Lecture 9 – Modeling, Simulation, and Systems Engineering

- Development steps
- Model-based control engineering
- Modeling and simulation
- Systems platform: hardware, systems software.

Control Engineering Technology

- Science
  - abstraction
  - concepts
  - simplified models
- Engineering
  - building new things
  - constrained resources: time, money,
- Technology
  - repeatable processes

- Control platform technology
- Control engineering technology

Controls development cycle

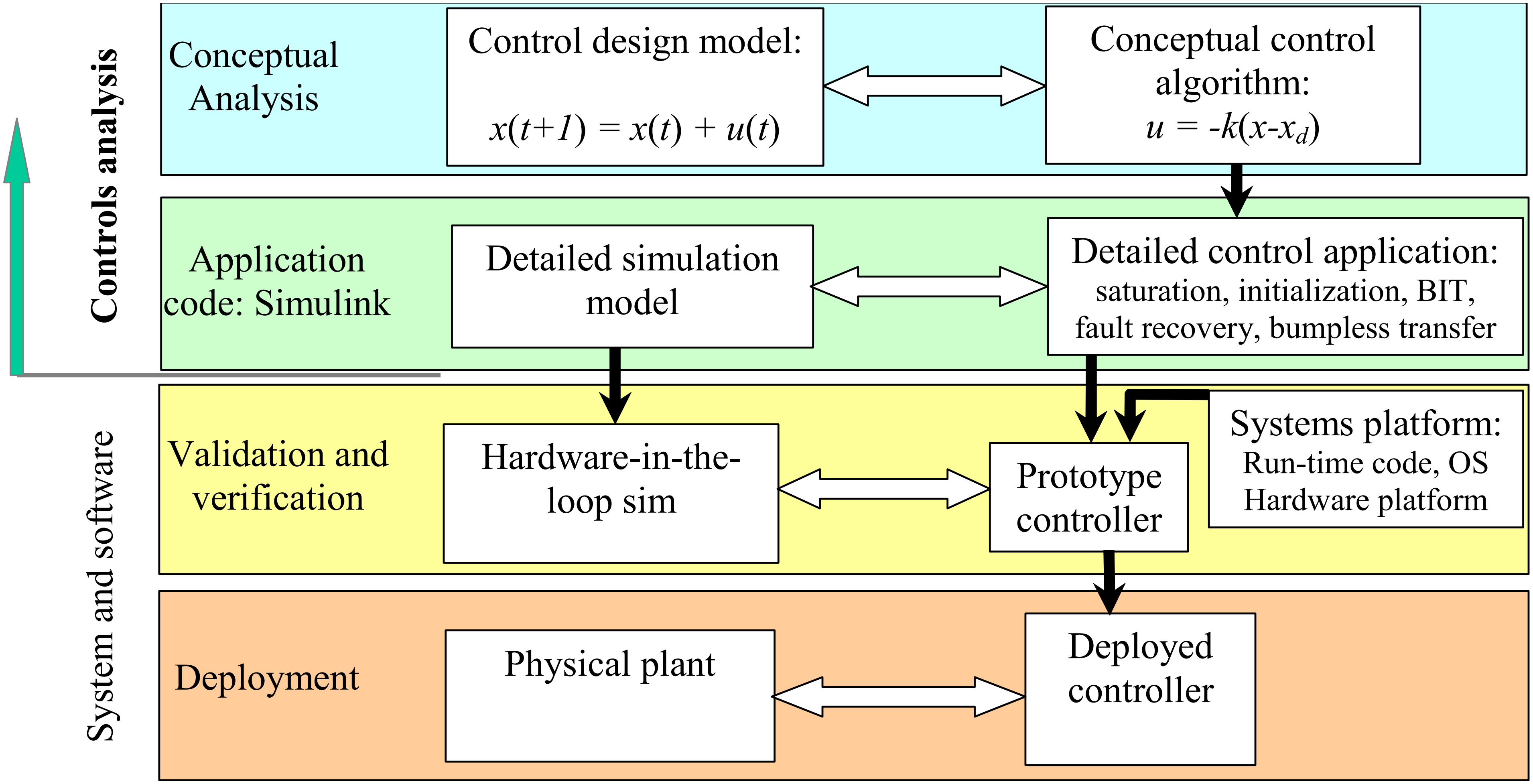
- Analysis and modeling
  - Control algorithm design using a simplified model
  - System trade study - defines overall system design
- Simulation
  - Detailed model: physics, or empirical, or data driven
  - Design validation using detailed performance model
- System development
  - Control application software
  - Real-time software platform
  - Hardware platform
- Validation and verification
  - Performance against initial specs
  - Software verification
  - Certification/commissioning

Algorithms/Analysis

Much more than real-time control feedback computations

- modeling
- identification
- tuning
- optimization
- feedforward
- feedback
- estimation and navigation
- user interface
- diagnostics and system self-test
- system level logic, mode change

Model-based Control Development



Controls analysis

System and software

Conceptual Analysis

Control design model:

$x(t+1) = x(t) + u(t)$

Conceptual control algorithm:

$u = -k(x-x_d)$

Application code: Simulink

Detailed simulation model

Detailed control application: saturation, initialization, BIT, fault recovery, bumpless transfer

Validation and verification

Hardware-in-the-loop sim

Prototype controller

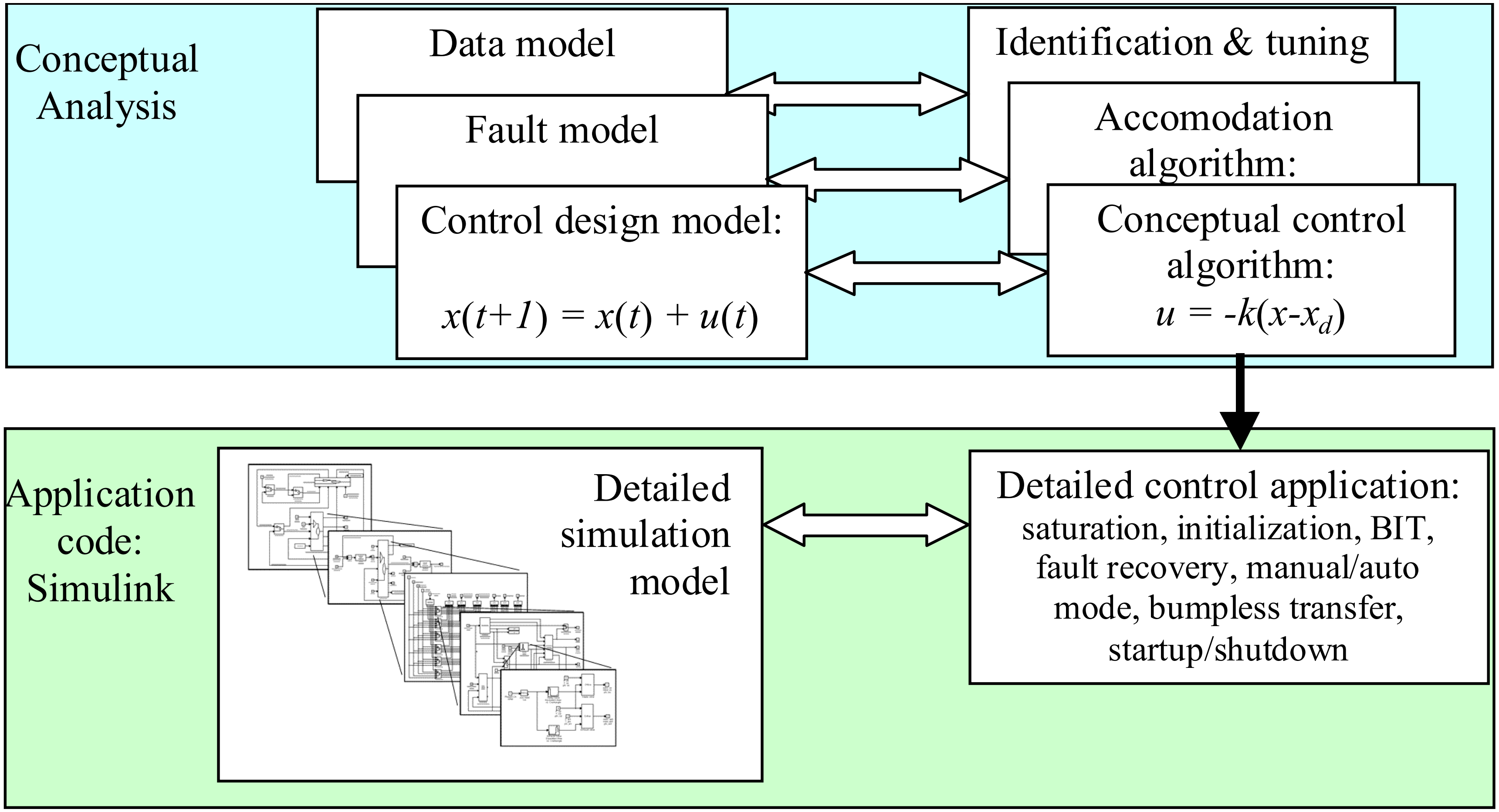
Systems platform: Run-time code, OS Hardware platform

Deployment

Physical plant

Deployed controller

Controls Analysis



Conceptual Analysis

Data model

Fault model

Control design model:

$x(t+1) = x(t) + u(t)$

Identification & tuning

Accomodation algorithm:

Conceptual control algorithm:

$u = -k(x-x_d)$

Application code: Simulink

Detailed simulation model

Detailed control application:

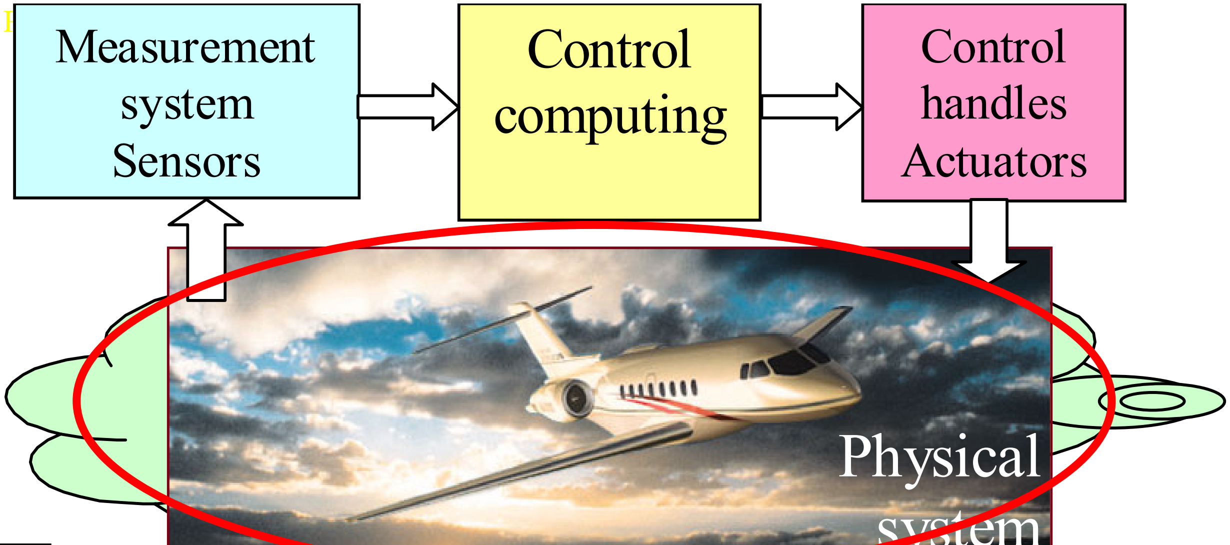
saturation, initialization, BIT, fault recovery, manual/auto mode, bumpless transfer, startup/shutdown

