Experimental Research

Edited and Presented by
Alberto Sangiovanni Vincentelli,
Co-PI
UC Berkeley

Chess Review
November 18, 2004
Berkeley, CA

Overview

- Experimental research is an essential component of CHESS
  - Feedback on approach
  - Inspiration for new theory
  - Impact
- Wide range
  - Industrial cases (to be covered in the afternoon)
    - Automotive (safety-critical distributed systems)
    - System-on-Chip (high-complexity platforms)
  - Internal experimental test benches
    - Wireless Sensor Networks (security, low power)
    - UAVs (complex control, sensor integration)
  - New domains:
    - Scientific Workflows: Kepler Stochastic
    - Hybrid Systems in Systems Biology

Overarching Criteria

- An application should exercise
  - Theory: hybrid models, Models of Computation, control algorithms
  - Tools and Environments
  - Path to implementation
- An application should be relevant for industry or for government agencies

Some Applications Addressed

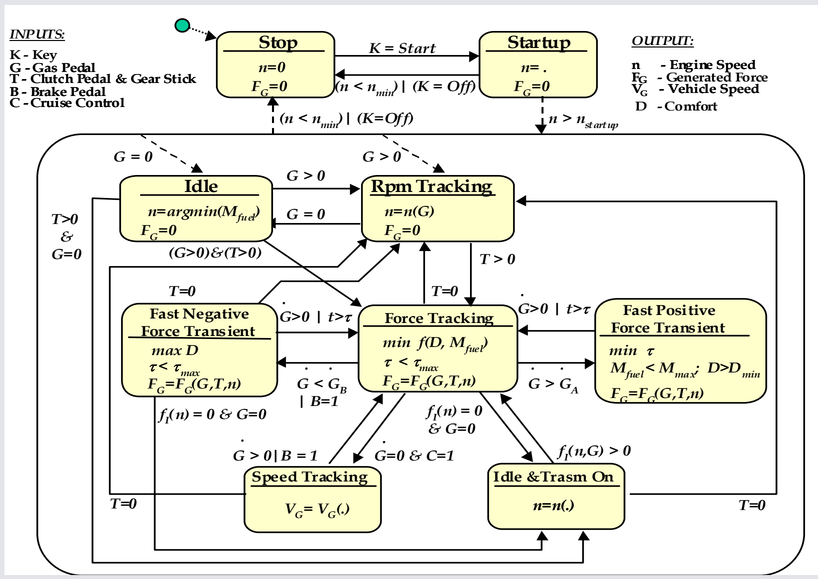
Automotive

Avionics: UAVs

Networked Embedded Systems

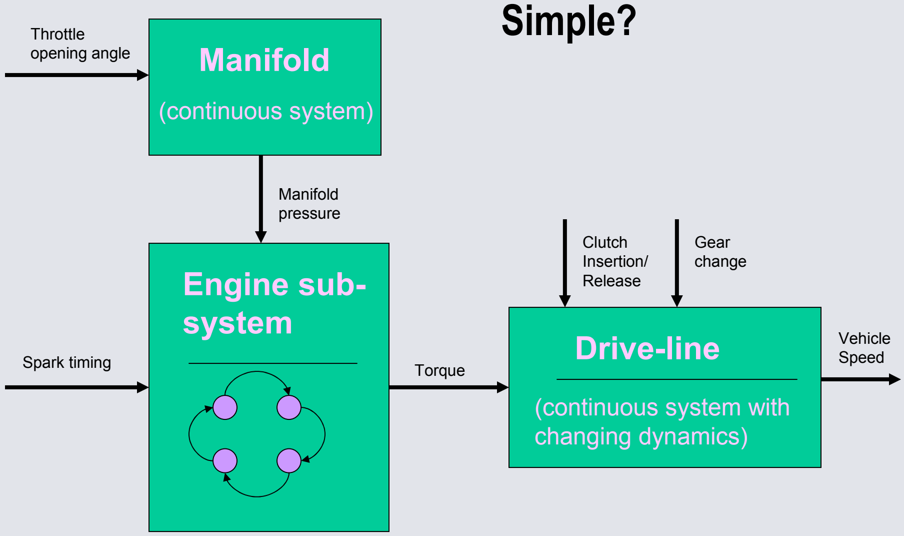
Systems Biology

Power-train Specification (ASV)

