Foundations of Modelling and Simulation

Hans Vangheluwe

Modelling, Simulation and Design Lab (MSDL)

School of Computer Science, McGill University,
Montréal, Canada

Department of Mathematics and Computer Science,
University of Antwerp, Belgium

Hierarchy of System Specification of Structure and Behaviour

- Basis of System Specification:
  sets theory, time base, segments and trajectories
- Hierarchy of System Specification (**causal**, **deterministic**)
  1. I/O Observation Frame
  2. I/O Observation Relation
  3. I/O Function Observation
  4. I/O System
- Multicomponent Specifications
- Non-causal models

ref: Wayne Waymore, Bernard Zeigler, George Klir, ...

Set Theory

Properties:

$$\{1, 2, \ldots, 9\}$$

$$\{a, b, \ldots, z\}$$

$$\mathbb{N}, \mathbb{N}^+, \mathbb{N}^+_\infty$$

$$\mathbb{R}, \mathbb{R}^+, \mathbb{R}^+_\infty$$

$$EV = \{ARRIVAL, DEPARTURE\}$$

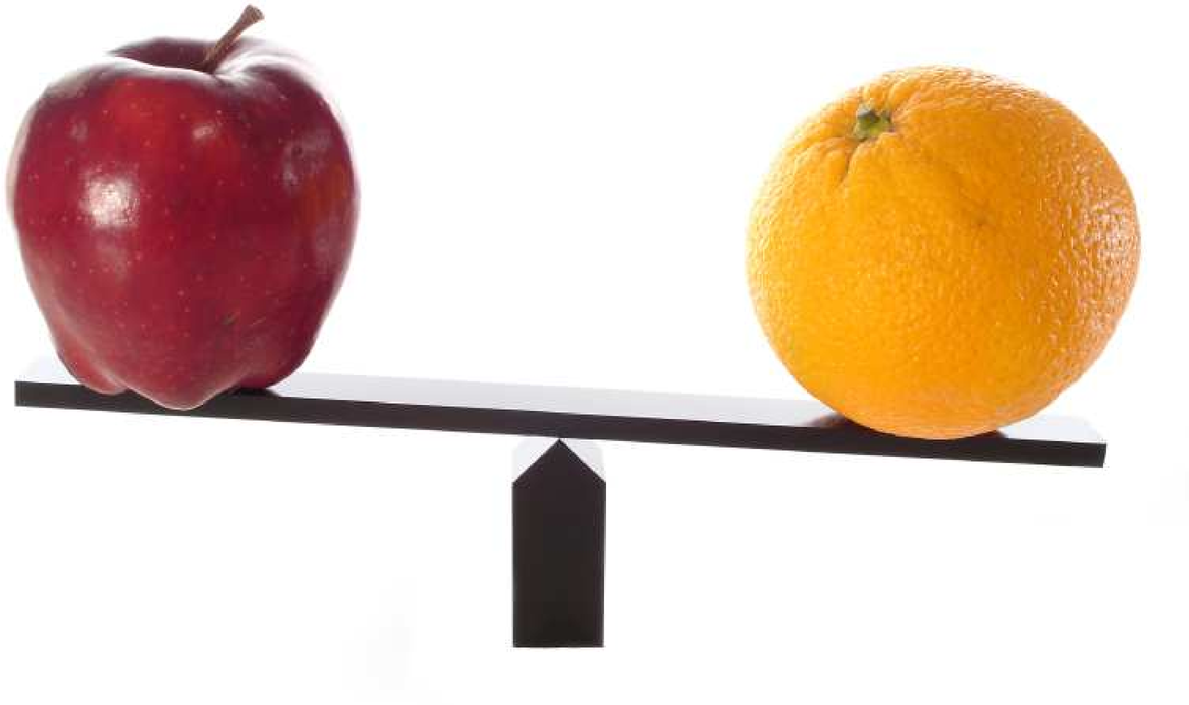
$$EV^\phi = EV \cup \{\phi\}$$

Structuring:

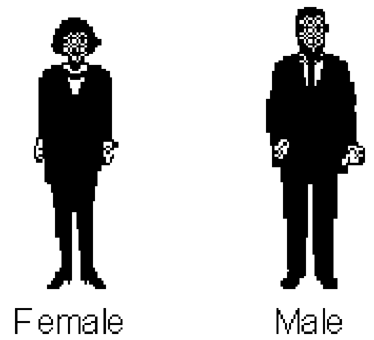
$$A \times B = \{(a, b) | a \in A, b \in B\}$$

$$G = (E, V), V \subseteq E \times E$$

Comparing things



Nominal Scale: e.g., gender

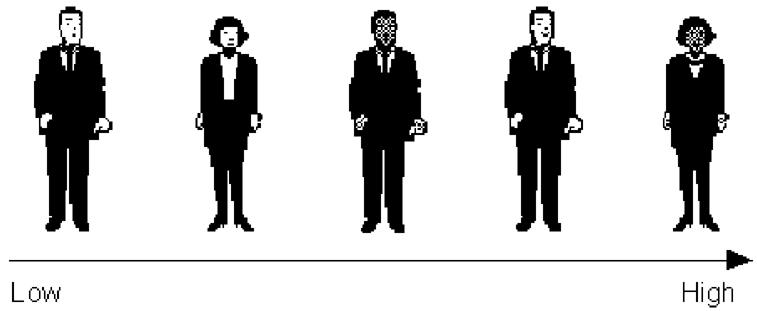


A scale that assigns a *category label* to an individual.
Establishes no explicit ordering on the category labels.

Only a notion of *equivalence* "$=$" is defined with properties:

1. Reflexivity: $x = x \vee x \neq x$.
2. Symmetry of equivalence: $x = y \Leftrightarrow y = x$.
3. Transitivity: $x = y \wedge y = z \rightarrow x = z$.

Ordinal Scale: e.g., degree of happiness



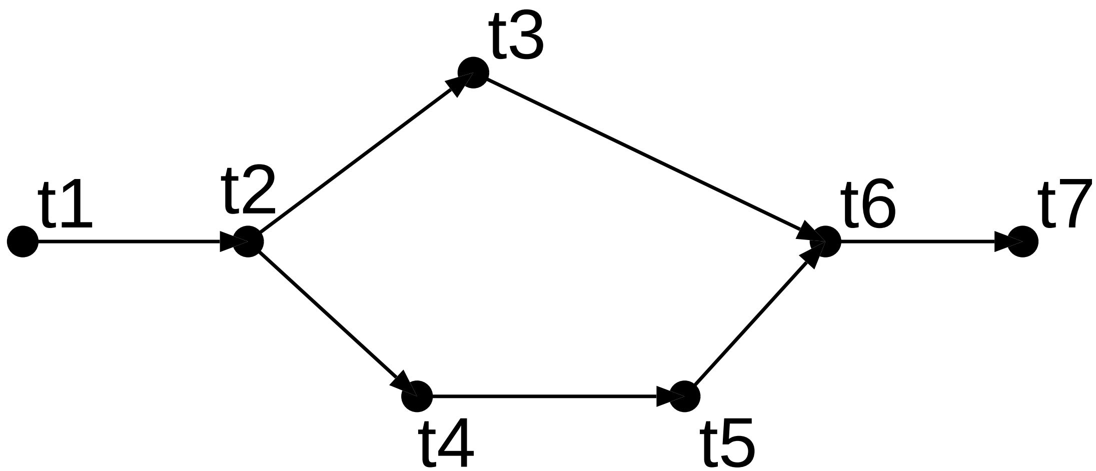
A scale in which data can be *ranked*, but in which no arithmetic transformations are meaningful. It is meaningless to talk about difference (distance).