The rest of the lecture

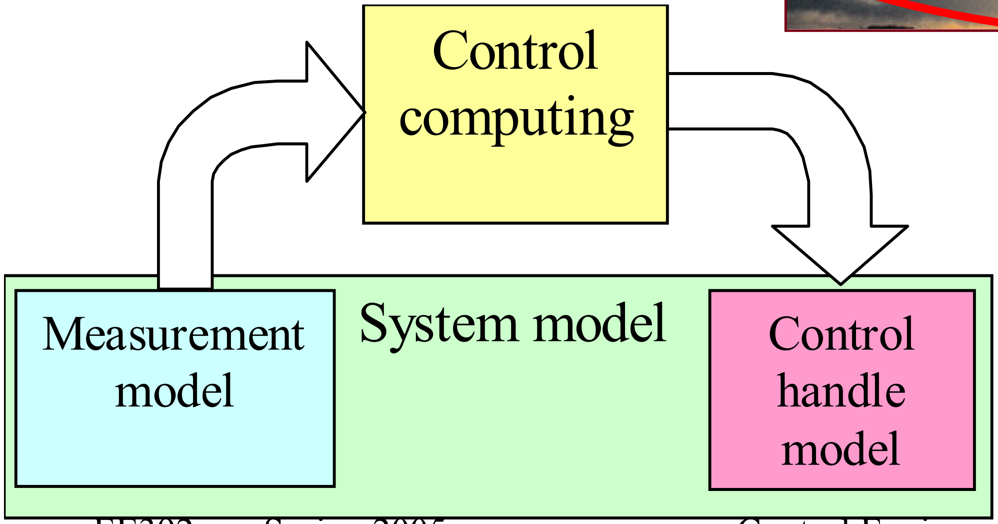
- Modeling and Simulation
- Deployment Platform
- Controls Software Development

Modeling in Control Engineering

- Control in a system perspective



- Control analysis perspective



Models

- Why spend much time talking about models?
  - Modeling and simulation could take 80% of control analysis effort.
- Model is a mathematical representations of a system
  - Models allow simulating and analyzing the system
  - Models are never exact
- Modeling depends on your goal
  - A single system may have many models
  - Large ‘libraries’ of standard model templates exist
  - A conceptually new model is a big deal (economics, biology)
- Main goals of modeling in control engineering
  - conceptual analysis
  - detailed simulation

Modeling approaches

- Controls analysis uses deterministic models. Randomness and uncertainty are usually not dominant.
- White box models: physics described by ODE and/or PDE
- Dynamics, Newton mechanics

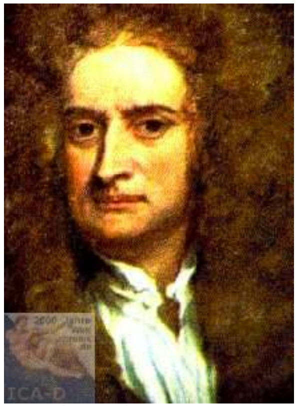
$$\dot{x} = f(x,t)$$

- Space flight: add control inputs u and measured outputs y

$$\dot{x} = f(x,u,t)$$

$$y = g(x,u,t)$$

Orbital mechanics example

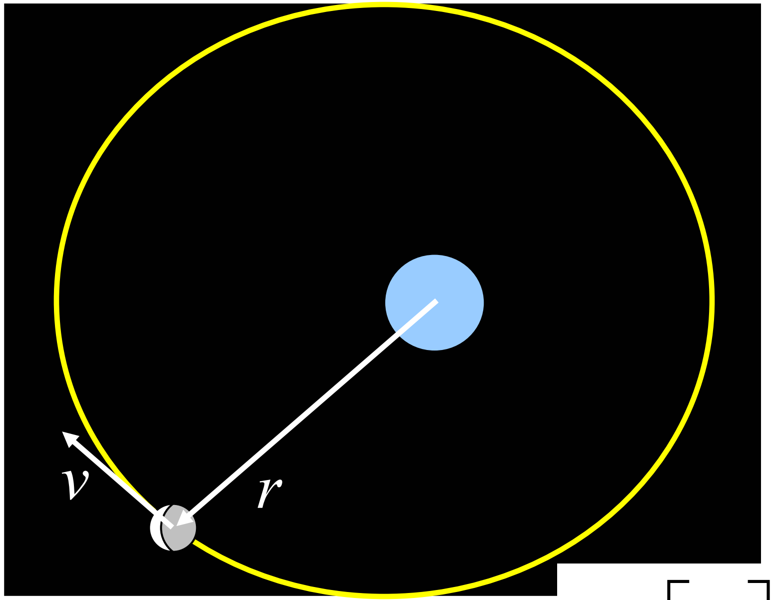


1643-1736

- Newton's mechanics
  - fundamental laws
  - dynamics

$$\dot{v} = -\gamma m \cdot \frac{r}{|r|^3} + F_{pert}(t)$$

$$\dot{r} = v$$



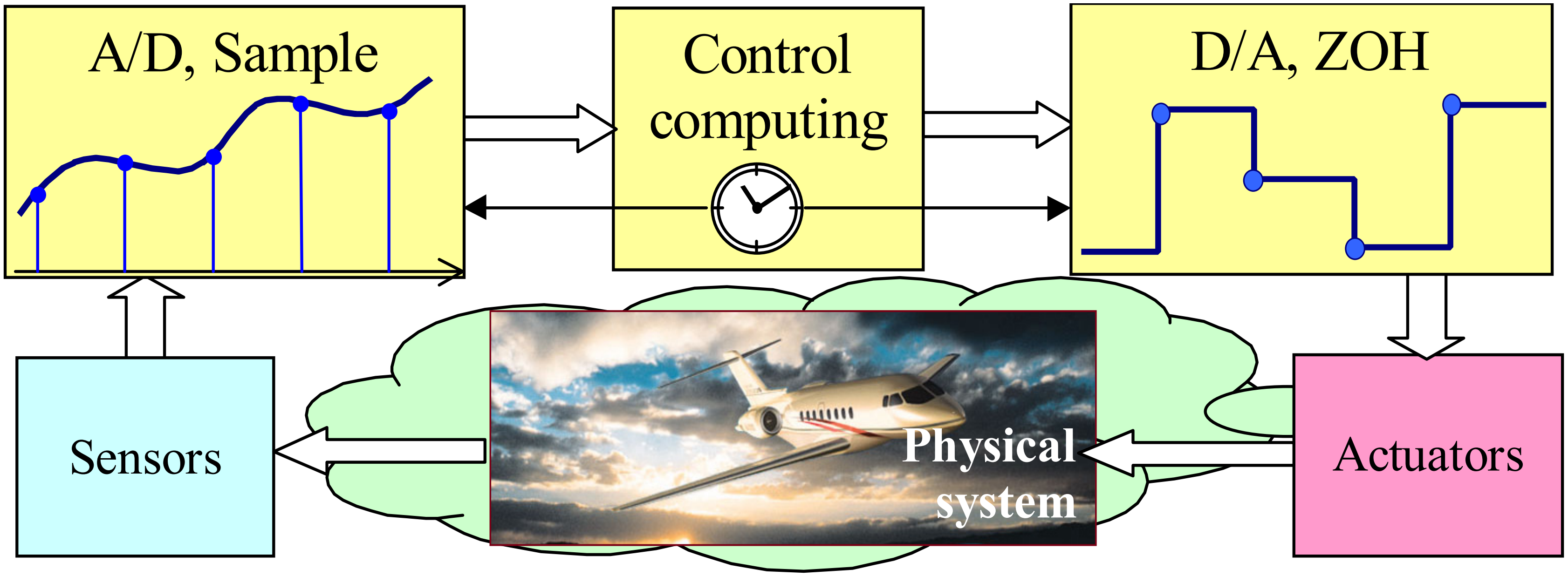
1749-1827

- Laplace
  - computational dynamics (pencil & paper computations)
  - deterministic model-based prediction

$$\dot{x} = f(x,t) \qquad x = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

Sampled and continuous time

- Sampled and continuous time together
- Continuous time physical system + digital controller
  - ZOH = Zero Order Hold



A/D, Sample

Control computing

D/A, ZOH

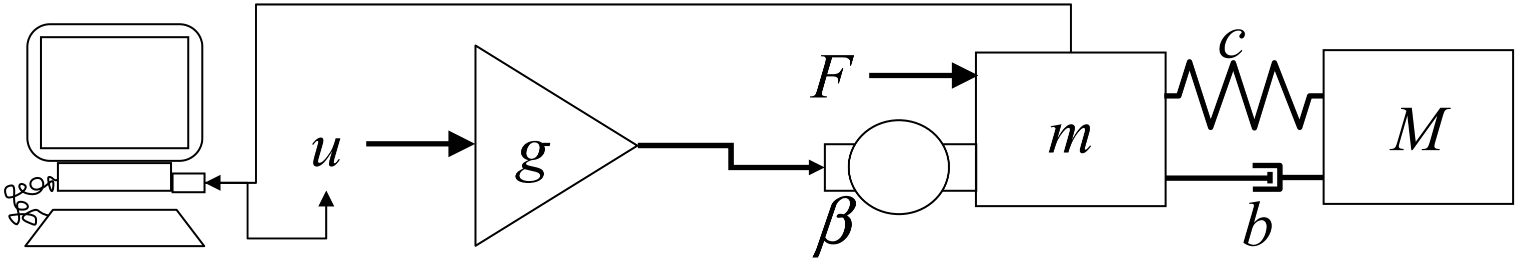
Sensors

Physical system

Actuators

Servo-system modeling

- Mid-term problem
- First principle model: electro-mechanical + computer sampling
- Parameters follow from the specs