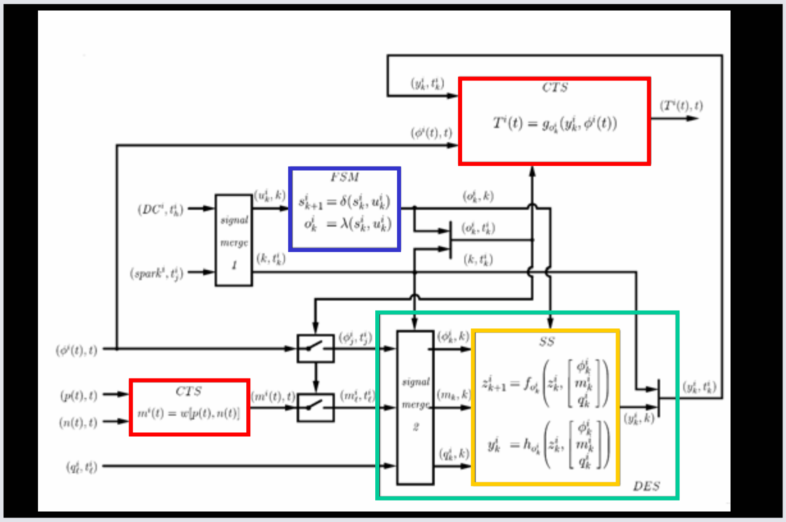


Model of Power-train (ASV)

Simple?

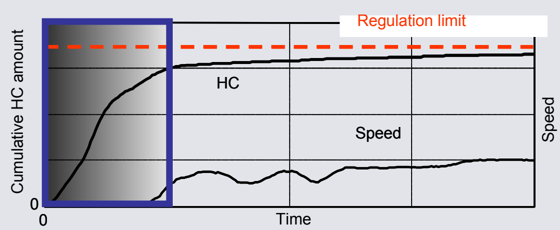


Single Cylinder Hybrid Model

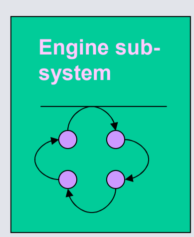


Cold-Start Problem (E. Lee, K. Hedirck, Tomizuka)

Most of the hydrocarbon emissions from an engine are emitted in the first few seconds of running. Modal control laws can mitigate this effect.



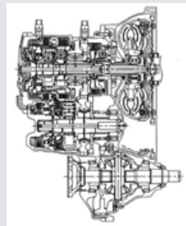
The approach is to overlay on the hybrid systems model of the cylinder cycles a modal controller, itself a hybrid system.

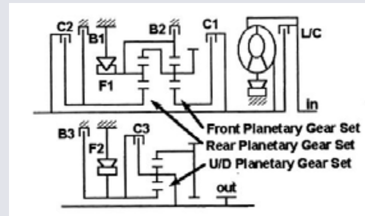


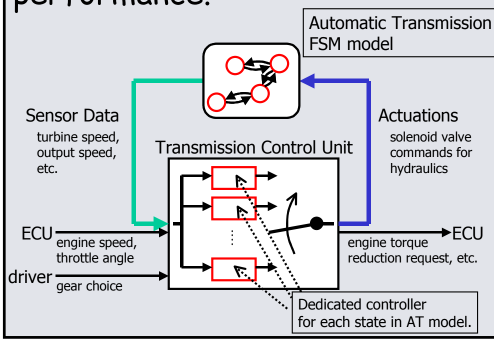
This project is pursued collaboratively with Karl Hedrick, Mechanical Engineering, and Toyota. See the poster for details.

Transmission Problem

Hybrid systems models of transmission coupled with models of engine torque control are being used to improve transmission performance.







Shift Quality Control aims to reduce undesired shock during gear shift. When engaging into a new gear, actuations (hydraulic pressures) are set such that output speed/torque have smooth transients.

This project is pursued collaboratively with Prof. Tomizuka, Mechanical Engineering, and Toyota. See the poster for details.

