steps in inference
Sawtooth pattern is the result of the project-compose pattern

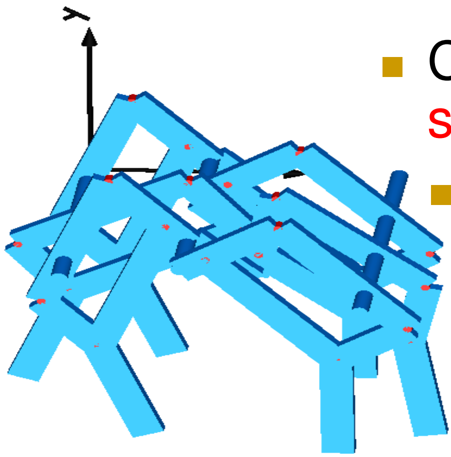
The Institute for Systems Research

Smart Robotics

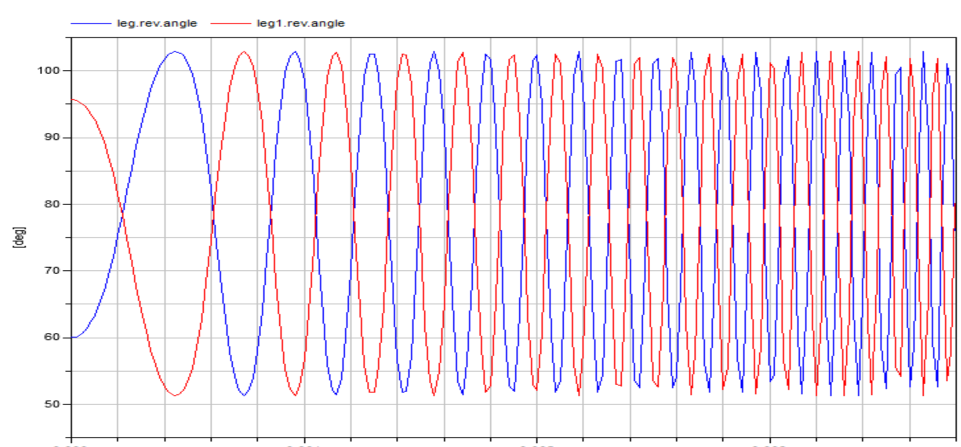
MBSE for Robotic Arms and Grippers

- Transcend areas of application: ***from space to micro robotics***
- Include ***material selection*** in design
- Include energy sources, resilience, reliability, cost
- Include validation-verification and testing
- Use integrated SysML and Modelica modeling environment
- Link it to tradeoff tools Comsol-Optcad, CPLEX - ILOG Solver
- Demonstrate reuse, traceability, change impact and management