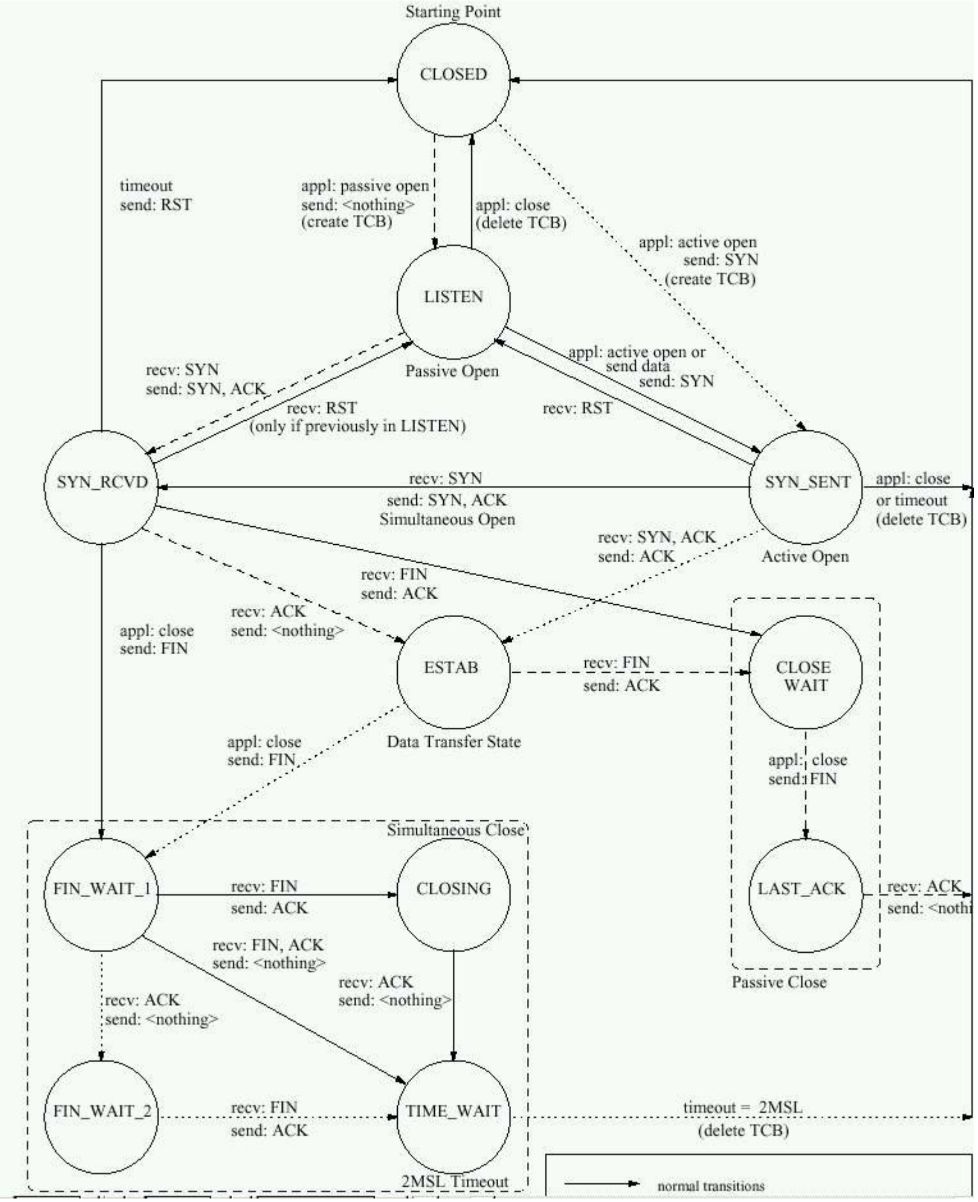
$$m\ddot{y} + \beta\dot{y} + b(\dot{y} - \dot{x}) + c(y - x) = F$$

$$M\ddot{x} + b(\dot{x} - \dot{y}) + c(x - y) = 0$$

$$F = fI, \qquad T_I\dot{I} + I = gu$$

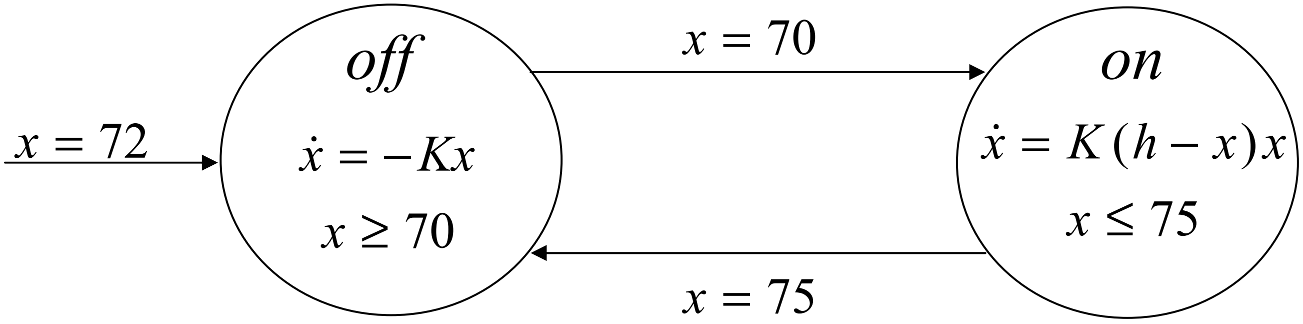
Finite state machines

- TCP/IP State Machine



Hybrid systems

- Combination of continuous-time dynamics and a state machine
- Thermostat example
- Analytical tools are not fully established yet
- Simulation analysis tools are available
  - Stateflow by Mathworks



$x = 72$

off

$\dot{x} = -Kx$

$x \geq 70$

$x = 70$

on

$\dot{x} = K(h - x)x$

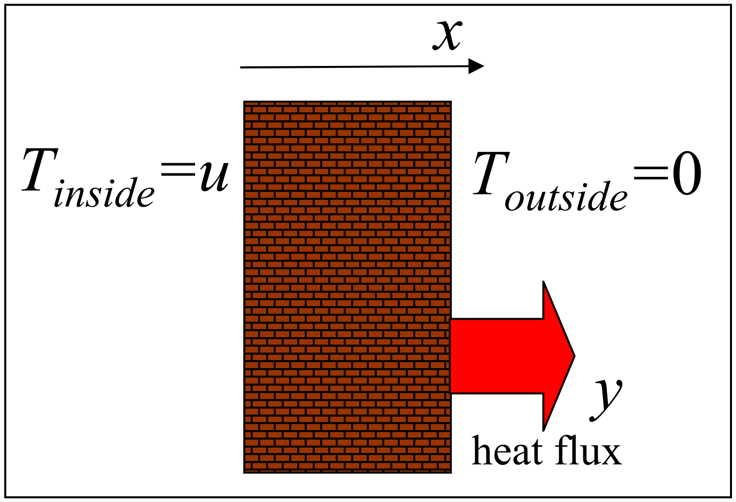
$x \leq 75$

$x = 75$

PDE models

- Include functions of spatial variables
  - electromagnetic fields
  - mass and heat transfer
  - fluid dynamics
  - structural deformations
- For 'controls' simulation, model reduction step is necessary
  - Usually done with FEM/CFD data
  - Example: fit step response

Example: sideways heat equation



$$\frac{\partial T}{\partial t} = k\frac{\partial^2 T}{\partial x^2}$$

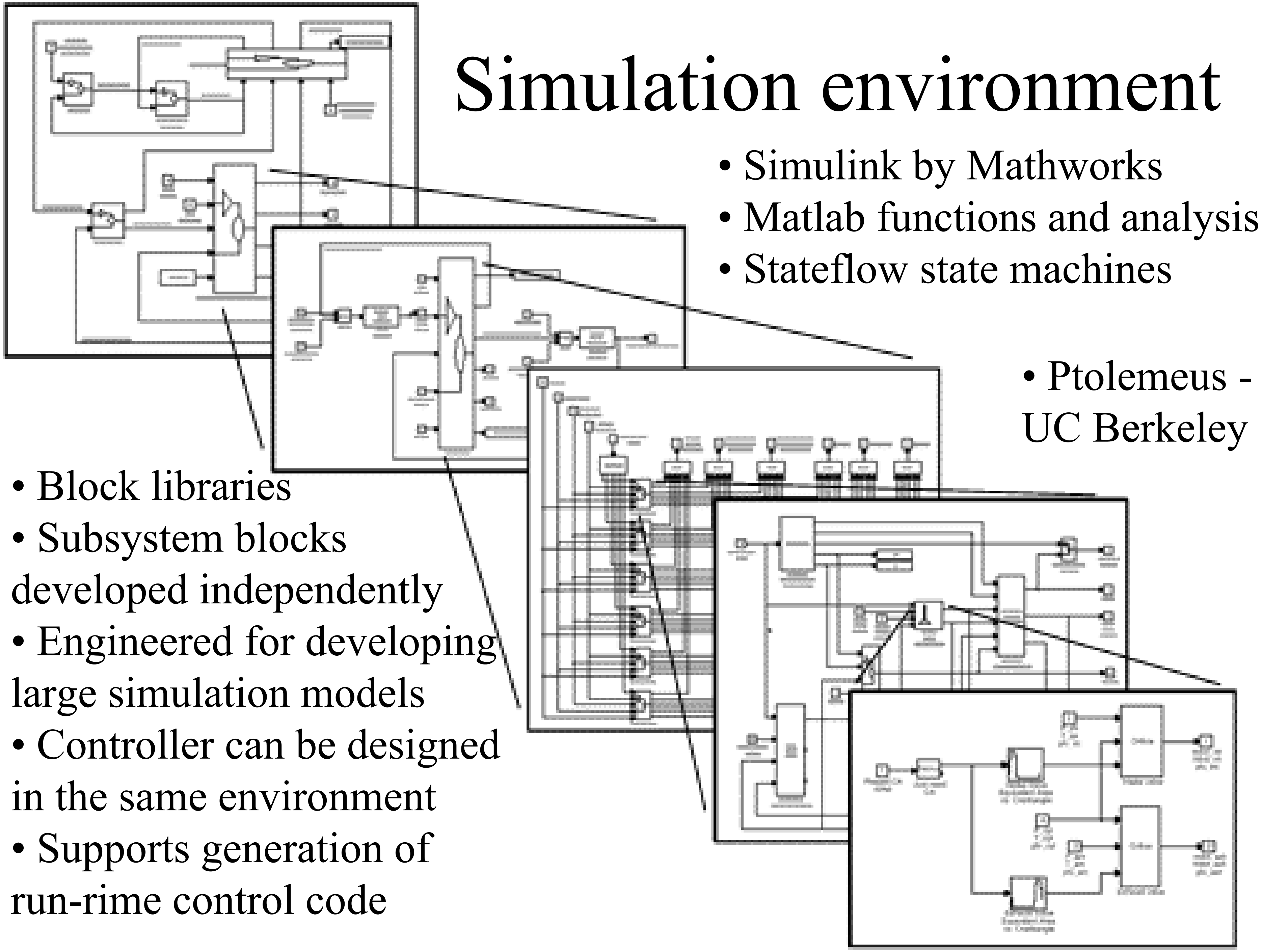
$$T(0) = u; \qquad T(1) = 0$$

$$y = \left.\frac{\partial T}{\partial x}\right|_{x=1}$$

Simulation

- ODE solution
  - dynamical model: $\dot{x} = f(x,t)$
  - Euler integration method: $x(t+d) = x(t) + d \cdot f\left(x(t),t\right)$
  - Runge-Kutta method: **`ode45`** in Matlab
- Can do simple problems by integrating ODEs
- Issues with modeling of engineered systems:
  - stiff systems, algebraic loops
  - mixture of continuous and sampled time
  - state machines and hybrid logic (conditions)
  - systems build of many subsystems
  - large projects, many people contribute different subsystems