Outline

- Internal experimental test benches
  - Automotive
  - Wireless Sensor Networks (security, low power)
  - UAVs (complex control, sensor integration)
- New domains:
  - Scientific Workflows: Kepler
  - Stochastic Hybrid Systems in Systems Biology

Instrumenting the world

Extreme scaling

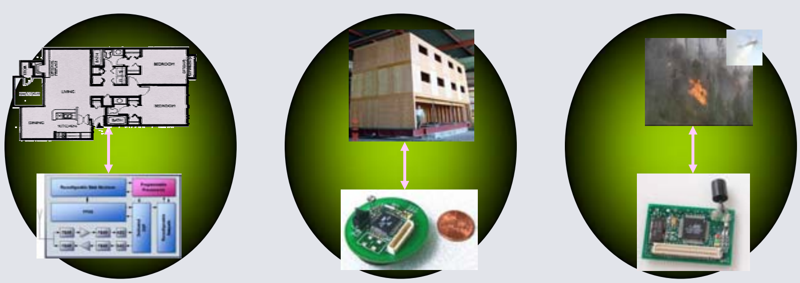
- Connected ?
- Information throughput ?
- Transmission power ?
- Routing ?
- Delay ?
- Reliability ?

Large scale networks theory

Let's look at the Application level

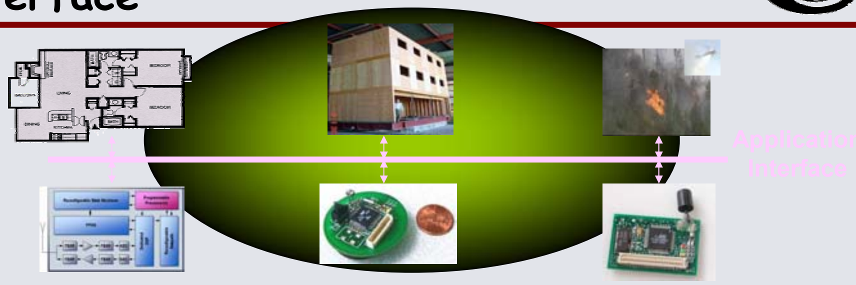
- Gilbert's model
- Extension with interference
- Spread out, unreliable connections
- Sensor placement algorithms
- Throughput capacity
- Closing the loop

Wireless Sensor Networks: Applications and Platforms Interoperability (ASV, J. Rabaey, BWRC)



- Applications ***bound to specific*** implementation platforms
- Need ***interoperability*** between applications and between implementation platforms
- Need to ***hide*** implementation details from application programmers

A Service-oriented Application Interface



- Application-level universally agreed Interface
  - In Internet Sockets support several applications and can be implemented by several protocols
- Define a standard set of services and interface primitives for Sensor Networks
  - accessible by the Application (hence called Application Interface)
  - independent on the implementation on any present and future sensor network platform

Adaptive Networked Infrastructure

Core partners: Berkeley (lead), Cornell, Vanderbilt

Outreach partners: San Jose State, Smith, Tennessee Tech, UC Davis, UC Merced.

Principal investigator: Edward A. Lee, Professor, EECS, UC Berkeley, eal@eecs.berkeley.edu

Enabling technologies: wireless networked embedded systems with sensors and actuators

Approach: Engineering methods for integrating computer-controlled, networked sensors and actuators in societal-scale infrastructure systems.

Resource management test beds:

- electric power
- transportation
- water

The ANI ERC

Deliverables: Engineering Methods, Models, and Toolkits for:

- design and analysis of systems with embedded computing
- computation integrated with the physical world
- analysis of control dynamics with software and network behavior
- programming the ensemble, not the computer
- computer-integrated systems oriented engineering curricula

Outline

- Internal experimental test benches
  - Automotive
  - Wireless Sensor Networks (security, low power)
  - UAVs (complex control, sensor integration)
- New domains:
  - Scientific Workflows: Kepler
  - Stochastic Hybrid Systems in Systems Biology

Berkeley Aerobots (Sastry, Henzinger, ASV)

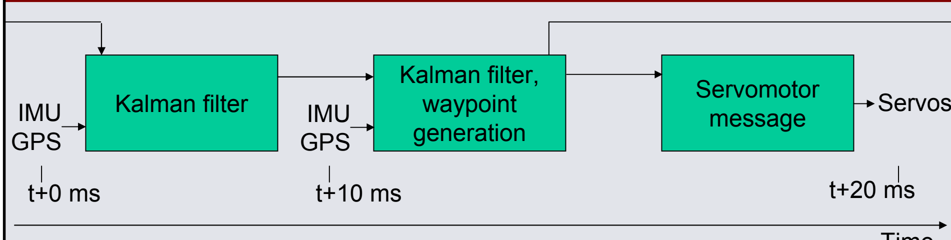
- The following have been demonstrated on the Yamaha R-50 helicopters:
  - Autonomous control.
  - Waypoint navigation.
  - Vision-based landing on a pitching deck.
  - Pursuit-evasion games with ground robots.
  - Coordinated flight with two helicopters.
  - Conflict detection and resolution of multiple helicopters

