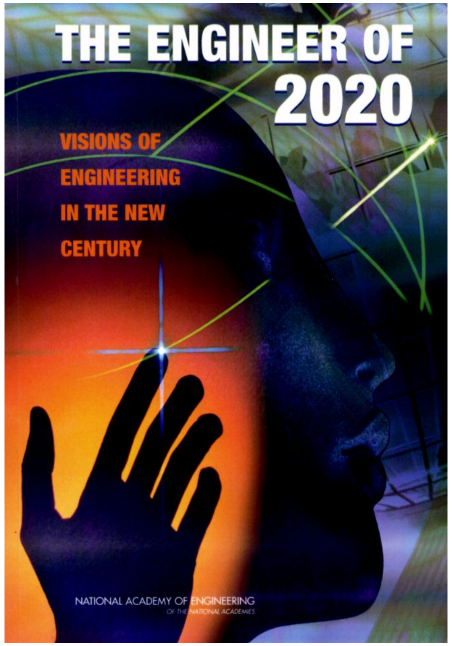
Origins of Equation-Based Modeling

Karl Johan Åström
Department of Automatic Control LTH
Lund University

Modeling is Important

There will be growth in areas of simulation and modeling around the creation of new engineering “structures”. Computer-based design-build engineering ... will become the norm for most product designs, accelerating the creation of complex structures for which multiple subsystems combine to form a final product.



NAE The Engineer of 2020

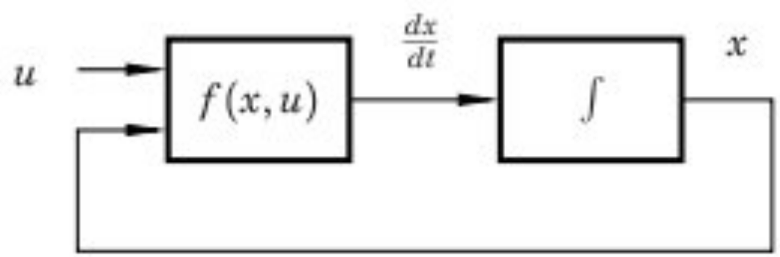
1. Introduction
2. **Block diagram modeling**
3. Equation-based modeling
4. Summary

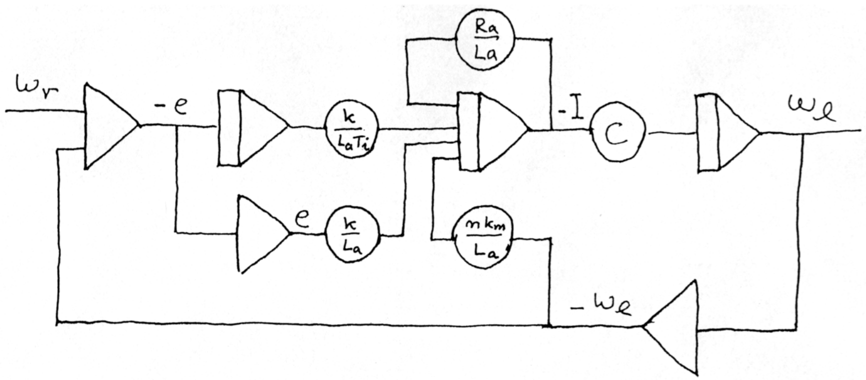
Vannevar Bush 1927

Engineering can progress no faster than the mathematical analysis on which it is based. Formal mathematics is frequently inadequate for numerous problems, a mechanical solution offers the most promise.