In addition to equivalence, a notion of *order* $<$ is defined with properties:

1. Symmetry of equivalence: $x = y \Leftrightarrow y = x$.
2. Asymmetry of order: $x < y \rightarrow y \not< x$.
3. Irreflexivity: $x \not< x$.
4. Transitivity: $x < y \wedge y < z \rightarrow x < z$.

Partial ordering

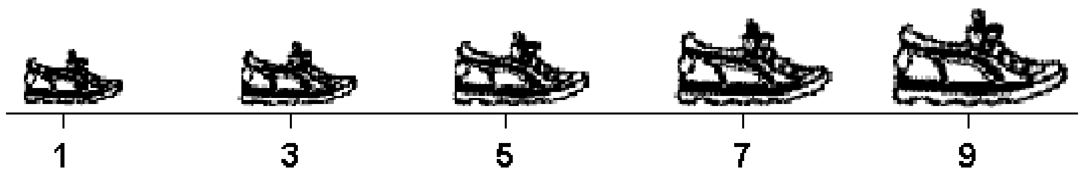
The ordering may be *partial* (some data items cannot be compared).



The ordering may be *total* (all data items can be compared).

$$\forall x, y \in X : x < y \vee y < x \vee x = y$$

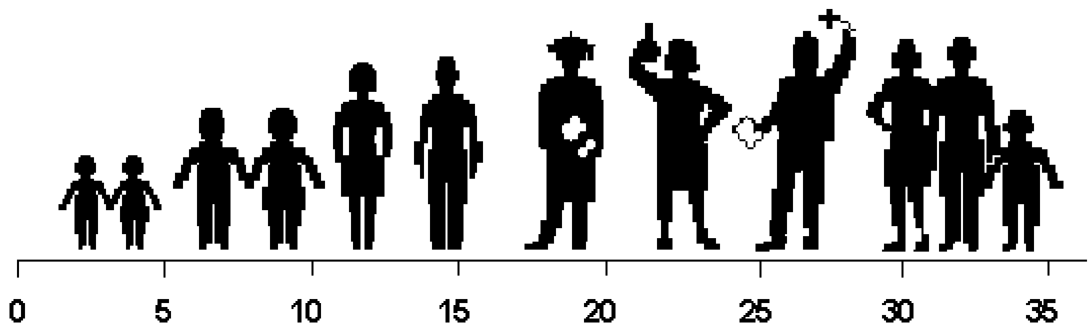
Interval Scale: e.g., Shoe Size



A scale where *distances* between data are meaningful. On interval measurement scales, one unit on the scale represents the *same magnitude* of the characteristic being measured across the whole range of the scale. Interval scales do not have a “true” zero point, however, and therefore it is not possible to make statements about how many times higher one value is than another.

In addition to equivalence and order, a notion of *interval* is defined. The choice of a zero point is arbitrary.

Ratio Scale: e.g., age



Both *intervals* between values and *ratios* of values are meaningful. A meaningful *zero* point is known. "A is twice as old as B".

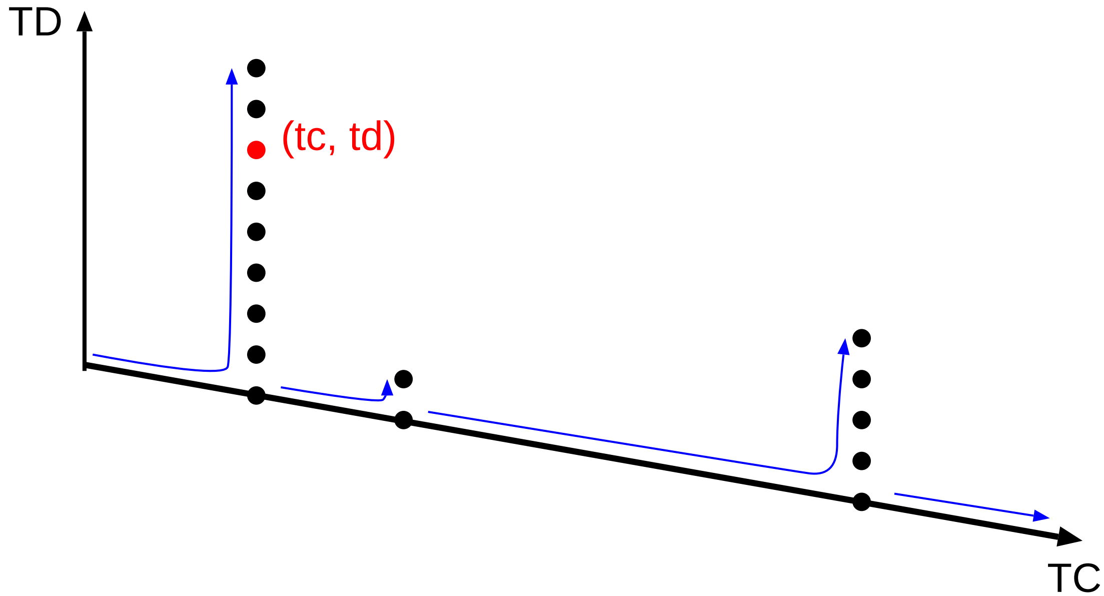
Time Base

- Simulation of **Dynamic** Systems: irreversible passage of *time*.

- Time Base T:
  - $\{NOW\}$ (instantaneous)
  - $\mathbb{R}$: *continuous-time*
  - $\mathbb{N}$ or isomorphic: *discrete-time*
- Ordering:
  - Ordinal Scale (possibly partial ordering, for concurrency)
  - Interval Scale
  - Ratio Scale