Smart Micro-Robots

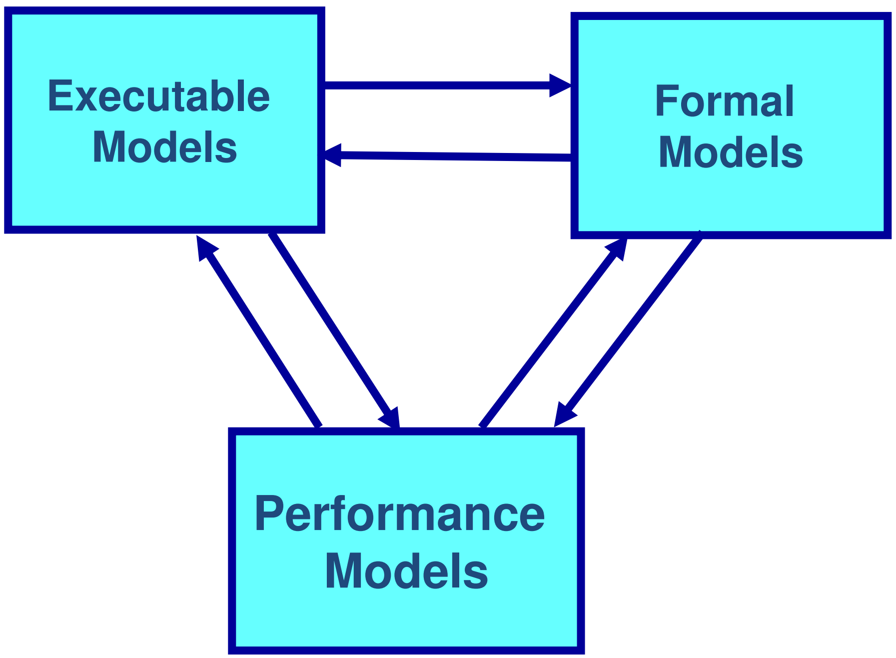


- Current prototype-based design for micro-robots is not systematic – design “on the bench”
  - Control algorithms for micro-robots require a more precise description of physical layers inside the loop. Separate development of control and physical layers is not possible in micro-robotics because of complex material constrains.
- Material level properties, which were not well-explored, are essential in micro-robotic design.
- Physical system modeling languages, such as Modelica, give a way to solve these problems when integrated with top level design languages such as SysML.



Universally Composable Security for CPS

- **Security, trust, safety is a cross-cutting challenge for all CPS and in particular for networked CPS.**



Studying compositionality is necessary!

Universally Composable Security of Network Protocols:

- Network with many agents running autonomously.
- Agents execute in mostly asynchronous manner, concurrenty several protocols many times. Protocols may or may have not been jointly designed, may or not be all secure or secure to same degree.

Key question addressed :

- **Under what conditions can the composition of these protocols be provably secure?**
- **Time and resources required for achieving this?**

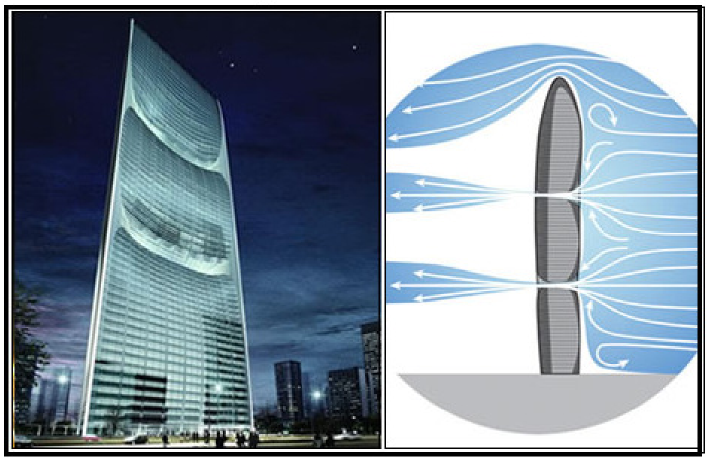
Physical Layer Authentication to aide UCS

- To date, security mechanisms in autonomous networks, including networked CPS, have largely neglected the physical layer, but the establishment of preconditions on the physical layer simplifies secure protocol composition for concurrent joint execution by many agents
- The network can authenticate physical devices based on immutable "device fingerprints" – properties like defects in the waveform or spectrum of a particular RF emitter, or the signature of faults in a chip.
- **The novel result is that a small percentage of nodes and links "hardened" by PHY layer security can facilitate UCS!**