- Limitations of 1st-generation control system:
  - Undisciplined use of shared memory.
  - No synchronization between reactions to IMU and GPS messages.
  - No means of testing except for flight testing.
  - Not robust with respect to GPS failure.

2^{nd}-Generation Control System

- Our control system is based around Giotto, a time-based language for embedded systems programming.
- Giotto is ideal for correcting shortcomings of 1^{st}-generation system:
  - Deterministic semantics.
  - Disciplined use of shared memory.
  - Carefully implemented inter-task synchronization.
  - Programmer-specified latencies between sensor input and actuator output.

A Giotto-based Controller



- Giotto's time-based semantics is a natural match for the controller implementation.
  - One complication: GPS, IMU are unsynchronized.
  - Solution: treat IMU message as clock, always use most recent GPS message.
- Latency, from IMU to servos, is 10 ms. For the RMAX, this is perfectly acceptable.

Giotto Implementation

- Implementing the controller required porting the Embedded Machine, a virtual machine for executing Giotto programs, to VxWorks: 1300 lines of C code.
  - Controller functionality: 700 lines of C code.
  - E Code, executed by the Embedded Machine: 100 lines.
  - Instrumentation code (IMU, GPS, and actuators): 1800 lines of C code.
- The controller is extremely reliable. Have yet to determine mean time before failure: runs for 34 days at least
- The Giotto controller will be flown on the R-Max by December 2004.

Future Work

Using xGiotto and Metro flow, we can quickly evaluate different implementation platforms and synchronization methods

- In particular, we wish to:
  - Implement controller on top of OCP (Boeing's Open Control Platform) which is being repositoried by ESCHER
  - Make controller robust to GPS failure.

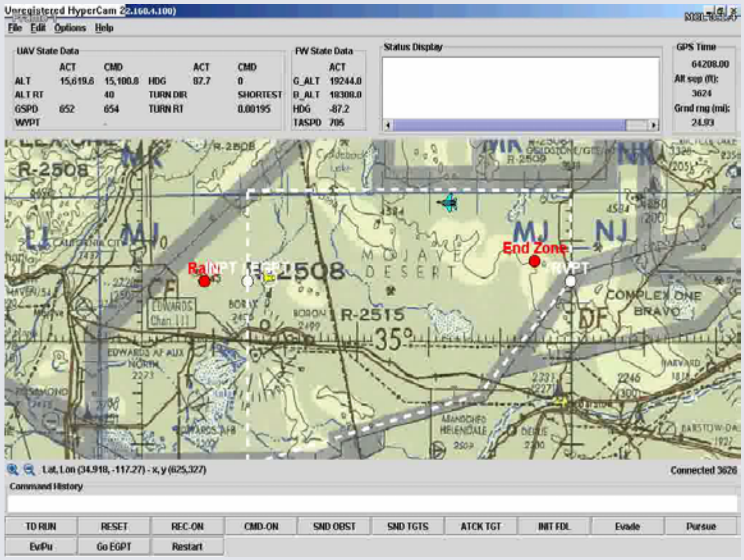
Pursuit Evasion Games (PEGs) (Sastry)

- Consider:
  - Intelligence gathering, remote-piloted UAV in flight
  - Alerted (internally, or by an observer) of an adversary over the horizon
- What to do?
  - Time lag is too large for remote "escape"
  - Adversary will most likely locate UAV shortly
  - Not enough time to transmit back all data
- Proposal:
  - Turn over control to a switched "intelligent" evader
    - Controller may also be capable of pursuit/targeting

PEGs: Status

- Experience with ground vehicles and rotorcraft combined for fixed-wing demo
- Reconfigurable nonlinear-model predictive control used to control evasive/pursuit maneuvers
- Flew in Boeing T-33 Jet, June 2004
  - Positive response from pursuing/"caught" pilot in F-15, "behaved just like a human"
- Possible applications for PEGs with autonomous vehicles
  - NASA robotics (long lag control)
  - Underwater security (unmanned submarines)