Simulation environment



- Simulink by Mathworks
- Matlab functions and analysis
- Stateflow state machines

- Ptolemeus - UC Berkeley

- Block libraries
- Subsystem blocks developed independently
- Engineered for developing large simulation models
- Controller can be designed in the same environment
- Supports generation of run-rime control code

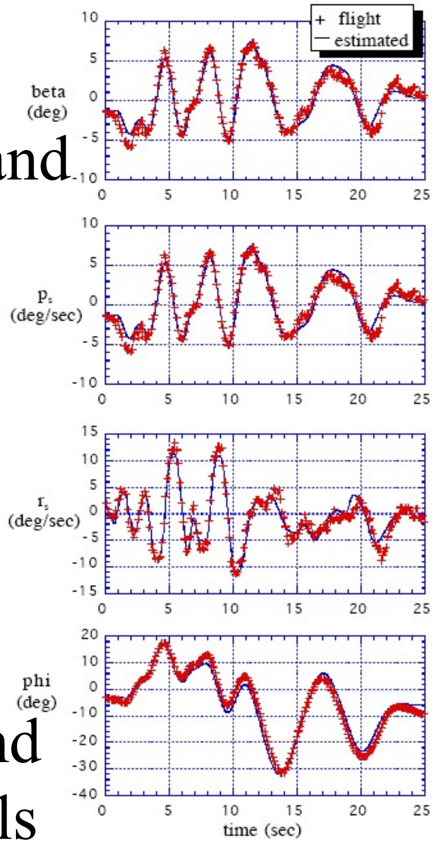
Model development and validation

- Model development is a skill

- White box models: first principles
- Black box models: data driven
- Gray box models: with some unknown parameters
- Identification of model parameters – necessary step
  - Assume known model structure
  - Collect plant data: special experiment or normal operation
  - Tweak model parameters to achieve a good fit

First Principle Models - Aerospace

- Aircraft models
- Component and subsystem modeling and testing
- CFD analysis
- Wind tunnel tests – to adjust models (fugde factors)
- Flight tests – update aerodynamic tables and flight dynamics models

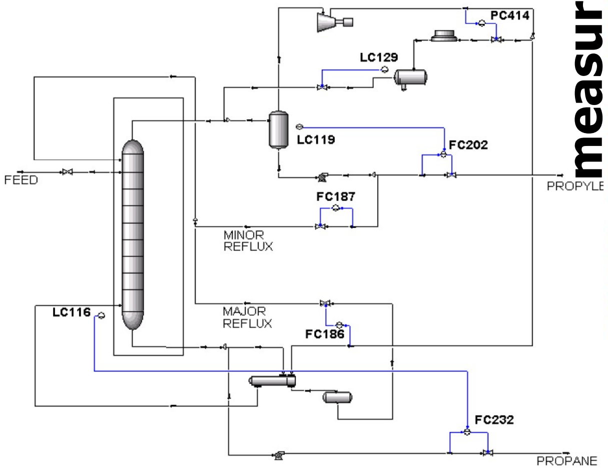


NASA Langley – 1998
HARV – F/A-18

Airbus 380: $13B development

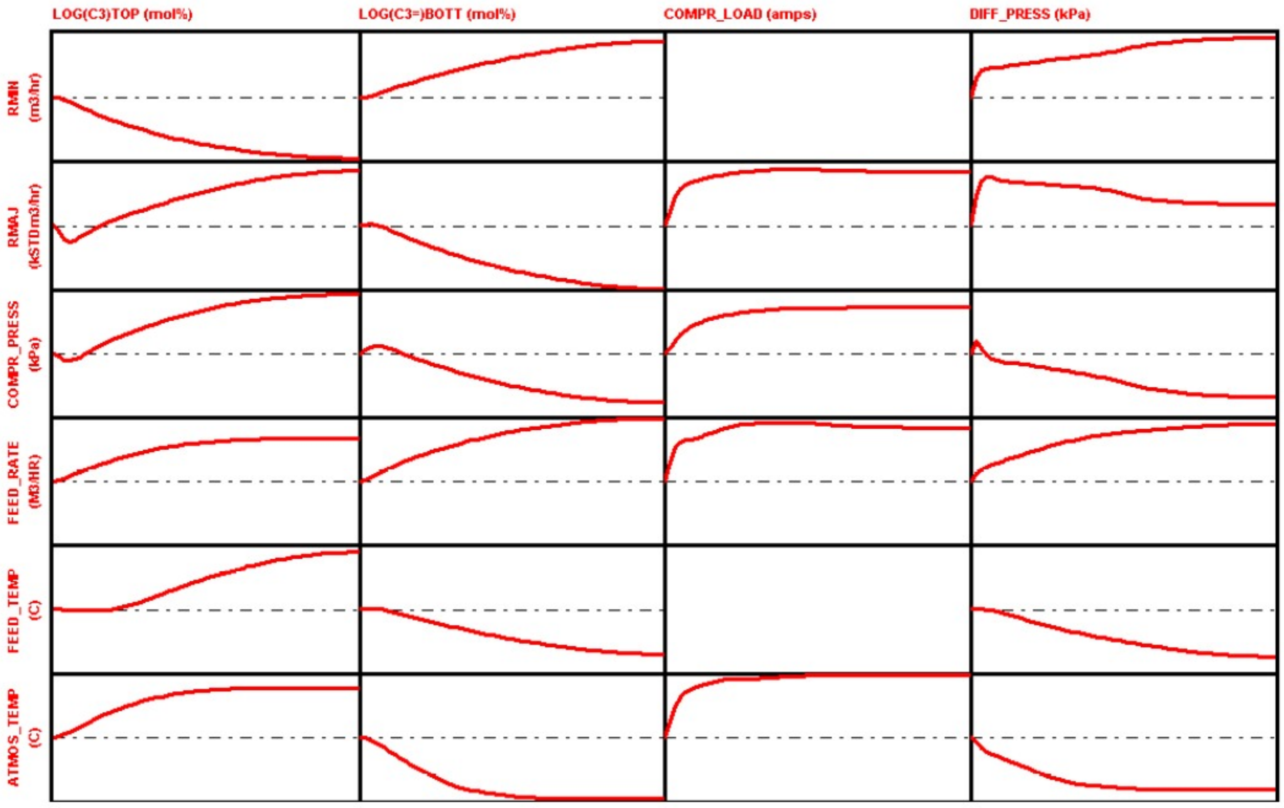
Step Response Model - Process

- Dynamical matrix control (DMC)
- Industrial processes



control inputs

measured outputs



Dynamic Matrix for Propylene/Propane Splitter

Approximate Maps

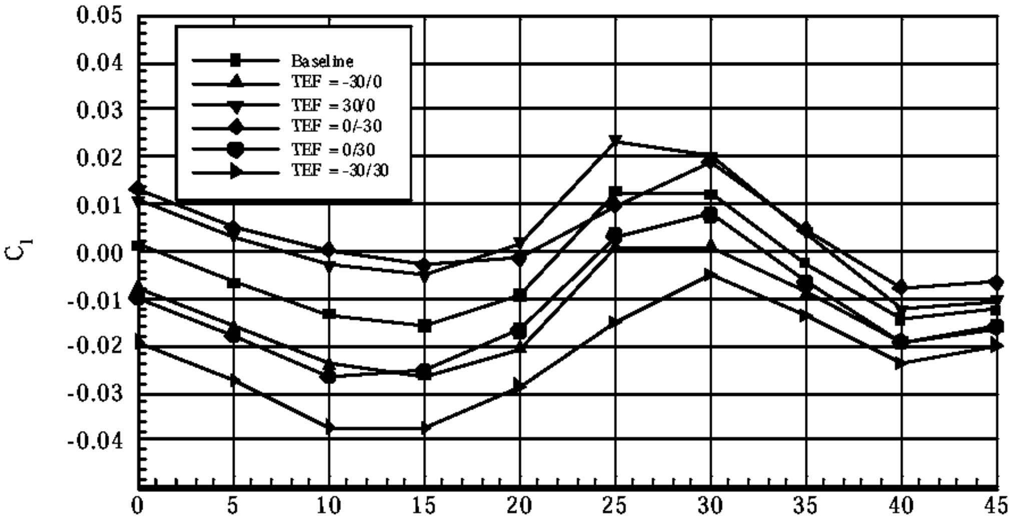
- Analytical expressions are rarely sufficient in practice
- Models are computable off line
  - pre-compute simple approximation
  - on-line approximation
- Models contain data identified in the experiments
  - nonlinear maps
  - interpolation or look-up tables
  - AI approximation methods
    - Neural networks
    - Fuzzy logic
    - Direct data driven models

Empirical Models - Maps

- Aerospace and automotive – have most developed modeling approaches

- Aerodynamic tables
- Engine maps
  - turbines – jet engines
  - automotive - ICE