Time Bases for hybrid system models

Time Bases for hybrid system models

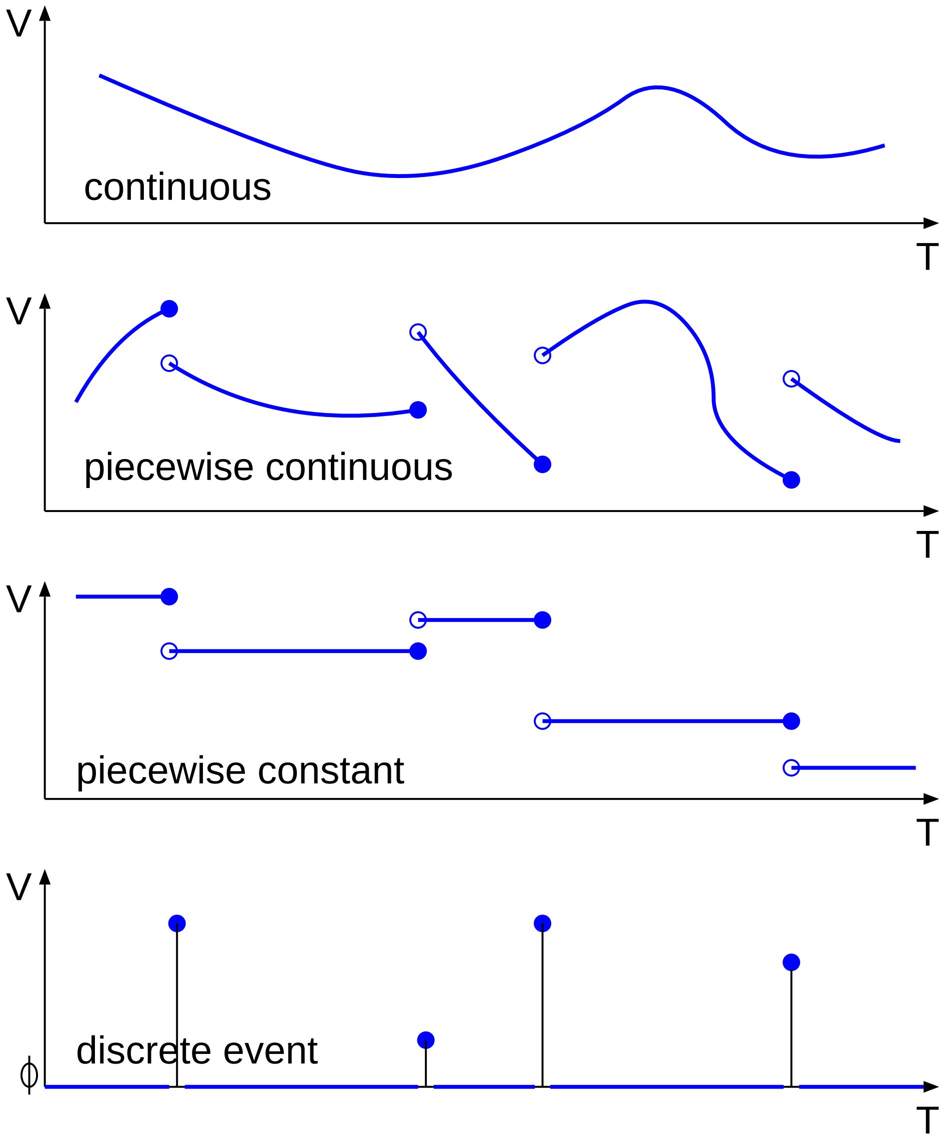


"nested time" for nested experiments.

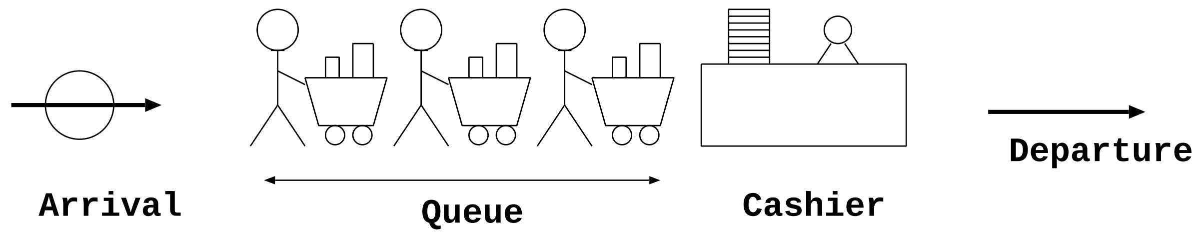
Behaviour $\equiv$ Evolution over Time

- With time base, describe *evolution over time*
- Time function, **trajectory**, signal: $f : T \to V$
- Restriction to $T' \subseteq T$
  $f|T' : T' \to V$, $\forall t \in T' : f|T'(t) = f(t)$
  - Past of f: $f|T_{t\rangle}$
  - Future of f: $f|T_{\langle t}$
- Restriction to an interval: **segment**
  $\omega : \langle t_1, t_2\rangle \to V$

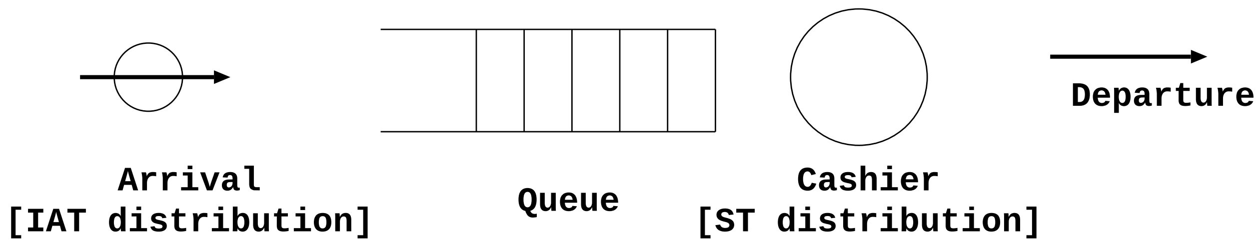
Types of Segments



Cashier-Queue System

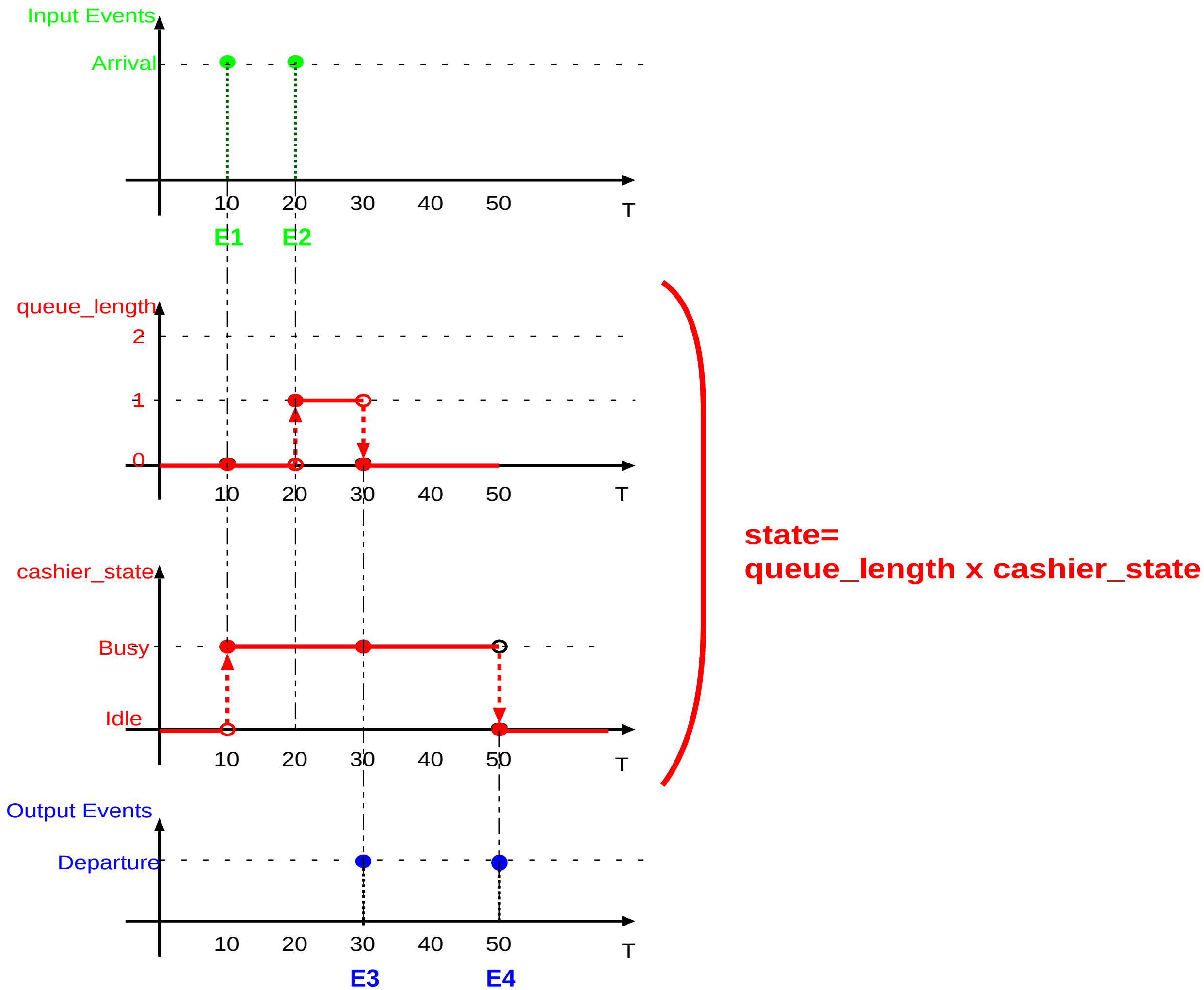


Physical View



Abstract View

Trajectories



I/O Observation Frame (causal)

$$O = \langle T, X, Y \rangle$$

- T is *time-base*: $\mathbb{N}$ (discrete-time), $\mathbb{R}$ (continuous-time)
- X input value set: $\mathbb{R}^n, EV^{\phi}$
- Y output value set: system response