PEGs: Application Demonstration



Outline

- Internal experimental test benches
  - Automotive
  - Wireless Sensor Networks (security, low power)
  - UAVs (complex control, sensor integration)
- New domains:
  - Scientific Workflows: Kepler
  - Stochastic Hybrid Systems in Systems Biology

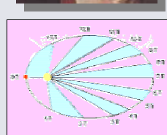
A New Application of Ptolemy II Kepler – Scientific Workflows (Lee)

A distributed workflow system to "provide access to any resource, any data, any service, anytime, anywhere" via a distributed computing platform (aka the "Grid") supporting "high performance computing, and federated, integrated, mediated databases within a user-friendly workbench / problem-solving environment".

This effort is compatible with the objectives of NSF Cyberinfrastructure, UK's e-Science Programme, and DOE's SciDAC (Scientific Discovery through Advanced Computing).

KEPLER/CSP: Contributors, Sponsors, Projects
(or loosely coupled Communicating Sequential Persons ;-)

Ilkay Altintas *SDM, Resurgence*
Kim Baldridge *Resurgence, NMI*
Chad Berkley *SEEK*
Shawn Bowers *SEEK*
Terence Critchlow *SDM*
Tobin Fricke *ROADNet*
Jeffrey Grethe *BIRN*
Christopher H. Brooks *Ptolemy II*
Zhengang Cheng *SDM*
Dan Higgins *SEEK*
Efrat Jaeger *GEON*
Matt Jones *SEEK*
Werner Krebs, *EOL*
Edward A. Lee *Ptolemy II*
Kai Lin *GEON*
Bertram Ludaescher *SEEK, GEON, SDM, BIRN, ROADNet*
Mark Miller *EOL*
Steve Mock *NMI*
Steve Neuendorffer *Ptolemy II*
Jing Tao *SEEK*
Mladen Vouk *SDM*
Xiaowen Xin *SDM*
Yang Zhao *Ptolemy II*
Bing Zhu *SEEK*
...