Analog Computing

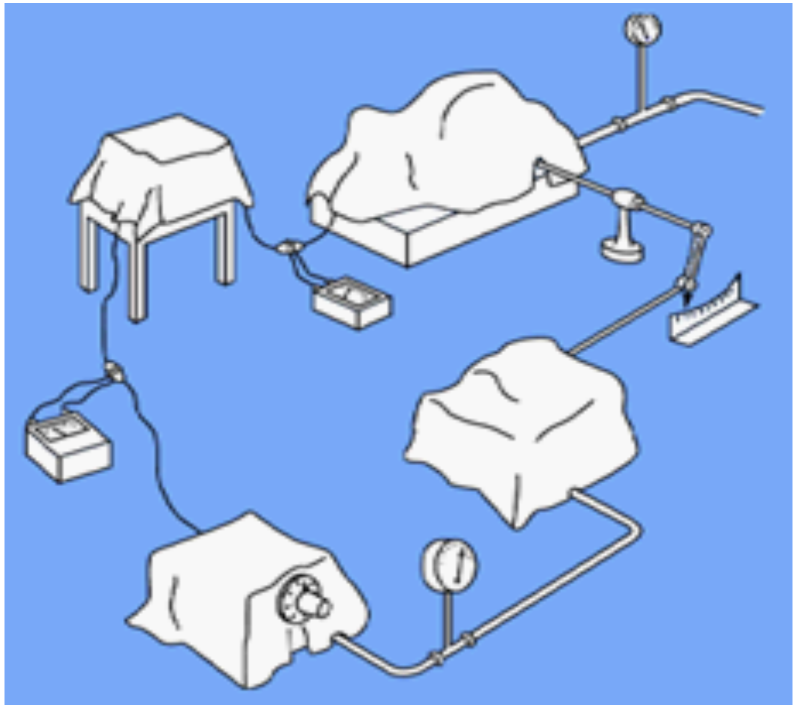
- Use a feedback loop to solve ODEs
- Integrators and function generation
- Linear systems integrators, +, -, *
- Parallelism
- Algebraic loop (loop without integrator)
- Scaling and alarms for out of scale!!





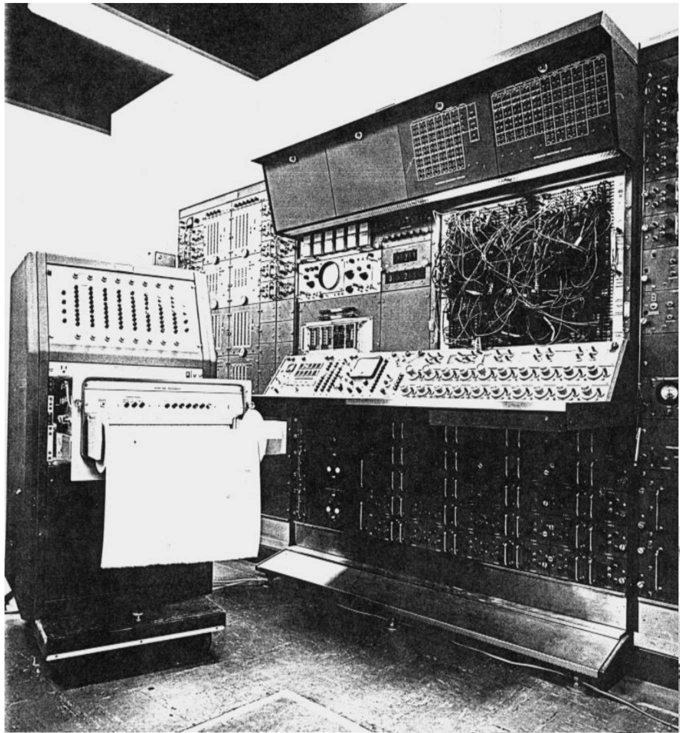
Block Diagram Modeling

- Information hiding
- Very useful abstraction
- Essential for control
- Causal inputs-output models
- Blocks described by ODE
- Base for analog computing
- BUT not for serious physical modeling