Example

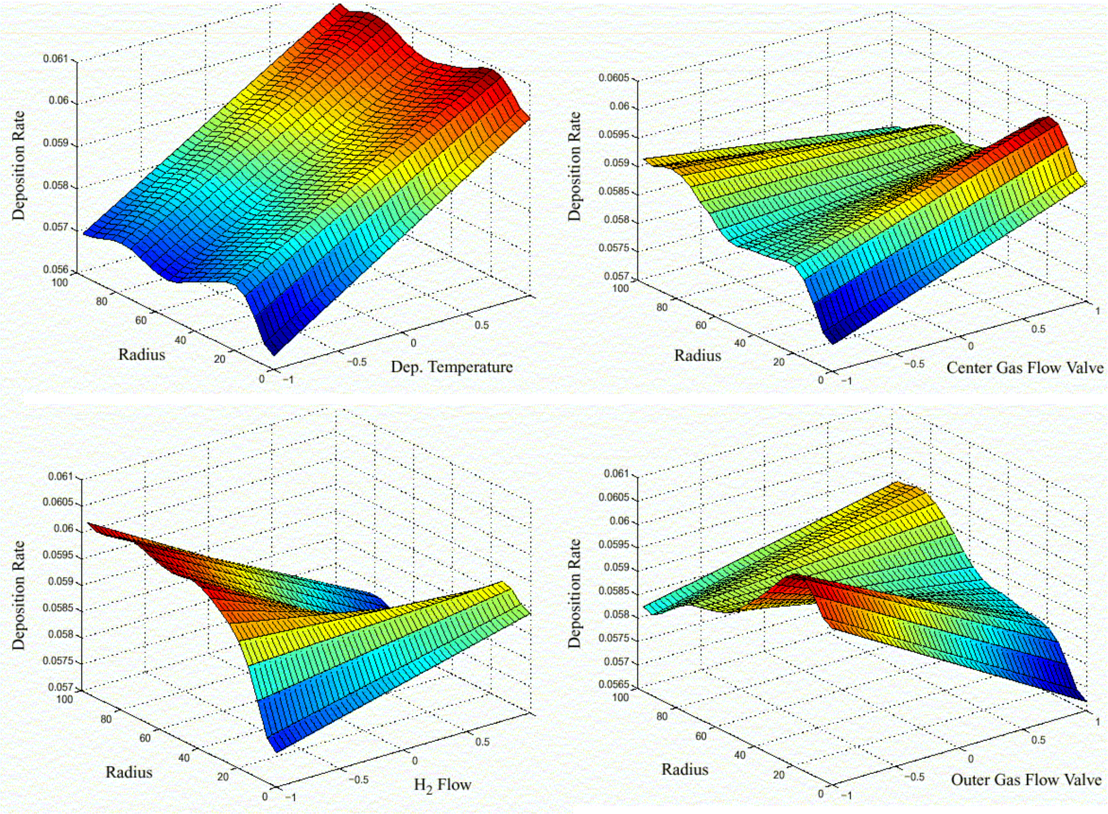


Rolling-Moment Coefficient as a Function of Angle of Attack with TEF Deflections, β = 10°.

TEF=Trailing Edge Flap

Empirical Models - Maps

- Process control mostly uses empirical models

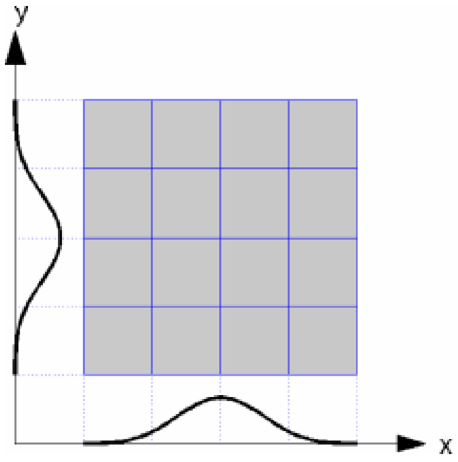


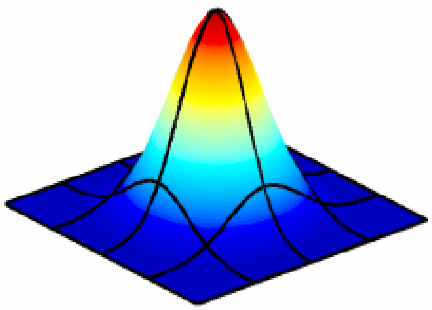
- Process maps in semiconductor manufacturing
- Epitaxial growth (semiconductor process)
  - process map for run-to-run control

Multivariable B-splines

- Regular grid in multiple variables
- Tensor product of B-splines
- Used as a basis of finite-element models

$$y(u,v) = \sum_{j,k} w_{j,k} B_j(u) B_k(v)$$



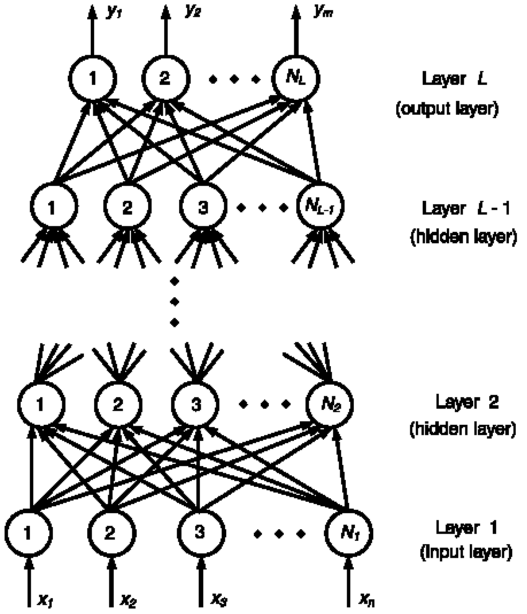


Neural Networks

- Any nonlinear approximator might be called a Neural Network
  - RBF Neural Network
  - Polynomial Neural Network
  - B-spline Neural Network
  - Wavelet Neural Network

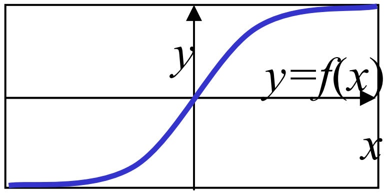
Linear in parameters

- MPL - Multilayered Perceptron
  - Nonlinear in parameters
  - Works for many inputs



$$y(\bar{x}) = w_{1,0} + f\left(\sum_j w_{1,j} y_j^1\right), y_j^1 = w_{2,0} + f\left(\sum_j w_{2,j} x_j\right)$$

$$f(x) = \frac{1-e^{-x}}{1+e^{-x}}$$



Multi-Layered Perceptrons

- Network parameter computation
  - training data set
  - parameter identification

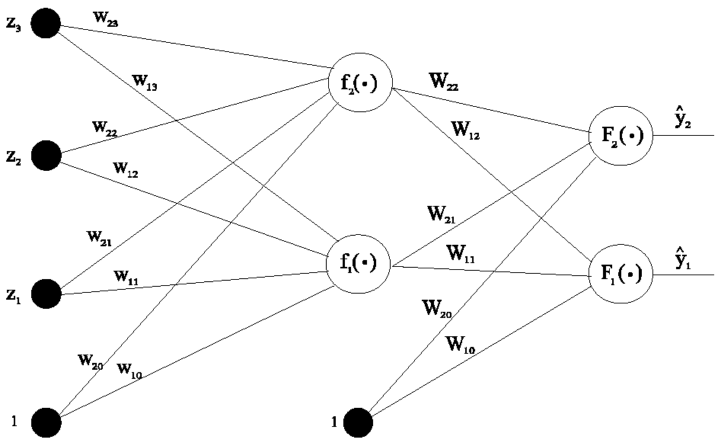
$$y(\overline{x}) = F(\overline{x};\theta)$$

- Noninear LS problem

$$V = \sum_j \left\| y^{(j)} - F(\overline{x}^{(j)};\theta) \right\|^2 \rightarrow \min$$

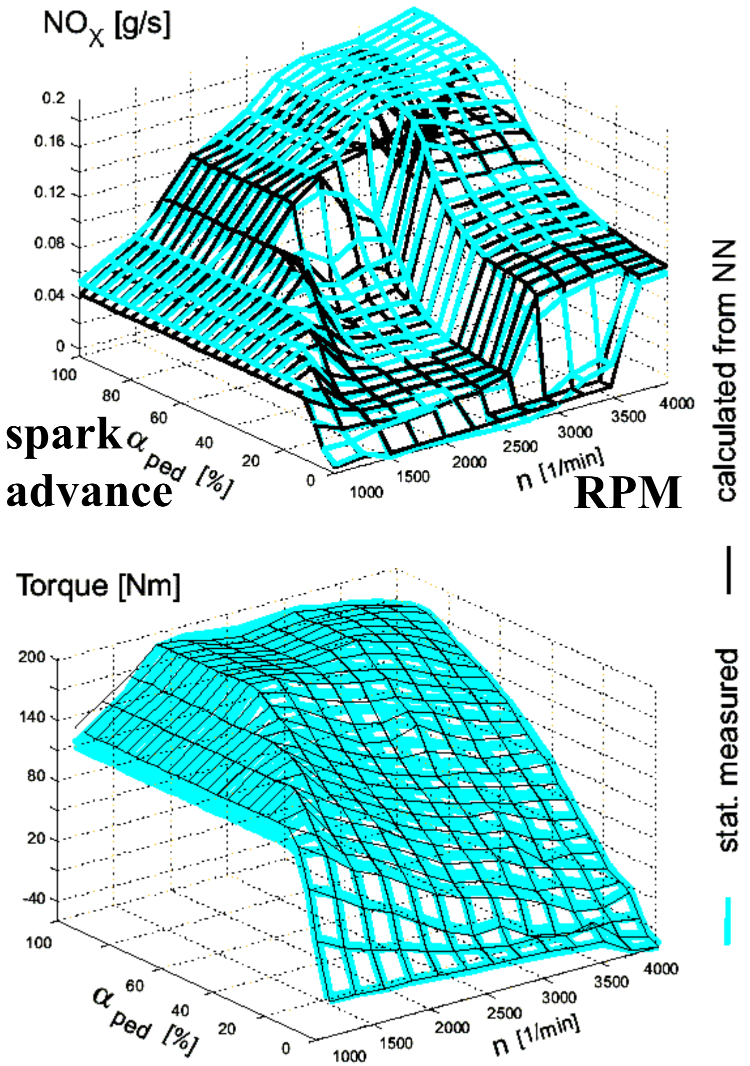
- Iterative NLS optimization
  - Levenberg-Marquardt
- Backpropagation
  - variation of a gradient descent

$$Y = \left[\underset{\overline{x}^{(1)}}{\underset{\nearrow}{y^{(1)}}} \quad \underset{\dots}{\dots} \quad \underset{\overline{x}^{(N)}}{\underset{\nearrow}{y^{(N)}}} \right]$$



Neural Net application

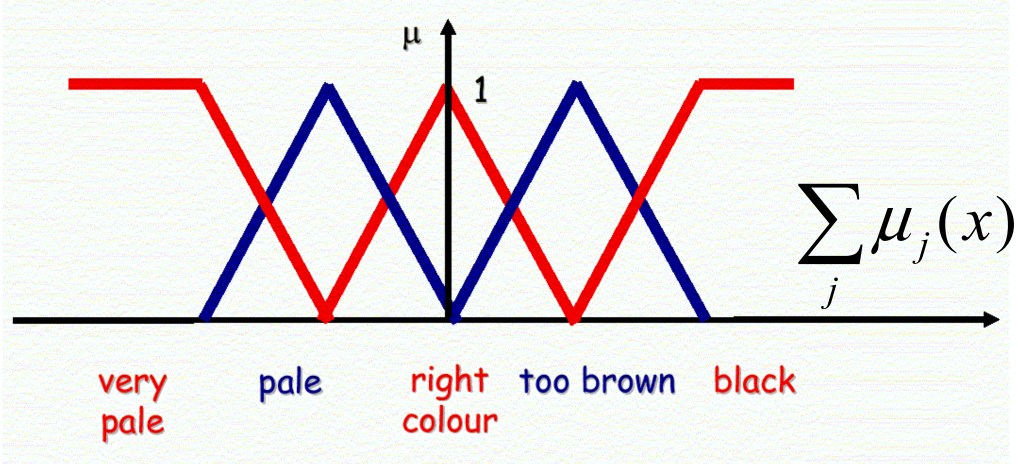
- Internal Combustion Engine maps
- Experimental map:
  - data collected in a steady state regime for various combination of parameters
  - 2-D table
- NN map
  - approximation of the experimental map
  - MLP was used in this example
  - works better for a smooth surface