Ptolemy II

ROADNet

SEEK

GEON The Geosciences Network

BIRN Biomedical Informatics Research Network

SciDAC Scientific Discovery through Advanced Computing

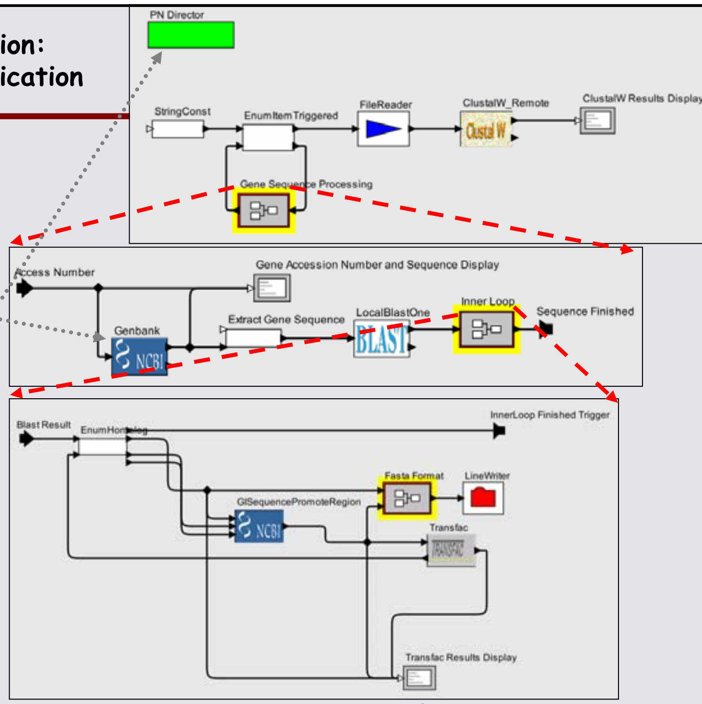
SDM

DARPA NSF

Thanks to Bertram Ludäscher, SDSC

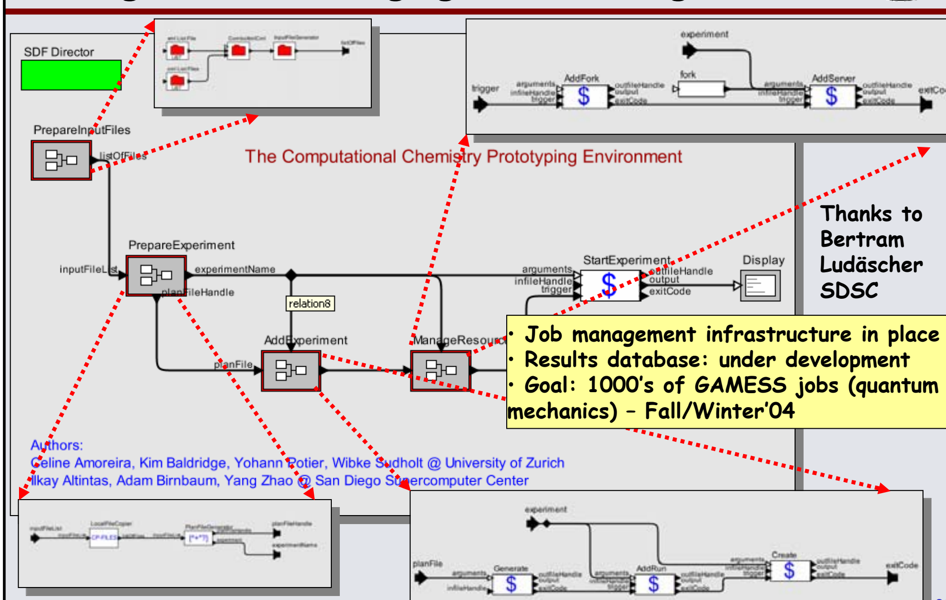
Example Application: Promoter Identification

- Scientist models application as a "workflow" of connected components ("**actors**")
- If all components exist, the workflow can be automated/ executed
- Different **directors** can be used to pick appropriate execution model (often "pipelined" execution: PN director)



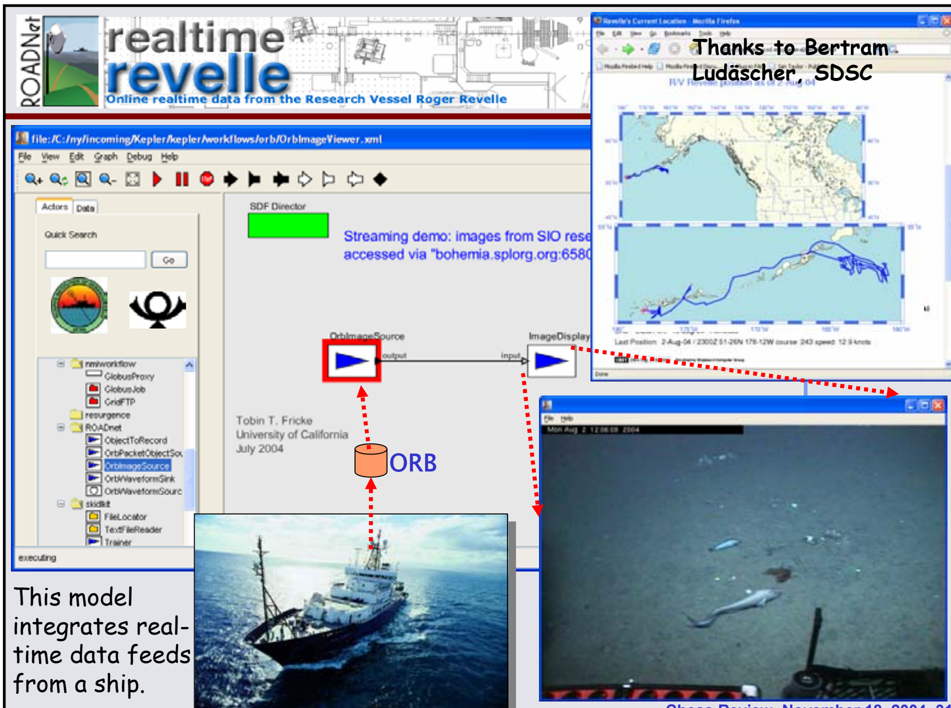
Thanks to
Bertram Ludäscher
SDSC

Another Application: Distributed Job Management (leveraging NIMROD, grid tool)



Thanks to
Bertram Ludäscher
SDSC

- Job management infrastructure in place
- Results database: under development
- Goal: 1000's of GAMESS jobs (quantum mechanics) – Fall/Winter'04



This model integrates real-time data feeds from a ship.