I/O Relation Observation

$$IORO = \langle T, X, \Omega, Y, R \rangle$$

- $\langle T, X, Y \rangle$ is Observation Frame
- Ω is the set of all possible input segments
- R is the *I/O relation*
  $\Omega \subseteq (X, T)$, $R \subseteq \Omega \times (Y, T)$
  $(\omega, \rho) \in R \Rightarrow dom(\omega) = dom(\rho)$
- $\omega : \langle t_i, t_f \rangle \rightarrow X$: input *segment*
- $\rho : \langle t_i, t_f \rangle \rightarrow Y$: output *segment*
- note: not really necessary to observe over same time domain

I/O Function Observation

$$IOFO = \langle T, X, \Omega, Y, F \rangle$$

- $\langle T, X, \Omega, Y, R \rangle$ is a Relation Observation
- Ω is the set of all possible input segments
- F is the *set of I/O functions*
  $f \in F \Rightarrow f \subset \Omega \times (Y, T)$, where
  f is a **function** such that $dom(f(\omega)) = dom(\omega)$
- *f* = *initial state*: **unique** response to ω
- $R = \bigcup_{f \in F} f$

I/O System

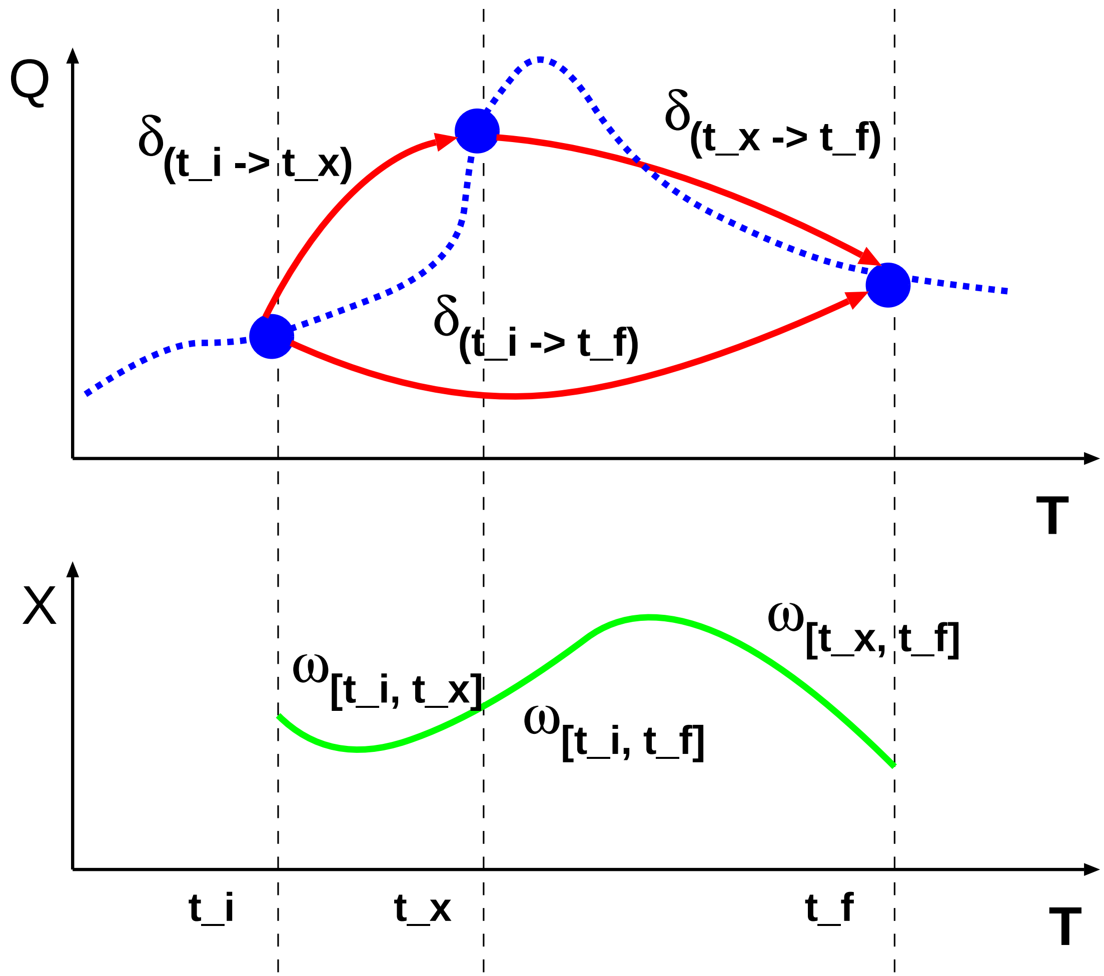
- From **Descriptive Variables** (properties) to **State**.
- **State** summarizes the past behaviour of the system.
- Future is uniquely determined by
  - **current state**
  - **future input**

$$SYS = \langle T, X, \Omega, Q, \delta, Y, \lambda \rangle$$

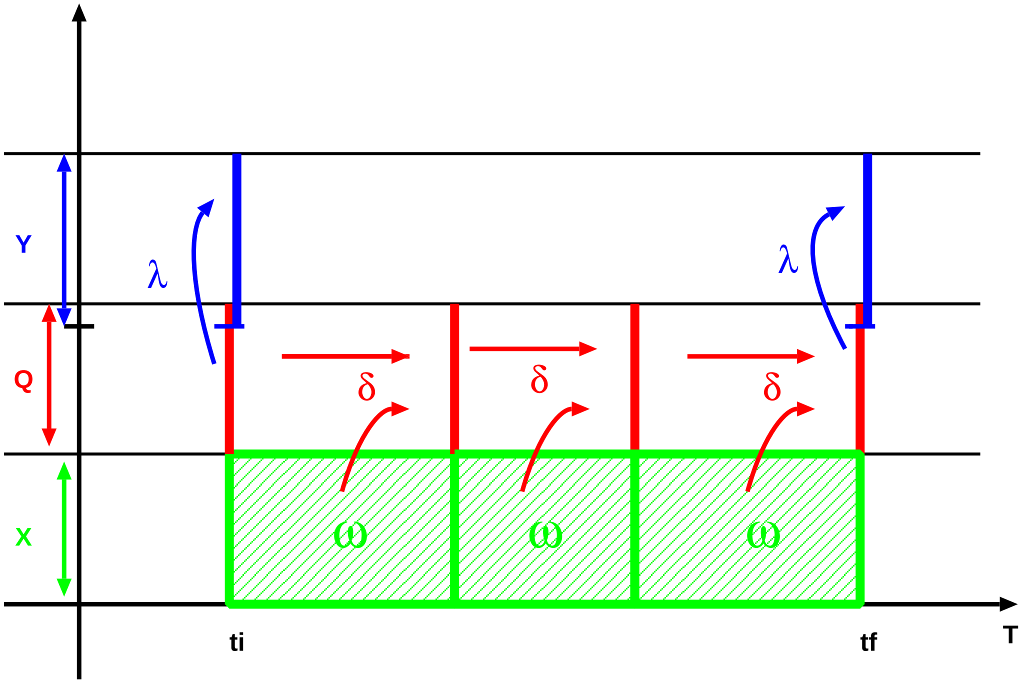
T	time base
X	input set
$\omega : T \rightarrow X$	input segment
Q	state set
$\delta : \Omega \times Q \rightarrow Q$	transition function
Y	output set
$\lambda : Q \rightarrow Y$ (or $Q \times X \rightarrow Y$)	output function

$$\forall t_x \in [t_i, t_f] : \delta(\omega_{[t_i,t_f]}, q_i) = \delta(\omega_{[t_x,t_f]}, \delta(\omega_{[t_i,t_x]}, q_i))$$