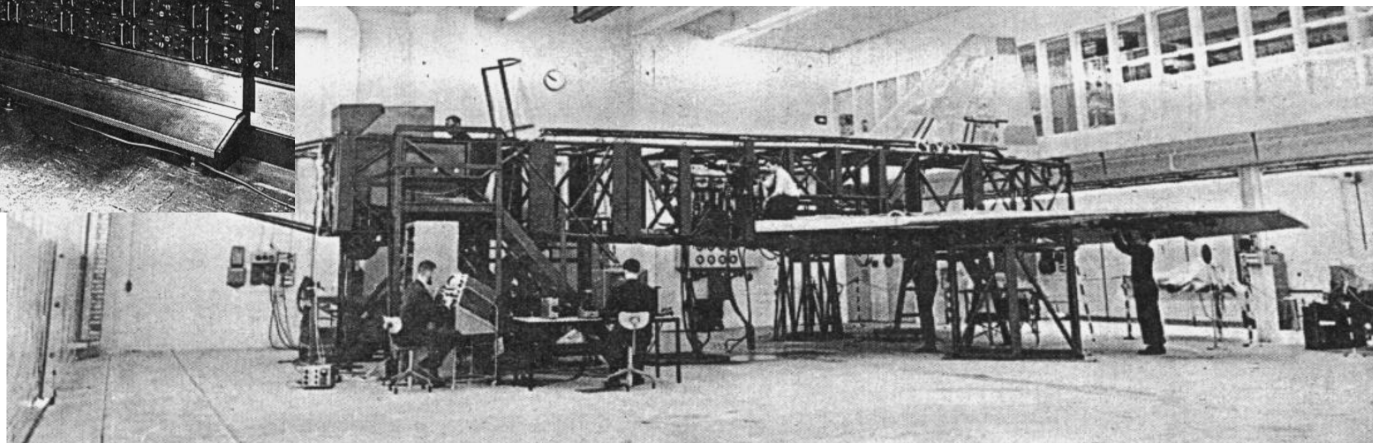


Oppelt 1954

Analog Simulation - HIL



- Ordinary differential equations $dx/dt=f(x,p)$
- Scaling, patching
- Set initial conditions and parameters
- Direct manipulation of parameters
- Manifestation of algebraic loops
- Print results
- Hardware in the loop simulation
- Simulation centers



Digital Emulators

- Precompilers to FORTRAN
- MIMIC Wright-Patterson 1965
- CSMP IBM 1962
- Babels tower > 30 emulators by 1965
- CSSL Simulation Council 1967
- ACSL Gauthier and Mitchell 1975
- SIMNON Elmqvist 1975
- MATLAB Cleve Moler 1980
- System Build, MatrixX 1984
- LabView 1986
- PC Matlab 1984, Simulink 1991

LTH in the 70s

- New control department at LTH (1965) in new school (1961) close to an old university
- Research program in Control Department: Optimization, Computer Control, System Identification, Adaptive Control, Applications:, Computer Aided Control Engineering (CACE)
- Embedded systems taught in the control department from 1970
- Interactive computing Wieslander: INTRAC, SYNPAC, IDPAC, MODPAC. FORTRAN based widely distributed
- A nonlinear simulator was missing

Simnon Elmqvist 1972

A block diagram language and an interactive simulator
Formal syntax in Bachus Naur format
Six basic commands: SYST, PAR, INIT SIMU, PLOT, AXES
Seven auxiliary: STORE, SHOW, DISP, SPLIT, HCOPY, ALGOR, ERROR