Outline

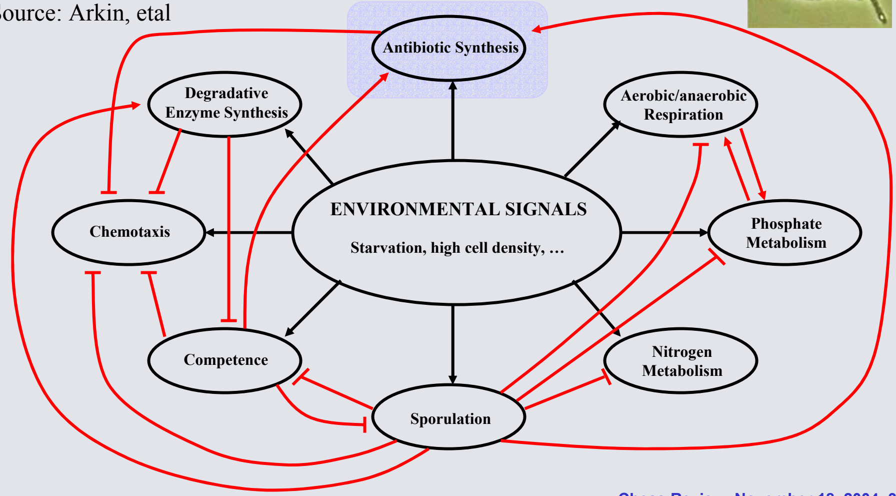
- Internal experimental test benches
  - Wireless Sensor Networks (security, low power)
  - UAVs (complex control, sensor integration)
- New domains:
  - Scientific Workflows: Kepler
  - Stochastic Hybrid Systems in Systems Biology

Example: B. Subtilis Antibiotic Production (Sastry)

Stress Response Network for B. Subtilis

Focusing on subtilin production

Source: Arkin, etal



Example: B. Subtilis Antibiotic Production

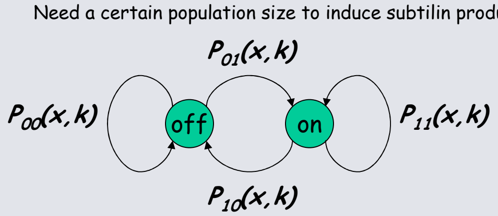
- *K*: total number of *B. subtilis* in a population
- *S*: number of *B. subtilis* producing subtilin
- *X*: amount of food available

$$dX(t)/dt = v_{in}+(K-S)v_0+Sv_1, \quad v_0 < 0 < v_1$$
$$dK(t)/dt = rK[1-K/f(X)], \quad r\text{: growth rate}$$

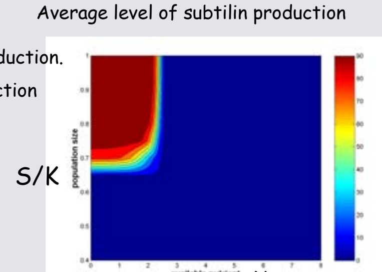
Transition probability P(X,K) of each B. subtilis

Small X ⇒ high probability of switching on subtilin production.

Need a certain population size to induce subtilin production



Average level of subtilin production



Stochastic Hybrid Systems

- A set of discrete states and *open* domains
- Dynamics inside each domain governed by a SDE
- Stop upon hitting domain boundary
- Boundary of each domain is partitioned into guards
- Jump to a new discrete state accordingly
- Reset randomly in the new domain

Challenges of Systems (post genomic biology)

- Hybrid Systems Models for Intracellular functioning: stochastic hybrid systems
- Hybrid Systems tools for ensembles of cells: group behavior of complex networked systems
- Biologically complex networks are an exemplar of how networked embedded systems could evolve, self-organize and reprogram themselves (network programming?)

Concluding Remarks

- Applications are critical to drive research and to test quality of results
- Safety-critical Real Time and secure system emphasis
- Industrial and Experimental Test Benches
- Rigorous methodology for hybrid distributed systems