Composition Property



Simulator: step through time



Formalism classification based on general system model

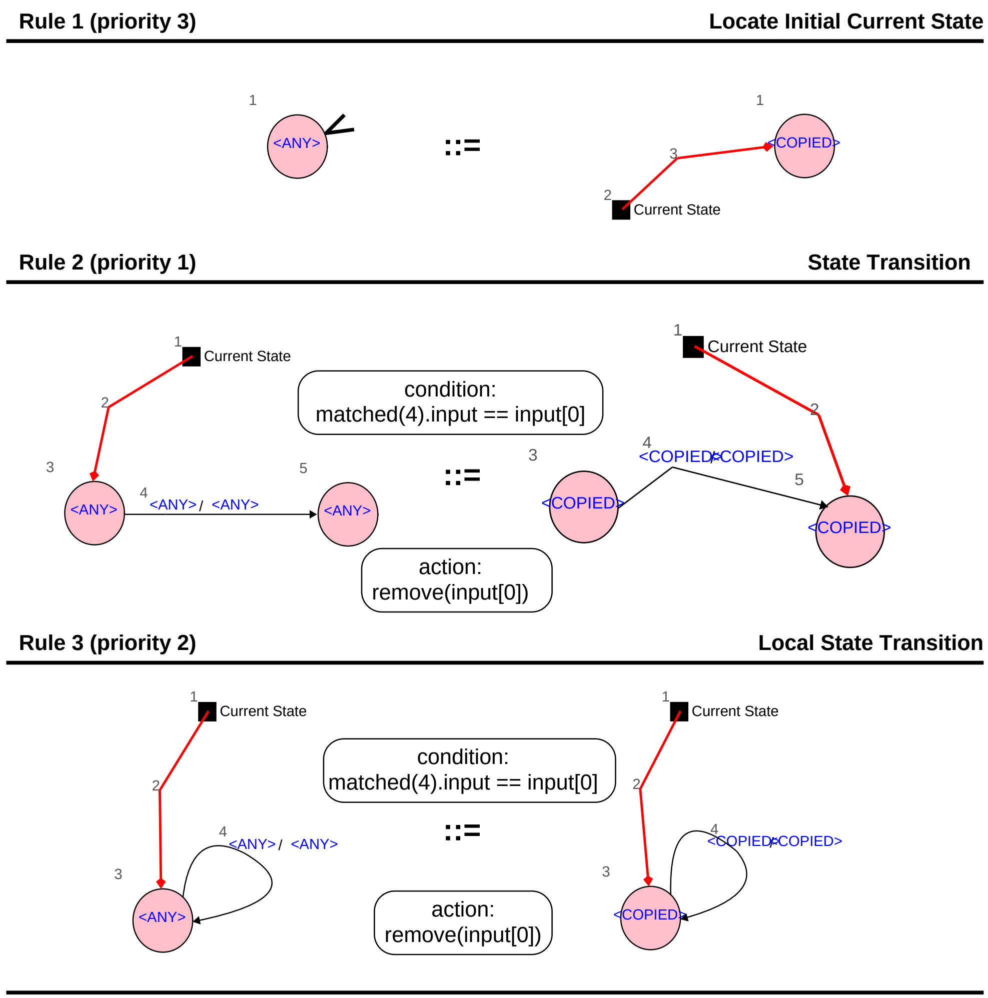
	T: **Continuous**	T: **Discrete**	T: $\{$**NOW**$\}$
Q: **Continuous**	ODE, DEVS	Difference Eqns. (DTSS)	Algebraic Eqns.
Q: **Discrete**	Discrete-event	Finite State Automata	Integer Eqns.

Basis for **general, standard software architecture of simulators**

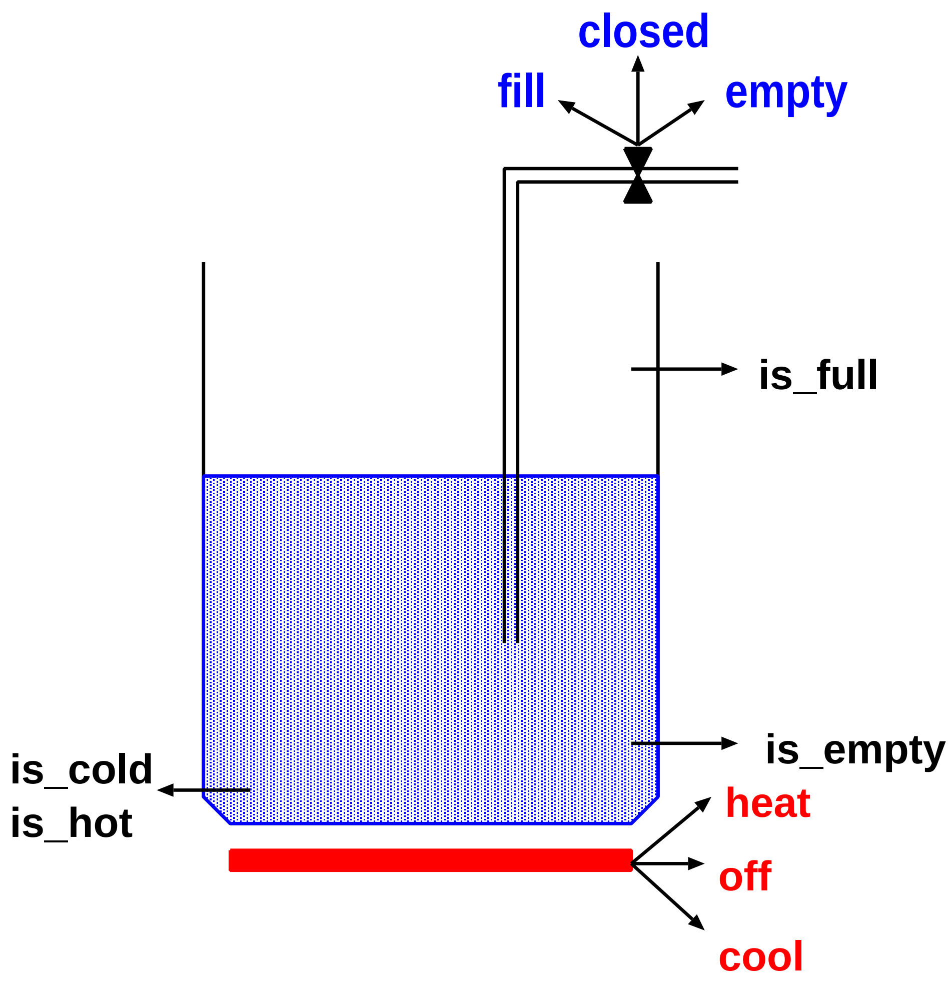
Further classifications based on **structure of formalisms**

(in particular of δ)

Rule-based specification of δ



System under study: T, h controlled liquid



Detailed (continuous) view, ALG + ODE

Inputs (discontinuous $\rightarrow$ hybrid model):

- Emptying, filling flow rate ϕ
- Rate of adding/removing heat W

Parameters:

- Temperature of influent T_{in}
- Cross-section surface of vessel A
- Specific heat of liquid c
- Density of liquid ρ

State variables:

- Temperature T
- Level of liquid l

Outputs (sensors):