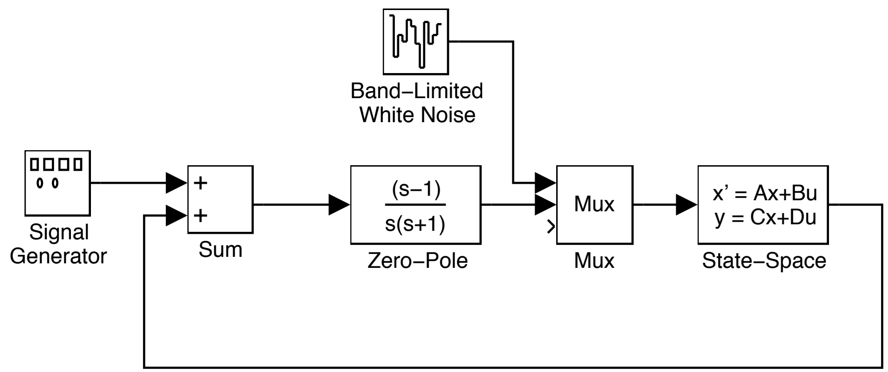
```
CONTINUOUS SYSTEM proc
Input u
Output y
State x
Der dx
dx=sat(u,0.1)
END


CONNECTING SYSTEM
yr(reg)=1; y(reg)=y(proc)
u(proc)=u(reg)
END
```

```
DISCRETE SYSTEM reg
Input yr y
Output u
State I
New nI
Tsamp ts
ts=t+h
v=k*e+I
u=sat(v.0.1)
nI=I+k*h*e/Ti+u-v
k:1
h:0.1
END
```

Simulink 1991 the Ultimate Block Diagram Tool

- Mimics the analog computer with more general blocks
- Each block a state model



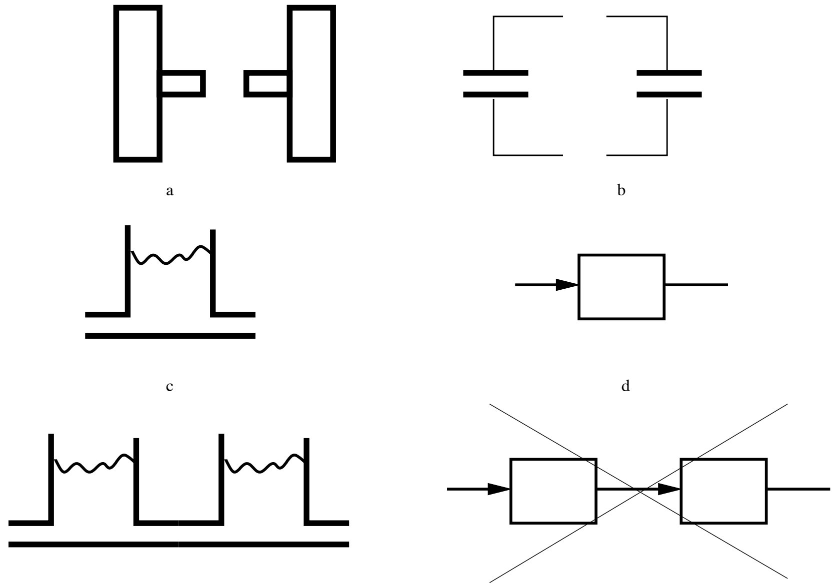
- MATLAB, Stateflow
- Granularity and Structuring
- Graphical aggregation and disaggregation
- Much manual manipulation from physics to blocks
- Neither formal syntax nor formal semantics

But!!

States may disappear when system are interconnected – warning algebraic loop!

Composition does not work!

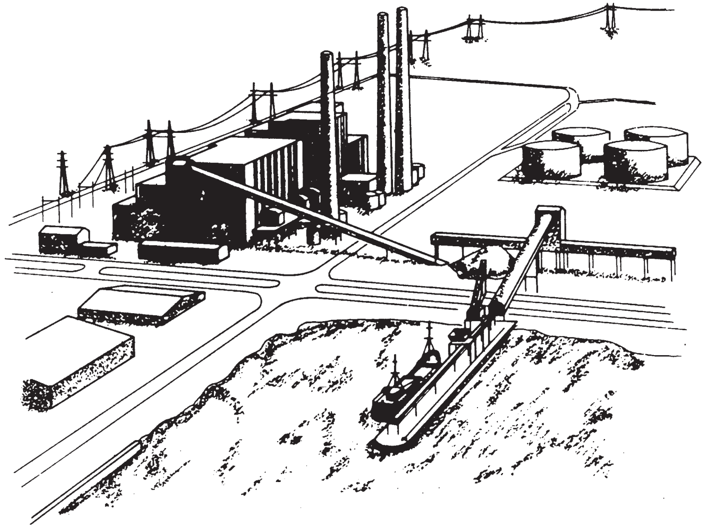
Much manual labor to go from physics to block diagrams