Fuzzy Logic

- Function defined at nodes. Interpolation scheme
- Fuzzyfication/de-fuzzyfication = interpolation
- Linear interpolation in 1-D

$$y(x) = \frac{\sum_j y_j \mu_j(x)}{\sum_j \mu_j(x)}$$

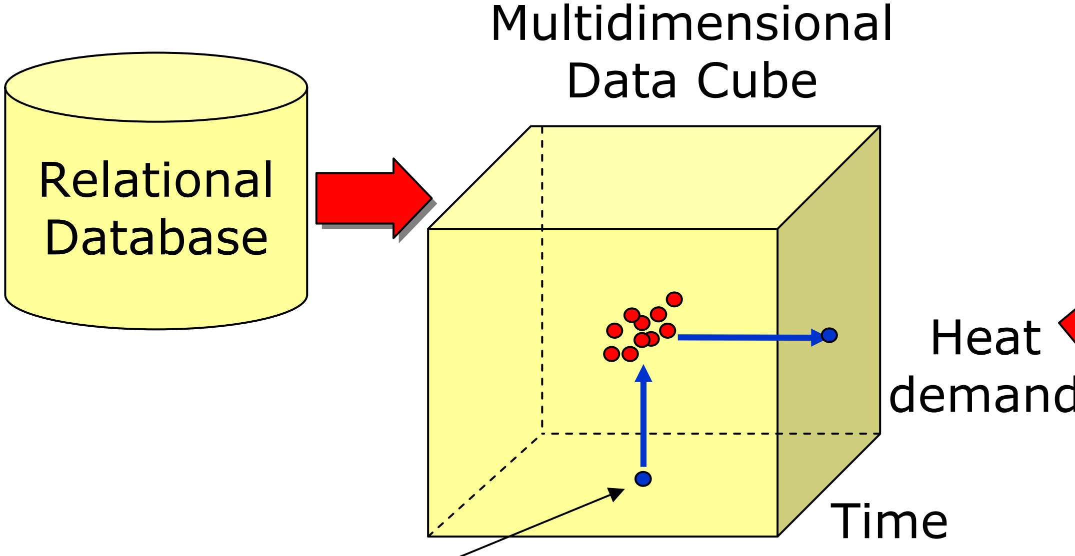


$$\sum_j \mu_j(x) = 1$$

- Marketing (communication) and social value
- Computer science: emphasis on interaction with a user
  - EE - emphasis on mathematical analysis

Local Modeling Based on Data

- Data mining in the loop
- Honeywell Prague product



Multidimensional
Data Cube

Relational
Database

Heat
demand

Time
of day

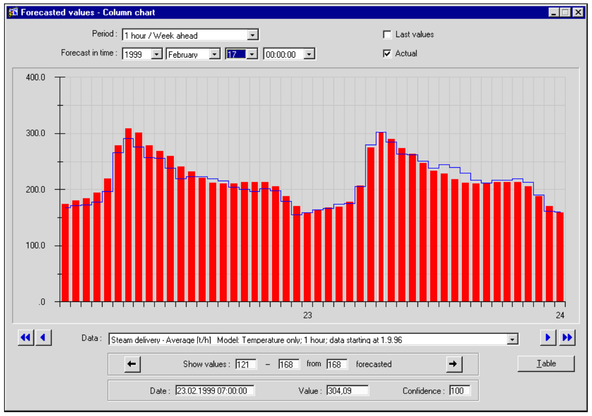
Outdoor
temperature

Query point
(What if ?)

Forecasted
variable

Explanatory
variables

Heat Loads



Forecasted values - Column chart

Period : 1 hour / Week ahead

Forecast in time : 1999 February 17 00:00:00

☐ Last values

☑ Actual

Data : Steam delivery - Average [t/h]   Model: Temperature only; 1 hour; data starting at 1.9.96

Show values : 121 – 168 from 168 forecasted

Date : 23.02.1999 07:00:00

Value : 304,09

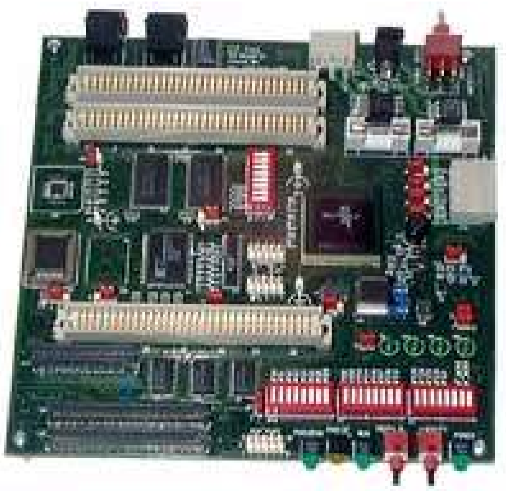
Confidence : 100

System platform for control computing

- Workstations
  - advanced process control
  - enterprise optimizers
  - computing servers (QoS/admission control)
- Specialized controllers:
  - PLC, DCS, motion controllers, hybrid controllers