- $is\_low, is\_high, is\_cold, is\_hot$

$$
\left\{
\begin{array}{rcl}
\frac{dT}{dt} & = & \frac{1}{l}\left[\frac{W}{c\rho A} - \phi(T - T_{in})\right] \\
\frac{dl}{dt} & = & \phi \\
is\_low & = & (l < l_{low}) \\
is\_high & = & (l > l_{high}) \\
is\_cold & = & (T < T_{cold}) \\
is\_hot & = & (T > T_{hot})
\end{array}
\right.
$$

$$SYS^{ODE}_{VESSEL} = \langle \mathcal{T}, X, \Omega, Q, \delta, Y, \lambda \rangle$$

$$\mathcal{T} = \mathbb{R}$$

$$X = \mathbb{R} \times \mathbb{R} = \{(W, \phi)\}$$

$$\omega : \mathcal{T} \to X$$

$$Q = \mathbb{R}^+ \times \mathbb{R}^+ = \{(T, l)\}$$

$$\delta : \Omega \times Q \to Q$$

$$\delta(\omega_{[t_i,t_f]}, (T(t_i), l(t_i))) =$$

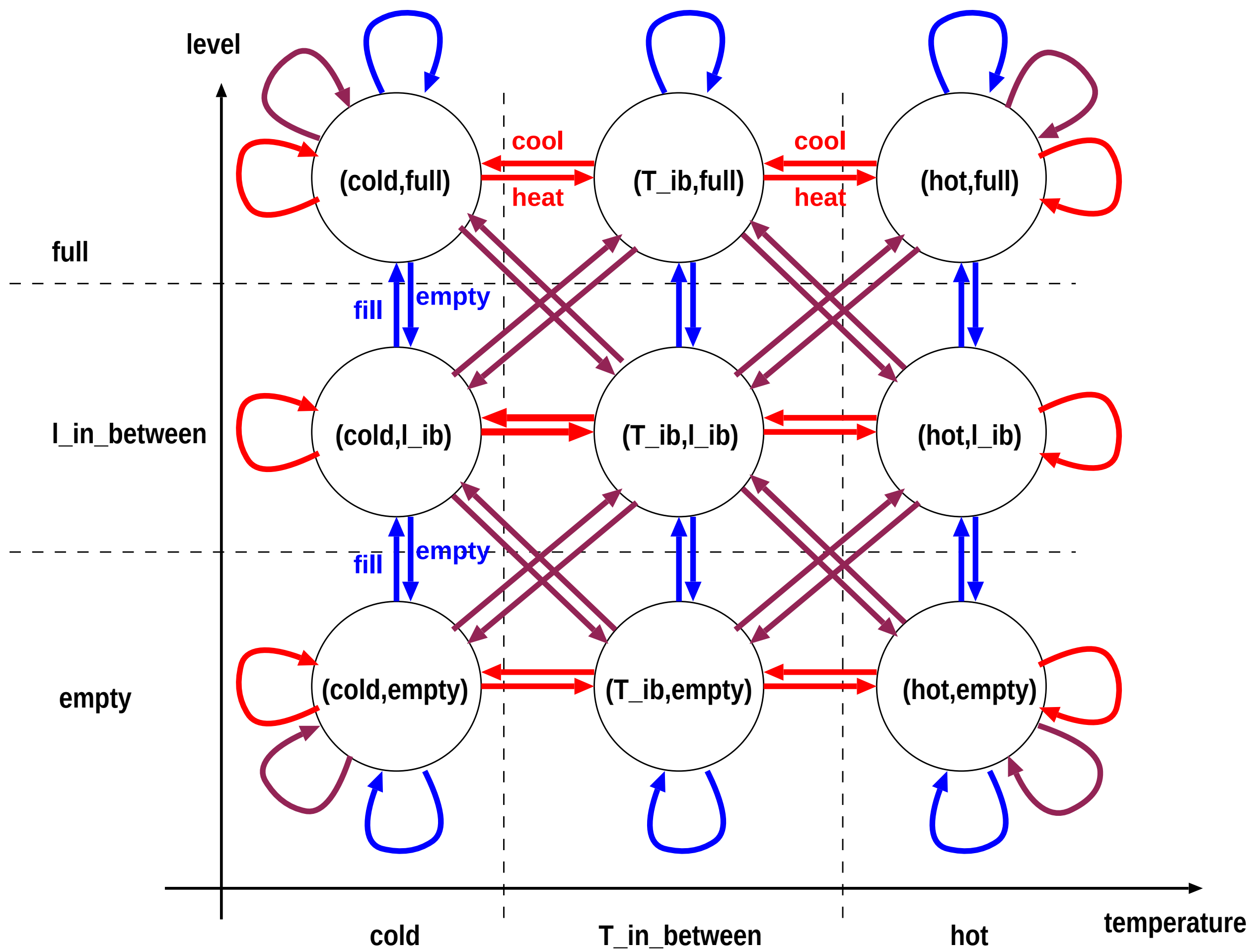
$$(T(t_i) + \int_{t_i}^{t_f} \frac{1}{l(\alpha)}[\frac{W(\alpha)}{c\rho A} - \phi(\alpha)T(\alpha)]d\alpha,\ l(t_i) + \int_{t_i}^{t_f} \phi(\alpha)d\alpha)$$

$$Y = \mathbb{B} \times \mathbb{B} \times \mathbb{B} \times \mathbb{B} = \{(is\_low, is\_high, is\_cold, is\_hot)\}$$

$$\lambda : Q \to Y$$

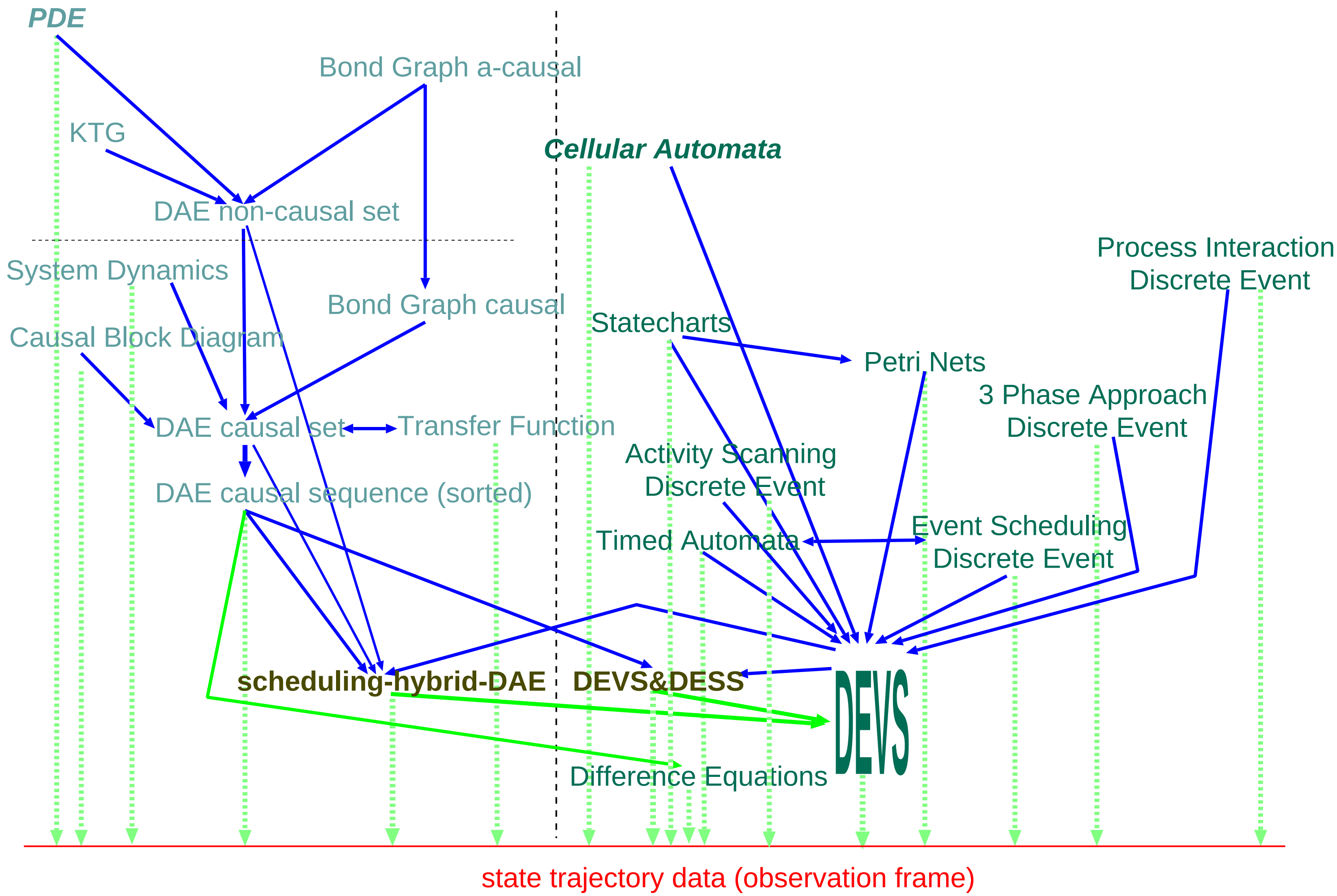
$$\lambda(T, l) = ((l < l_{low}), (l > l_{high}), (T < T_{cold}), (T > T_{hot}))$$