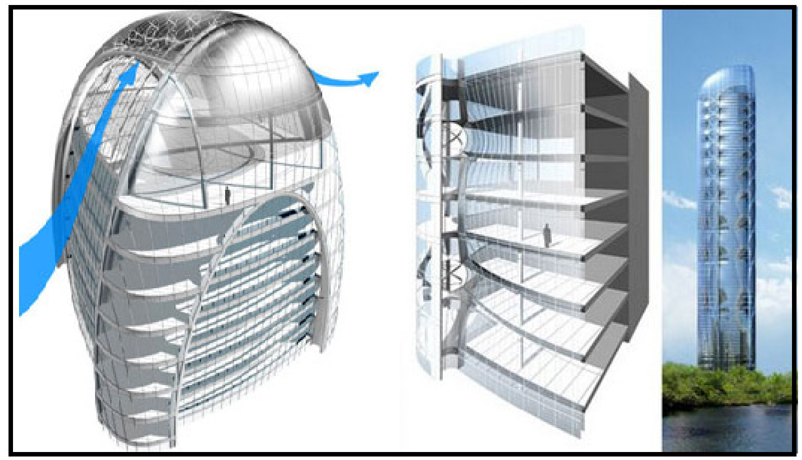
Cyber-Physical Building Systems

- **Research focus:** Platform-Based Design for Building-Integrated Energy Systems.

