Virtual Integration Platform: Systems – Network "Wind Tunnel"

Gabor Karsai and Janos Sztipanovits
Vanderbilt University

FORCES Kickoff Meeting
April 12, 2013


**History**


- C2 Wind Tunnel Applications/Extensions
  - AFRL project on resilient C2 architectures (cyber attacks impact on mission performance), (2010-2011)
  - Cyber C2 – Course of Action evaluation, AFRL, (2010-2012)
  - TRUST SCADA simulation platform, NSF, (2010-2011)
  - C2 Wind Tunnel/DETER integration, ARO, (2009-2012)
  - Experiments with smart grid simulation (with WSU) (2011)
  - Other users – USA, S. Korea, EU
Integrated control, communication, power system
Interfacing Existing Modeling Tools

Power system model (PSS/E, MATLAB)
Transmission
Distribution
Local Control

Sensors
Actuators

Communication

Communication Modeling (NS-2, OPNET…)

State Estimation
Stability Monitoring and Control

Power System

Integration
Electric Grid Modeling and Simulation: Gaps

- Modeling interdependencies of infrastructure accompanying the power grid, including sensors, control, communication network and computational components is challenge.
- No single tool to model power system, communication and control system
- No single tool to model the system at transmission and distribution level with attention to all details
- Tools needed to combine data that is multi-rate, multi-scale, multi-data, multi-use, multi-model from multiple heterogeneous interacting domains
- Solutions:
  - Single integrated tools
  - Compose multiple models and simulations: model and simulation integration challenge
5-Bus Example: Power Grid Model

Tool: SimPower/MATLAB
Modeling Language: Simulink
Semantics: Continuous Time

Other Tools: RTDS
Modeling Language: RSCAD
Semantics: Continuous RT
5-Bus Example: Communication Model

**Tool:** NS-2

**Modeling Language:** NS-2

**Semantics:** Discrete Event

**Other Tools:** OMNeT++

OPNET, TrueTime,..

**Semantics:** Discrete Event
5-Bus Example: Control Center Model

**Tool:** MATLAB

**Modeling Language:** Simulink

**Semantics:** Discrete time

**Other Tools:** DEVS, LabView,

**Semantics:** Discrete Event
How to integrate the simulators?
How to integrate the models?
How to execute coordinated experiments?

Simulators have different timing models
Execution needs to be coordinated
Data needs to be shared
Different time-scale and resolution
Logical time v.s. real time
Different simulation engines

Modeling languages are different
Semantics is different:
- continuous time
- discrete time
- discrete event
Simulated systems are interacting but modeling languages do not have construct to express them
No support for specifying experiments
Technology Needs

**Integrating models**
- Heterogeneous modeling for different domains: power systems, communication networks, control software, command and control organization, scenarios, etc.
- Needed: an overarching integration model that connects and relates the heterogeneous domain models in a logically coherent framework.

**Integrating the system**
- Heterogeneous simulators and emulators for different domains: Colored Petri Nets, OMNET++, NS-2, OPNET, DEVS, Simulink/Stateflow, DSMLs, etc.
- Needed: an underlying software infrastructure that connects and relates the heterogeneous simulators in a logically and temporally coherent framework.

*Key idea: Integration is about interactions across system components. Why don’t we model the interactions and use these models to facilitate model and system integration?*
Simulation Integration Frameworks:

- **SIMNET** (DARPA, live training, 80’s)
  - incremental updates (dead reckoning)
  - objects and events to communicate changes
  - human-in-the-loop real-time simulation

- **DIS** (DARPA, large-scale battlefield sims, 90’s)
  - time is assumed to be real-time
  - interaction based on UDP broadcast
  - 1278.1a-1998 IEEE standard

- **HLA** (DMSO, mid 90’s)
High Level Architecture (HLA)

• An IEEE standard for “interoperable” and “reusable” models and simulations.
  - Most used specification (also used in the demo) is IEEE HLA 1.3 (1998)
  - Most recent specification is IEEE HLA 1516 (2000+)

• Provides a general purpose infrastructure for “distributed” simulation and analysis
  - Interactions are defined via Federation Object Model, Simulation Object Model and Management Object Model
  - Pub/Sub interactions over TCP
  - Complex time management/coordination

• Run Time Infrastructure (RTI)
  - Open source (RTI Portico) and COTS versions
  - Several middleware variants supporting better scaling:
    - Test and Training Enabling Architecture (TENA)
    - TIBCO’s TIB, OMG’s DDS, IBM’s DCS, MS.NET, Cornell’s Quicksilver
Multi-modeling With Existing C2WT

Three levels of modeling & customization:

• **Modeling the modeling tools - Infrastructure**
  - Performed “one-time” when a new simulation platform (model type) is integrated [usually by developers]
  - Metamodeling and metamodel comp.

• **Component Modeling and Integration**

  **Modeling – Scenarios**
  - Performed when a new scenario has to be experimented [usually by subject matter experts who can describe scenarios]
  - Requires:
    - Models of model tools used (‘federates’)
    - Data models
    - Integration models: Interactions among the models

• **Modeling Situations – Experiments**
  - Execution platform, experiment setup, deployment [usually by experimenters who can parameterize experiments and analyze experimental results]

---

Simulink/Stateflow; CPN, C++/Java; DevsJava; OMNeT++; NS-2; Delta-3D; GoogleEarth
HLA-RTI Portico
Experiment design

Physical system libraries
Operator utilities
Network models
Network attack models
Scenario models
3D environment models
Data collection in MySQL

Experiment models
“Excursions”
Data analysis scripts
Deployment models

http://wiki.isis.vanderbilt.edu/OpenC2WT
Experimentation: Sequence model controls execution

Orchestration Engine executes the Sequence Model

Simulation engines co-operate via the C2WT
Conclusion

• We need highly scalable modeling and simulation infrastructure that is model-based, cyber-physical, scalable and highly automated.

• Feasibility of the proposed approach largely depends on semantically sound modeling and model integration, heterogeneous multi-model simulation and information management infrastructure.