High-abstraction-level (discrete) view: FSA

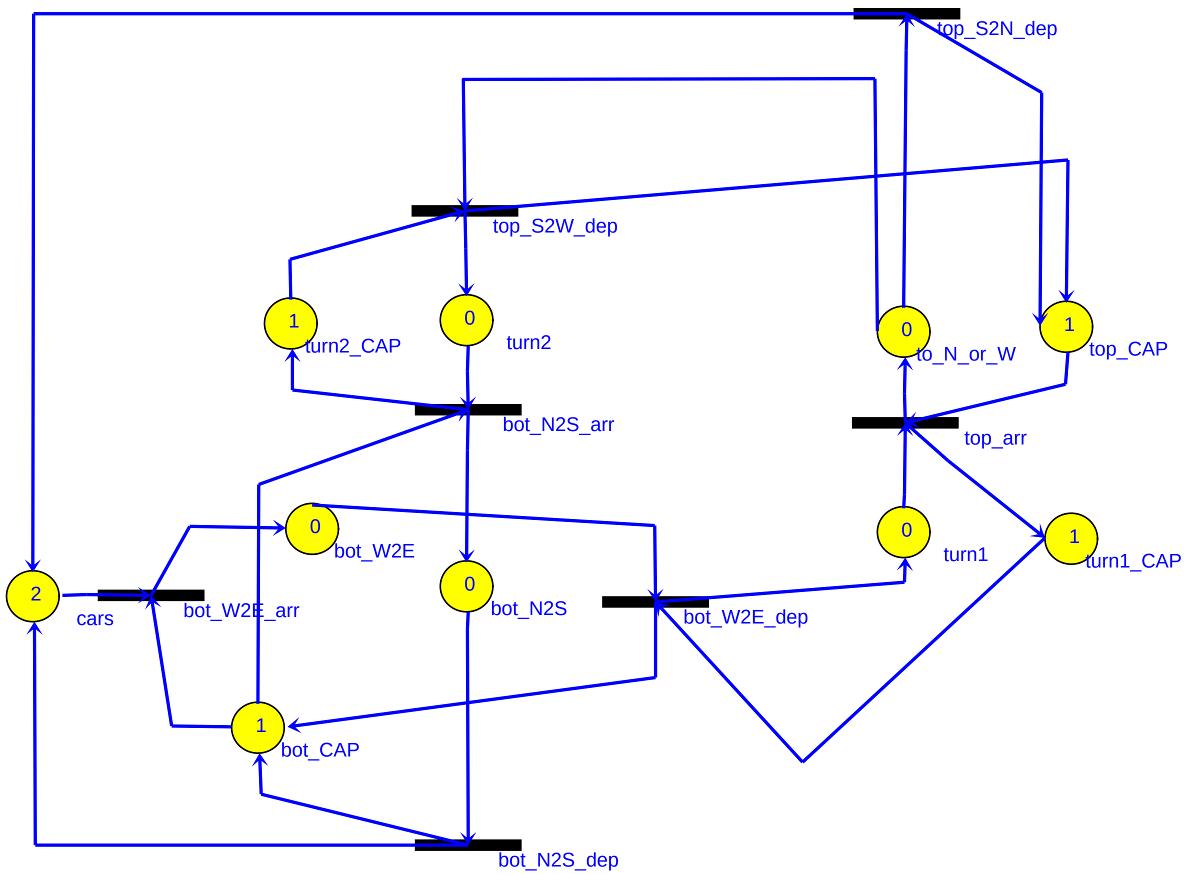


at this level: verification of properties possible

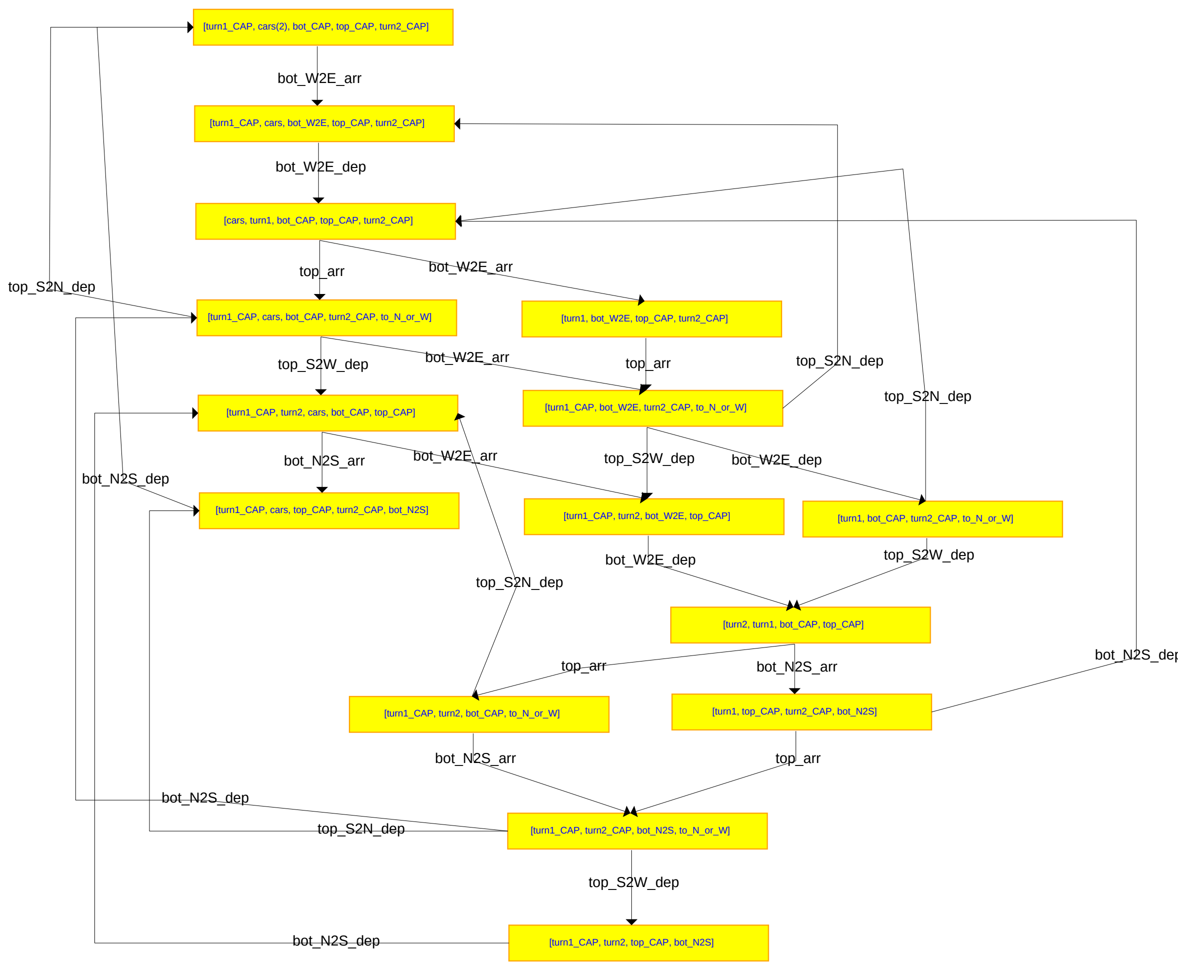
don't build simulator (Operational Semantics) but Transform (Transformational Semantics)