System platform for control computing

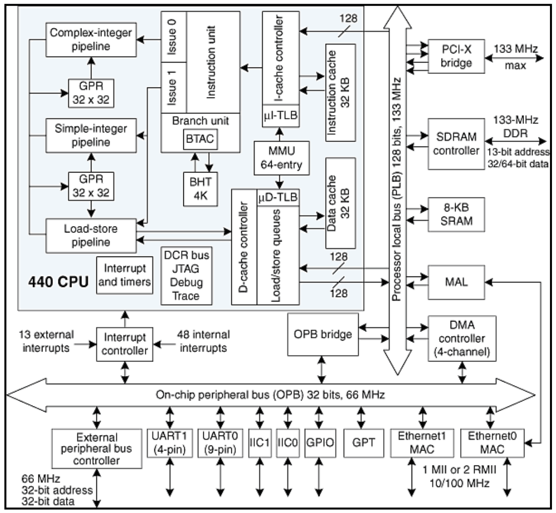
- Embedded: µP + software
- DSP



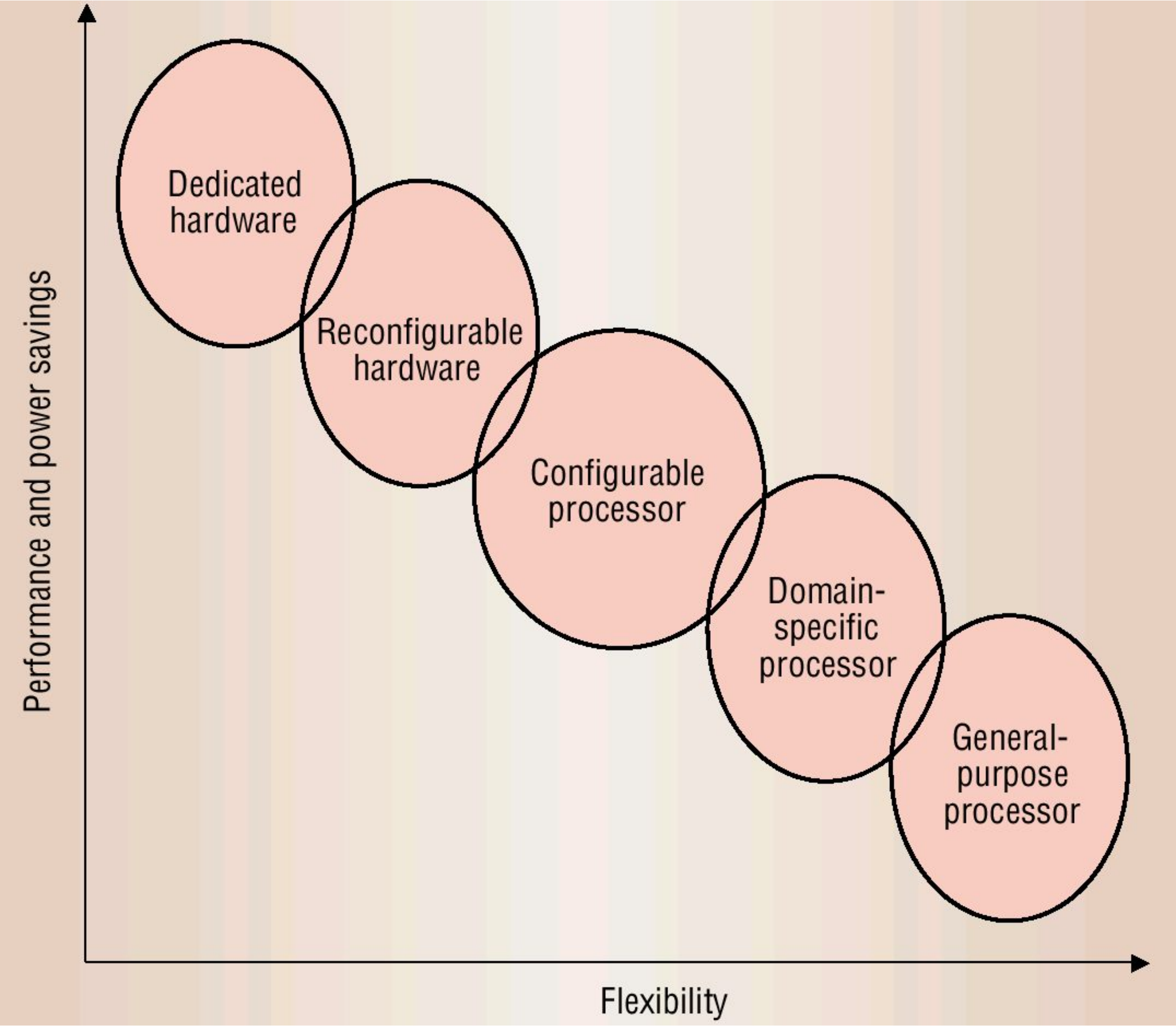
MPC555

- FPGA

- ASIC / SoC

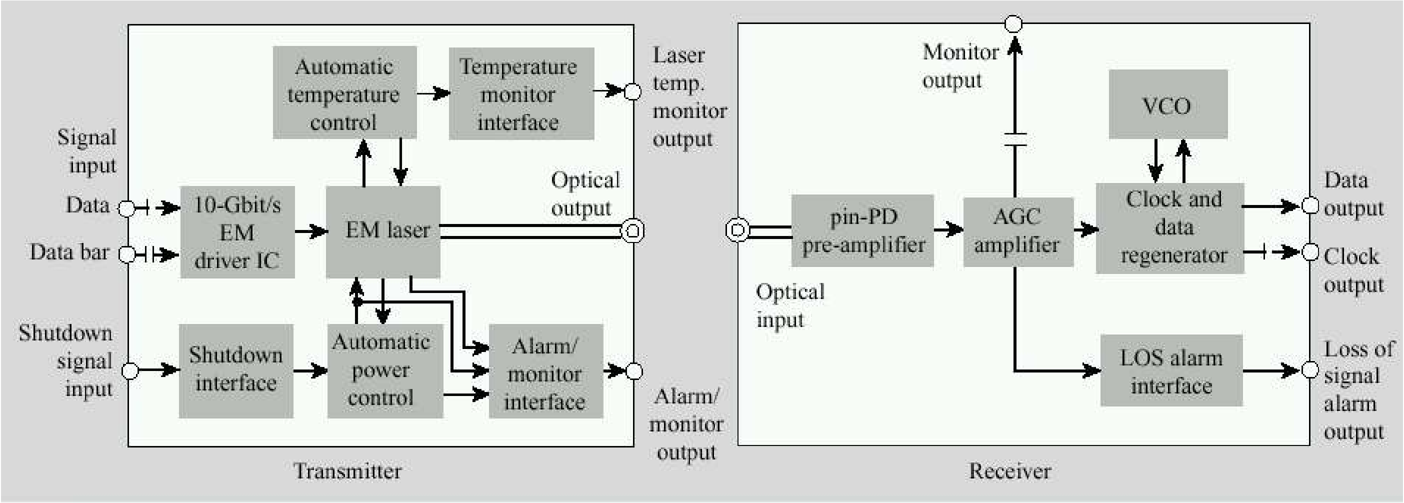


Embedded processor range



System platform, cont’d

- Analog/mixed electric circuits
  - power controllers
  - RF circuits
- Analog/mixed other
  - Gbs optical networks



EM = Electr-opt Modulator

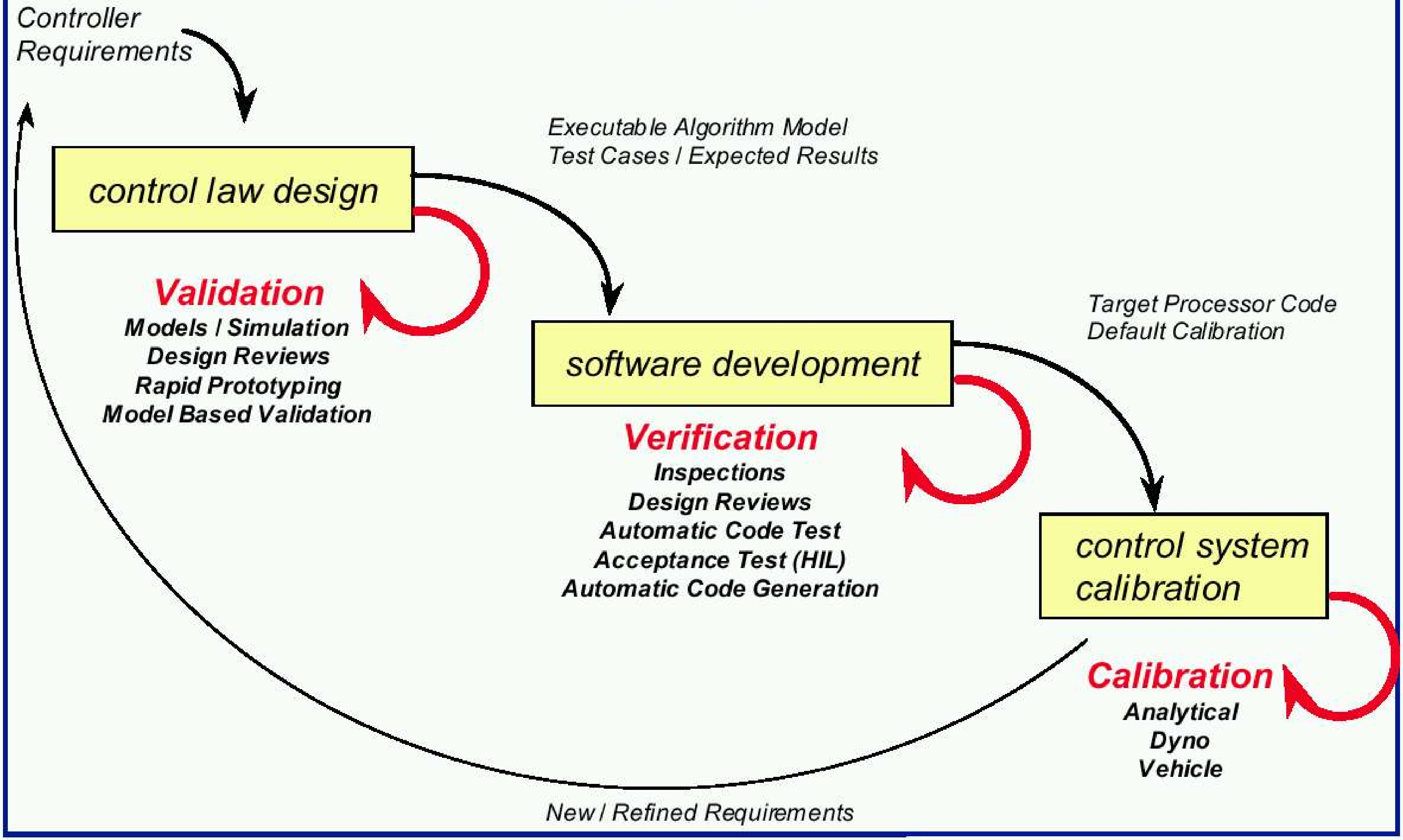
Functional Block Diagram of 10-Gbit/s Optical Transmitter/Receiver.

AGC = Auto Gain Control

Control Software

- Algorithms
- Validation and Verification

The Process



Controller Requirements

control law design

Executable Algorithm Model
Test Cases / Expected Results

Validation
Models / Simulation
Design Reviews
Rapid Prototyping
Model Based Validation

software development

Target Processor Code
Default Calibration

Verification
Inspections
Design Reviews
Automatic Code Test
Acceptance Test (HIL)
Automatic Code Generation

control system calibration

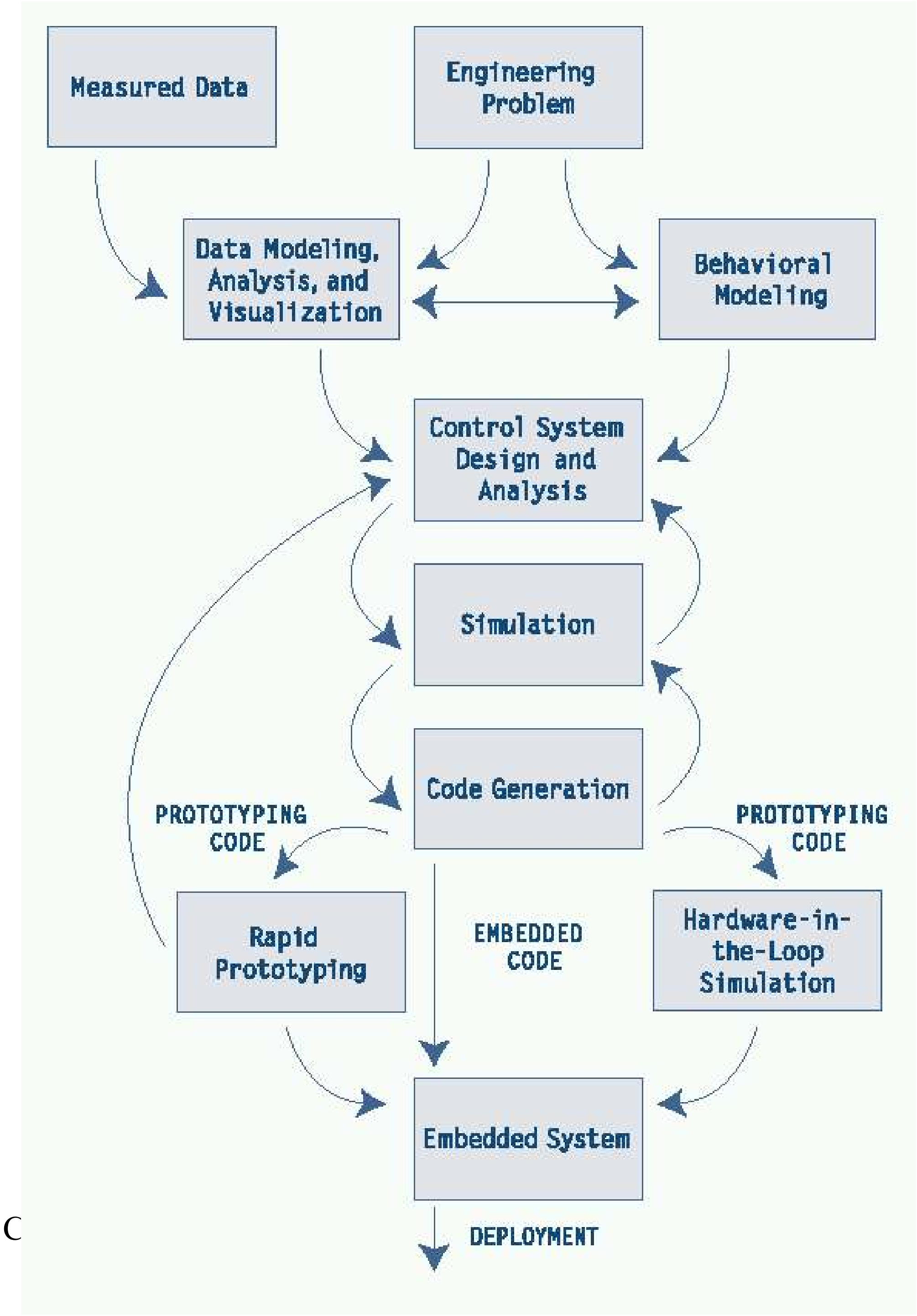
Calibration
Analytical
Dyno
Vehicle

New / Refined Requirements

Ford Motor Company

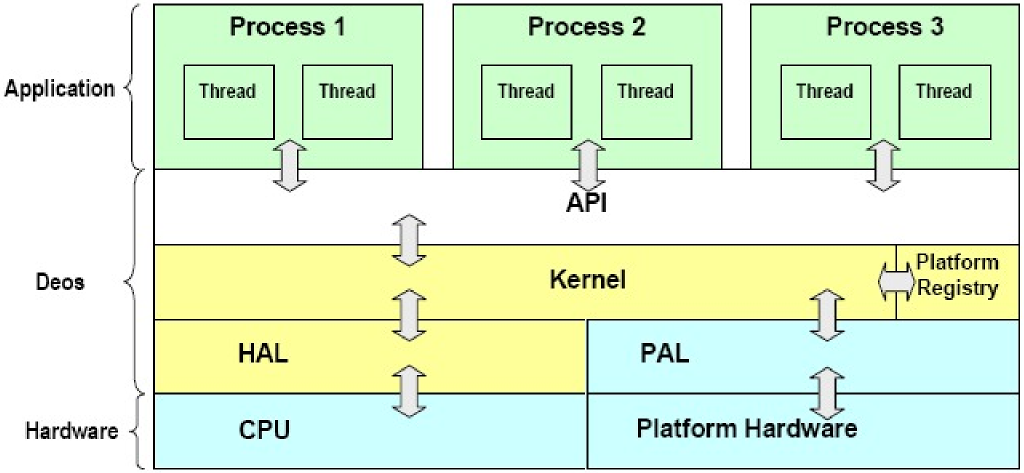
Control application software development cycle