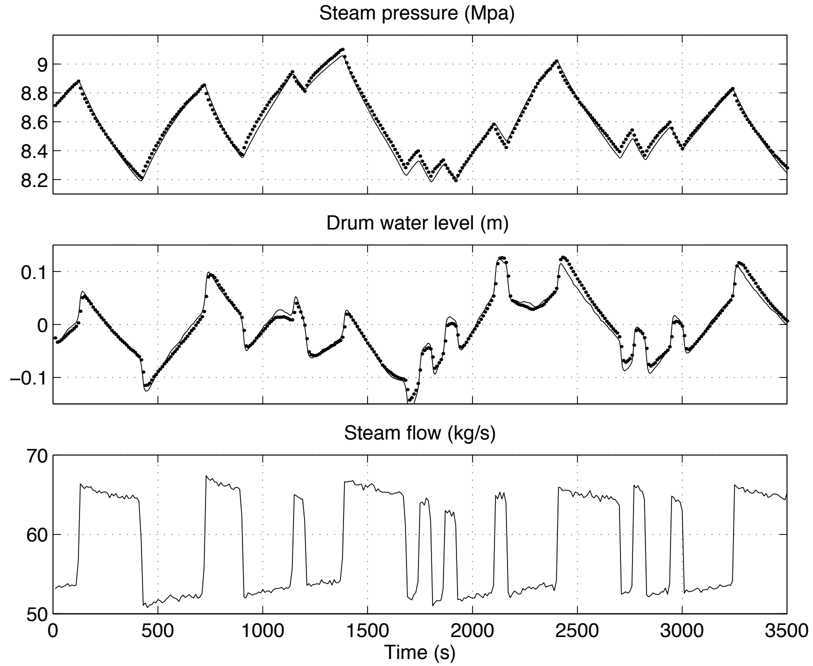


Lesson 1: Block diagrams not suitable for physical modeling
Lesson 2: Don't stick to a paradigm based on old technology when new technology emerges!!

1. Introduction
2. Block diagram modeling
3. **Equation-based modeling**
4. Summary

Boiler Control at LTH





- ➢ Experiments, modeling, system identification
- ➢ Eklund Linear DrumBoiler-Turbine Models 1971
- ➢ Lindahl Design and Simulation of a Coordinated Drum Boiler-Turbine Controller Dec 1976