Pearl River Tower Complex

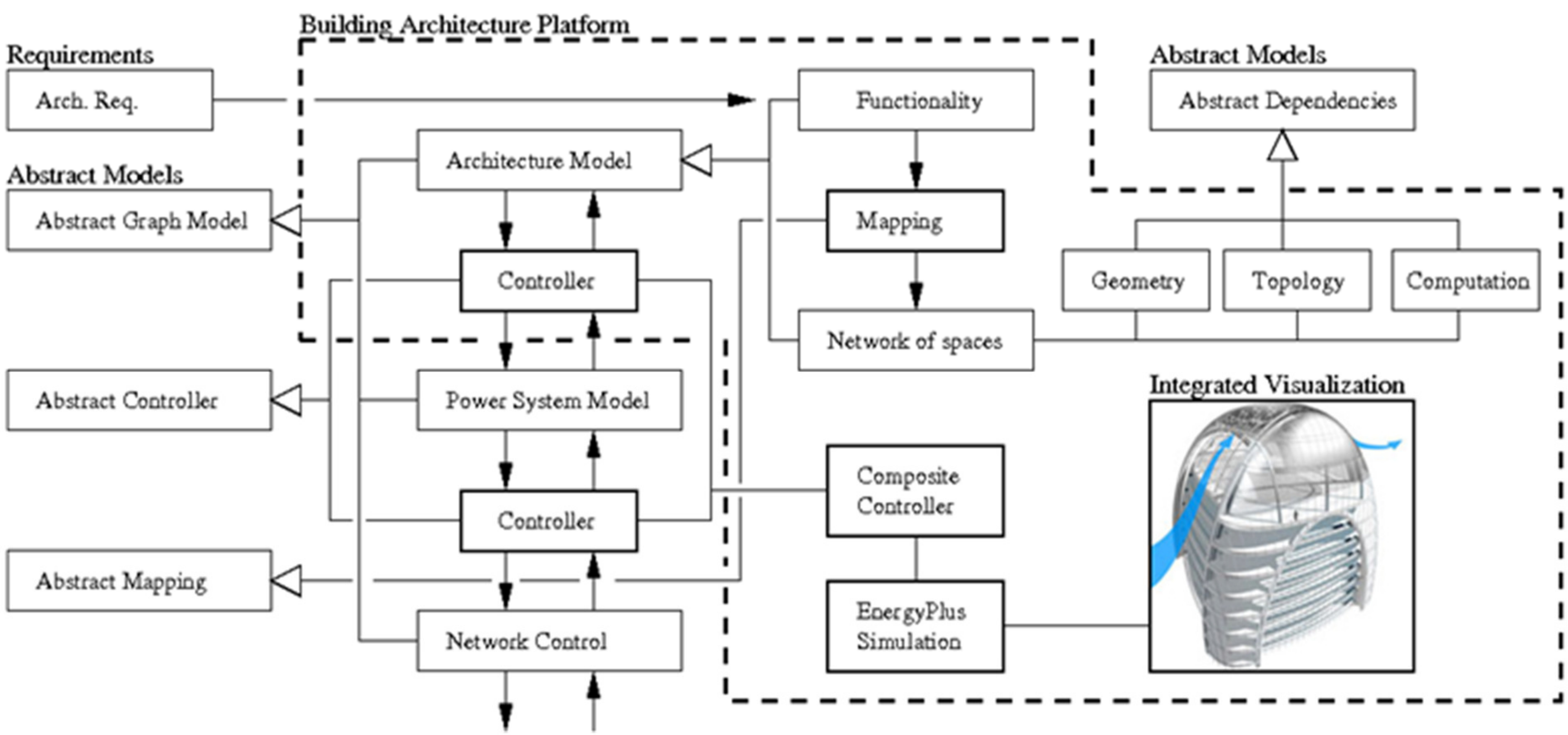


Green Technology Tower –– Architectural Proposal for Chicago



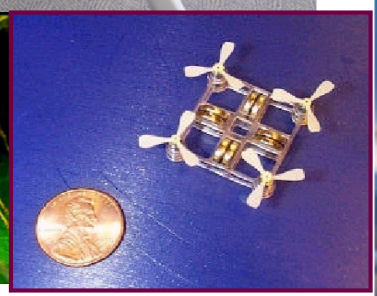
DESIGN PLATFORMS FOR SE BUILDING-INTEGRATED ENERGY SYSTEMS

Extensible framework for assembly of (model, controller, simulation, viewpoint) process networks and communication for platform-based design of building-integrated energy systems



AUTONOMOUS SWARMS – NETWORKED CONTROL

- *Component-based Architectures*
- *Communication vs Performance Tradeoffs*
- *Distributed asynchronous*
- *Fundamental limits*



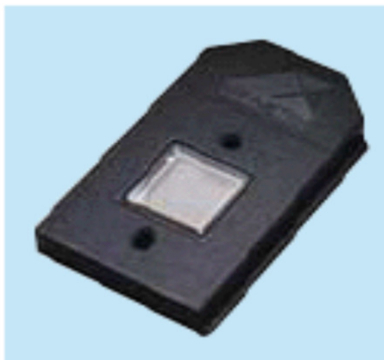
Revolutionizing Drug Manufacturing

- Rapidly approaching untenable situation in human health -- Blockbuster drugs, which cure major diseases afflicting huge populations, are being pulled from the shelves (e.g., Vioxx) for unforeseen side-effects. They are being replaced by drugs that have smaller market potential and more localized impact (subpopulations, e.g., FluMist).
- The current cost of developing a drug and getting it to market exceeds $1B and the process takes over ten years.
- These competing forces cannot be resolved without truly transformational changes in the way drugs are discovered, developed, and approved.
- This need is exacerbated by the emergence of personalized medicine – a natural outcome of high throughput sequencing technologies.

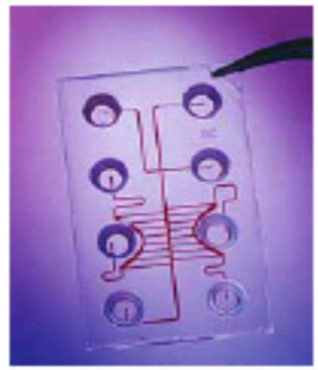
Hybrid Loc -- Biochips

Biochips are currently emerging with different form factors and technologies for applications in research, pharma and healthcare

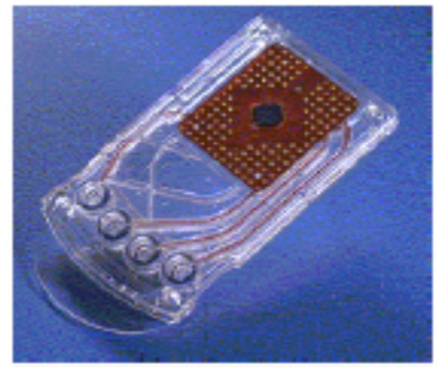
All biochip concepts are disposables



DNA µArray



µfluidic chip



µfluidic chip
+
DNA µArray

Applications:

- Basic research
- Pharma R&D / Drug development
- Healthcare

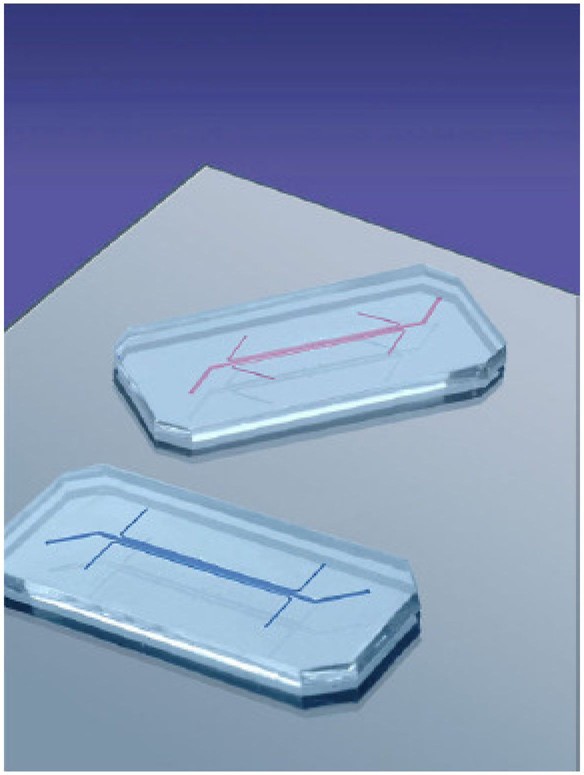
} "Red Biotech"

- Agriculture and environment — "Green Biotech"
- Industrial and process control — "Grey Biotech"

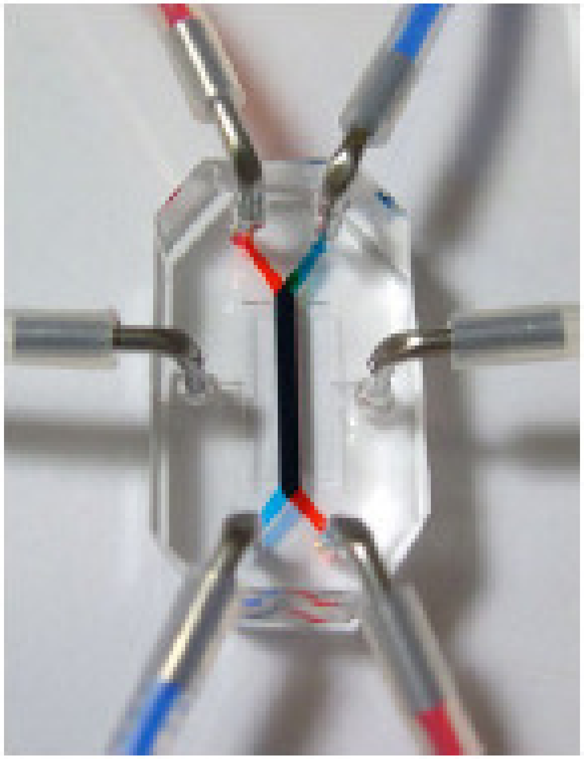
2

Organ-on-a Chip -- Biochips

Wyss-Lung on a chip -- 2010



Wyss-Gut on a chip -- 2012



Compositional Synthesis of Heterogeneous LOC for testing drug side-effects

Challenges

- Link molecular diagnostics to genetic profiles to drugs and their side effects
- Need to have several components integrated: silicon-hardware, software-cyber, living cells, microfluidics
- Currently no general design tools allowing users to design a LOC for a specific application
- Design from modular, reusable components
- Challenges everything we know about hybrid modeling, computation (even its foundation), communications (including molecular), processing AND medicine and “BIG” Pharma

Thank you!

baras@umd.edu

301-405-6606

http://www.isr.umd.edu/~baras

Questions?