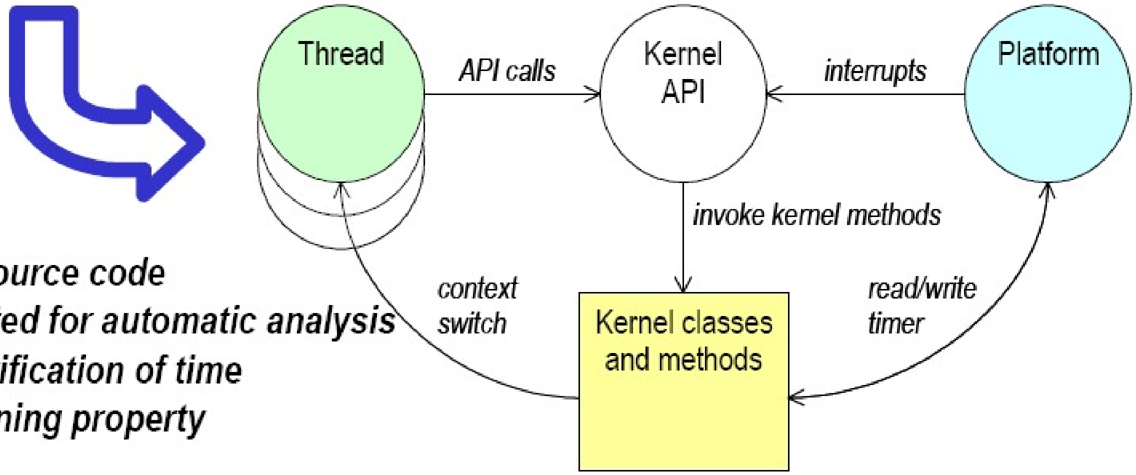
- Matlab+toolboxes
- Simulink
- Stateflow
- Real-time Workshop



Real-time Embedded Software

- Mission critical
- RT-OS with hard real-time guarantees
- C-code for each thread generated from Simulink
- Primus Epic, B787, A380

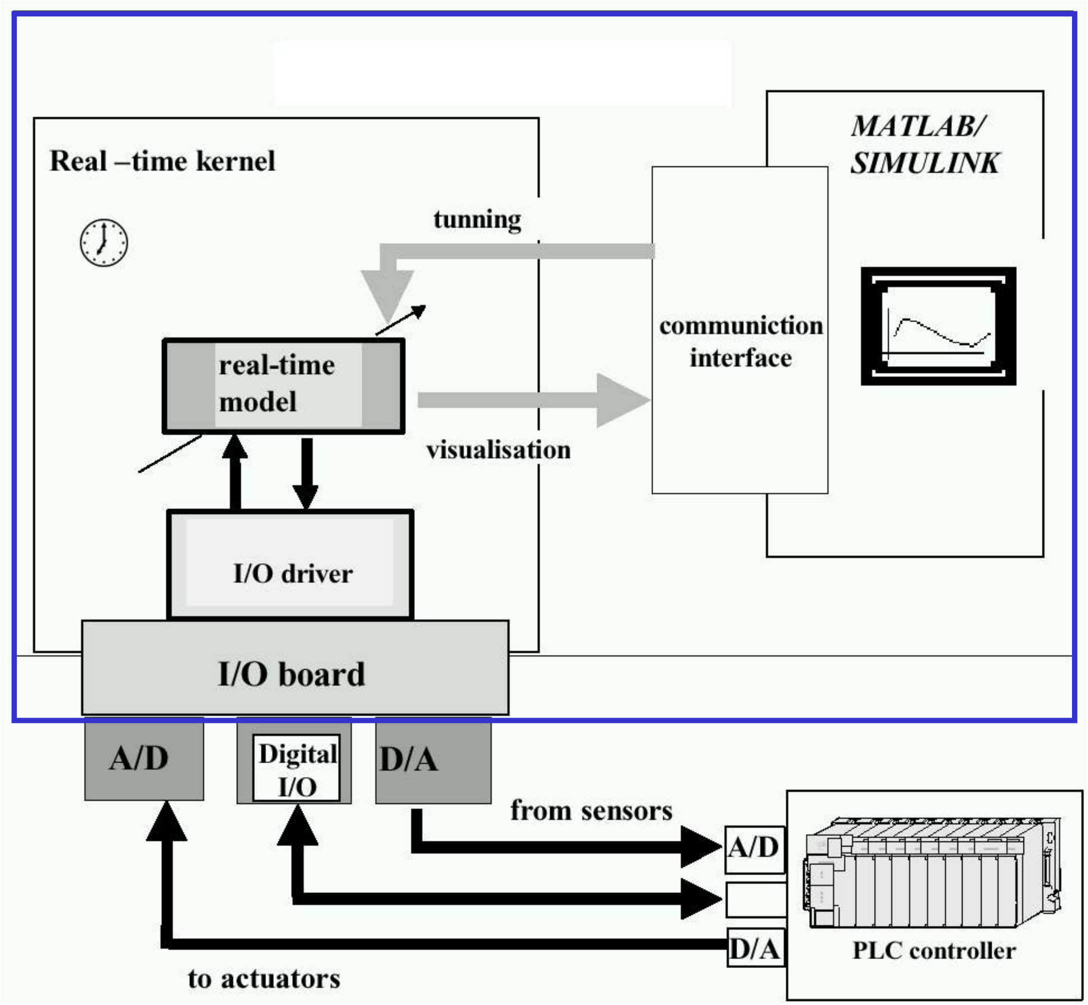


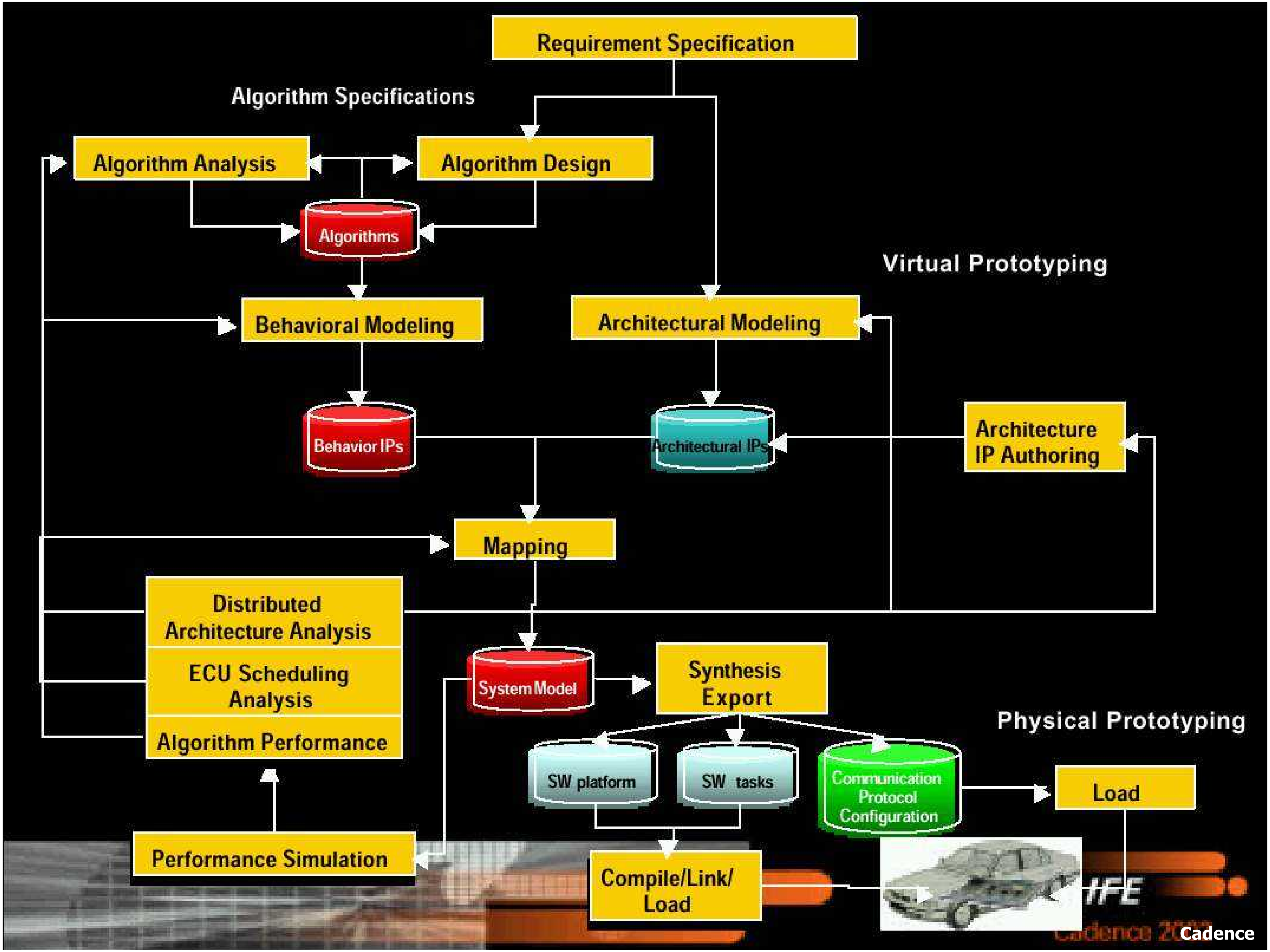


Deos source code translated for automatic analysis and verification of time partitioning property

Hardware-in-the-loop simulation

- Aerospace
- Process control
- Automotive





Requirement Specification

Algorithm Specifications

Algorithm Analysis

Algorithm Design

Algorithms

Virtual Prototyping

Behavioral Modeling

Architectural Modeling

Behavior IPs

Architectural IPs

Architecture IP Authoring

Mapping

Distributed Architecture Analysis

ECU Scheduling Analysis

Algorithm Performance

System Model

Synthesis Export

Physical Prototyping

SW platform

SW tasks

Communication Protocol Configuration

Load

Performance Simulation

Compile/Link/Load

Cadence