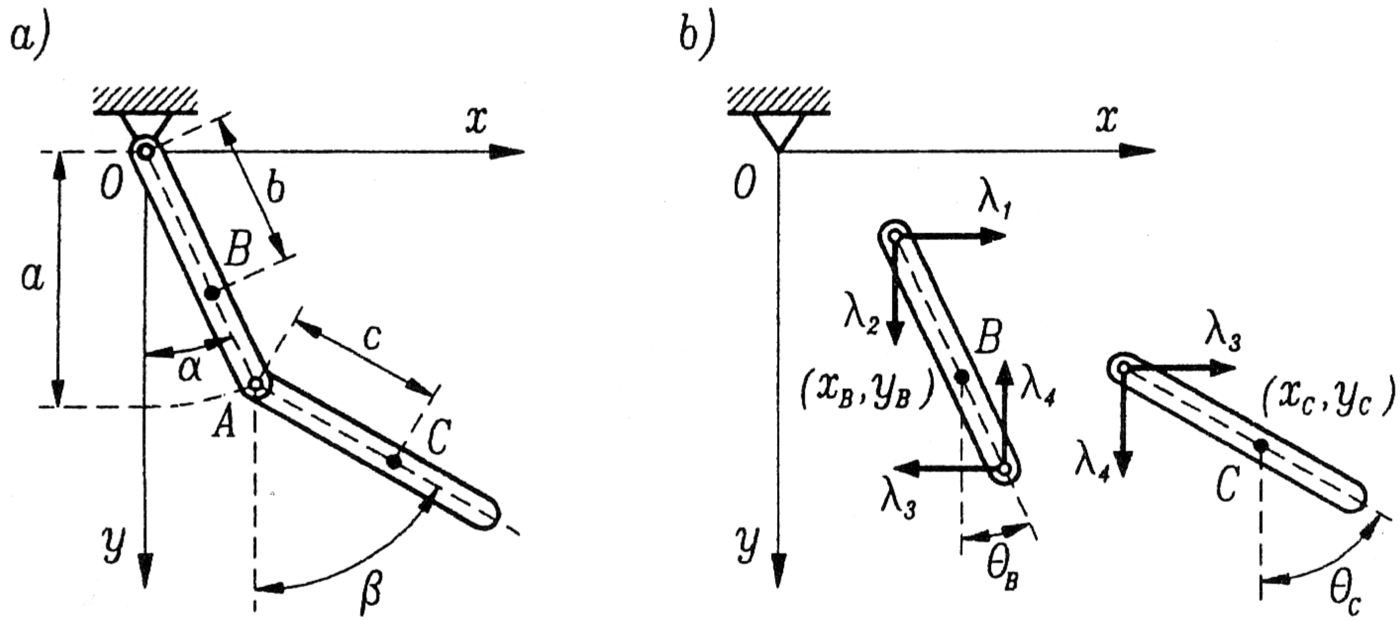
Inspiration

- Bond Graphs Henry Paynter MIT 1961
  Excellent if there is one dominating balance equation. Difficult to deal with many balances.
- Circuit theory
  Two ports systems: Kirchoffs current and voltage law
  Differential algebraic systems DAE Gear 1971 & Petzold
  Spice Peterson Berkeley 1973
  Good solution for circuits. Attempts at generalizations: System dynamics, through and across variables
- Multi-body systems: Adams, SolidWorks, ….
- Chemical Engineering: Complex plants, no dynamics, optimization

Good Old Physical Modeling

- Divide a system into subsystems
- Define interfaces and account for interactions
- Write mass, momentum and energy balances
- Add constitutive material equations
- Lumped parameters models DAE not ODE
- Symbolic computations DAE
- Connecting subsystems (many trivial equations)

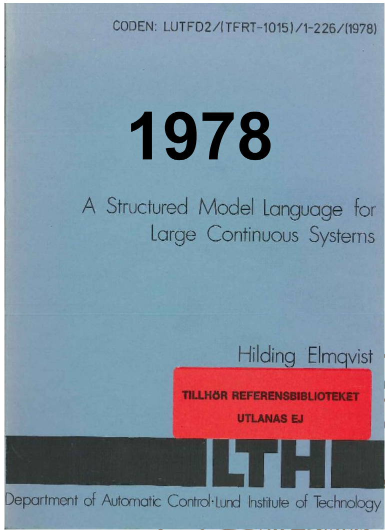
Mechanical Systems



- Split into subsystems (free body diagrams)
- Write equations of motion for each subsystem
- Add constraints to describe connections

Elmqvist's PhD Thesis

- Strong industrial interest in SIMNON, demands for extensions, matrices, hierarchies. Is this a good thesis topic? Transpiration/inspiration?
- More interesting to make a modeling language
- Modeling paradigm – balance equations
- Object orientation (Simula)
- Symbolic computations DAE
- Boiler model worked
- Great ideas but premature
- Demanding application useful



www.control.lth.se/Publication/elm78dis.html

Model Manipulations

- Eliminate redundant variables
- Use graph algorithms to reduce to lower block diagonal form LBD
- Solve linear blocks analytically
- Use tearing to generate iterative solution for nonlinear blocks
- Generate code for finding equilibria
- Generate code for DAE solvers
- Connect to optimizers
- Generate inverse models for feedforward control (reverse causality) e.g. computed torque
- Generate linear models for control design

Omola-Omsim

- Work on CACE stopped around 1980 because of FORTRAN and MATLAB
- New research project 1990 Object Oriented Modeling and Simulation: Sven Erik Mattsson, Mats Andersson, Bernt Nilsson, Dag Bruck, Jonas Eborn, Hubertus Tummescheit, Johan Åkesson
- Experiments with OO in Lisp & KEE
- C++ for object orientation
- Language (Omola) and simulator (OmSim)
- Extensive symbolic manipulation (Mattsson)
- Jmodelica.org Optimica

Modelica

- Intensive interaction with Dynasim 1991
- ESPRIT Simulation in Europe, Lund Sept 1996
- COSY meeting Lund Sept 5-7, 1996
- European groups: 23 participants, 17 talks by groups from Dynasim Lund, ETH Zurich, INRIA Paris, DLR Munich, VTT Helsinki, Imperial College London,LTH Lund, RWTH Aachen and universities in Barcelona,, Groningen, Valencia, Wien
- Formation of the Modelica language group
- First Modelica language specification Sept 1997
- 7 Modelica compilers at 9th Modelica conf 2012

Original Language Team

Hilding Elmqvist, Dynasim AB, Lund, Sweden
Fabrice Boudaud, Gaz de France,
Jan Broenink, University of Twente, Netherlands
Dag Bruck, Dynasim AB, Lund, Sweden
Thilo Ernst, GMD-FIRST, Berlin, Germany
Peter Fritzon, Linköping University, Sweden
Alexandre Jeandel, Gas de France
Kaj Juslin, VTT, Finland
Matthias Klose, Technical University of Berlin, Germany
Sven Erik Mattsson, Lund University, Sweden
Martin Otter, DLR, Oberpfaffenhofen,Germany
Per Sahlin, BrisData, Stockholm, Sweden
Hubertus Tummescheit, DLR Cologne, Germany
Hans Vangheluwe, University of Gent, Belgium

1. Introduction
2. Block diagram modeling
3. Equation-based modeling
4. **Summary**

Many Views on Modeling

- Engineering: Free body diagrams, circuit diagrams, block diagrams, P&I diagrams
- Behavioral systems Willems 1981 (CSM 2007)
- Physics: Mass, energy, momentum balances constitutive material equations
- Mathematics: ODE, DAE, PDE
- Computer Science: Languages, datastructures, programming, imperative, declarative
- Block Diagram Modeling: Causal modeling, imperative
- Equation-Based Modeling: Acausal, declarative,

Equation-based Modeling

- Has come a long way
- Serious industrial use
- 9th Modelica conference, several commercial compilers
- Strong potential for education
- Lower the entrance barrier
- Many challenges
- Much work remains
- Step back and think!
- This workshop and …

Challenges

- Is it time to sit it back and think about fundamentals?
- Make Modelica an international standard, compliance checking!
- Make it widely used!
- More than simulation
- Embedded systems
- Lower entrance barrier
- The tool chain

Modeling

➢ Solomon Golomb: Mathematical models – Uses and limitations. Aeronautical Journal 1968

Solomon Wolf Golomb (1932) mathematician and engineer and a professor of electrical engineering at the University of Southern California. Best known to the general public and fans of mathematical games as the inventor of polyominoes, the inspiration for the computer game Tetris. He has specialized in problems of combinatorial analysis, number theory, coding theory and communications.

Golomb On Modeling

- Don’t apply a model until you understand the simplifying assumptions on which it is based and can test their applicability. Validity ranges
- Distinguish at all times between the model and the real world. You will never strike oil by drilling through the map!
- Don’t expect that by having named a demon you have destroyed him
- The purpose of notation and terminology should be to enhance insight and facilitate computation – not to impress or confuse the uninitiated