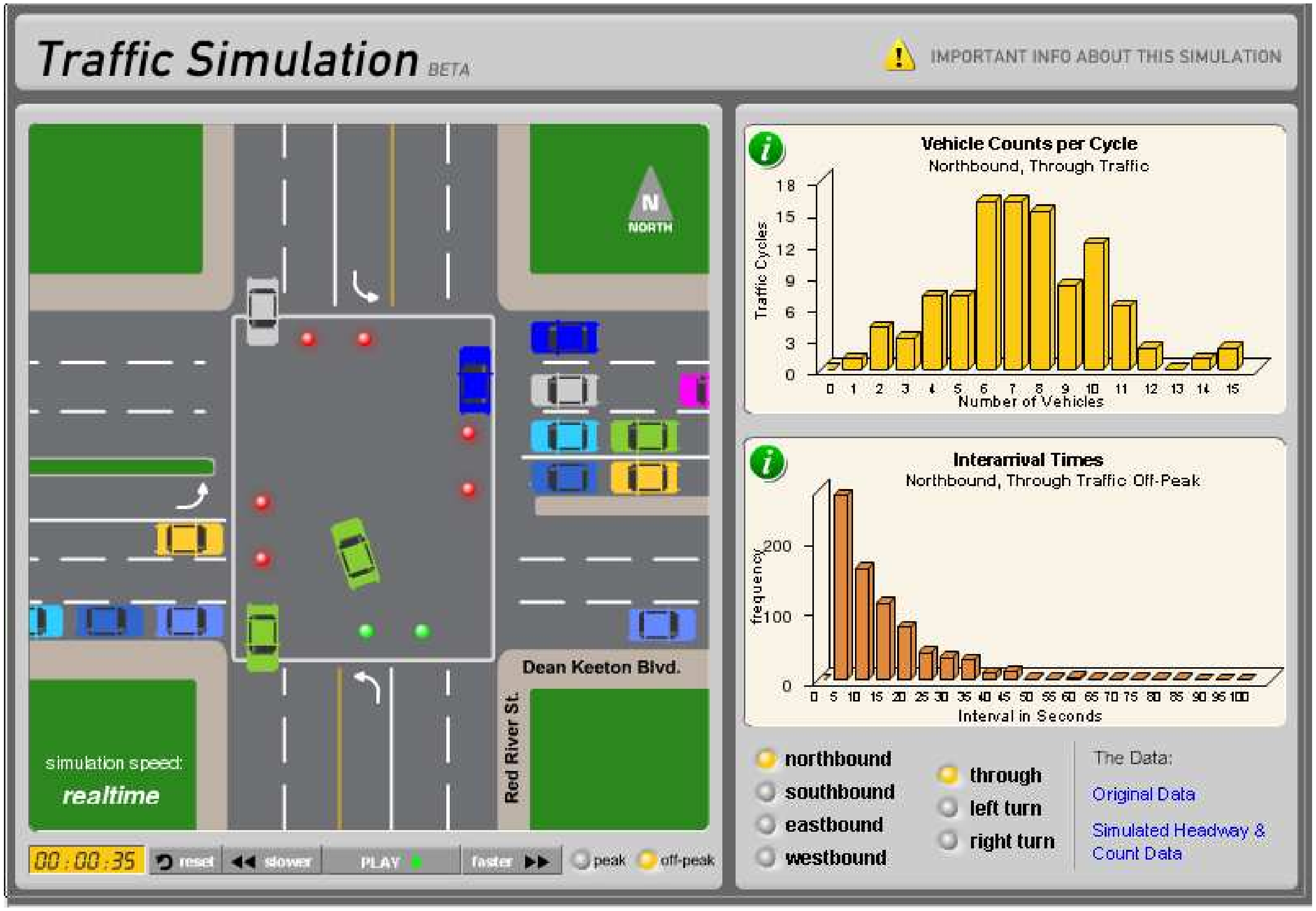
Non-determinism: Traffic network Petri Net



All traces → Reachability Graph

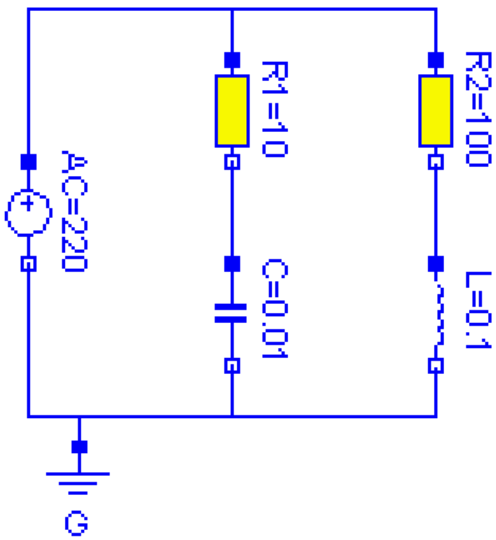


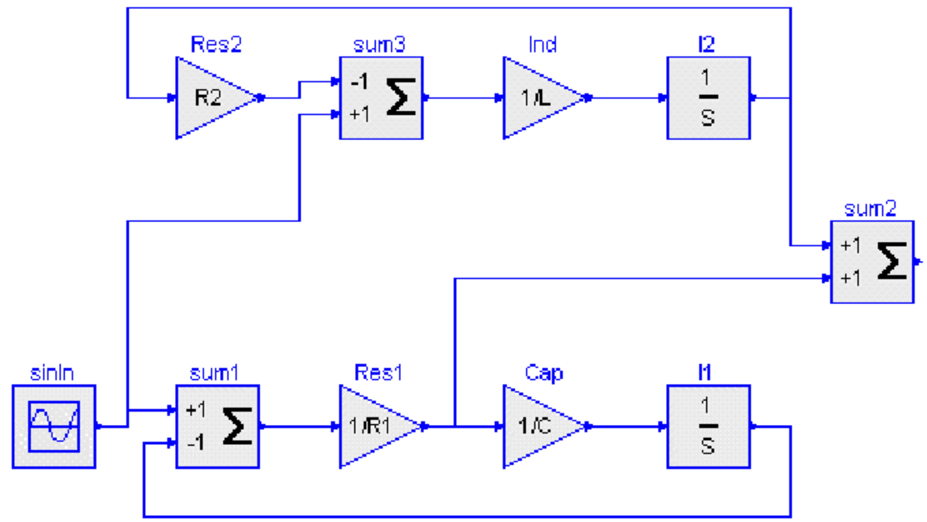
Probabilistic → Monte-Carlo Simulation



www.engr.utexas.edu/trafficSims/

Causality: Modelica vs. Matlab/Simulink

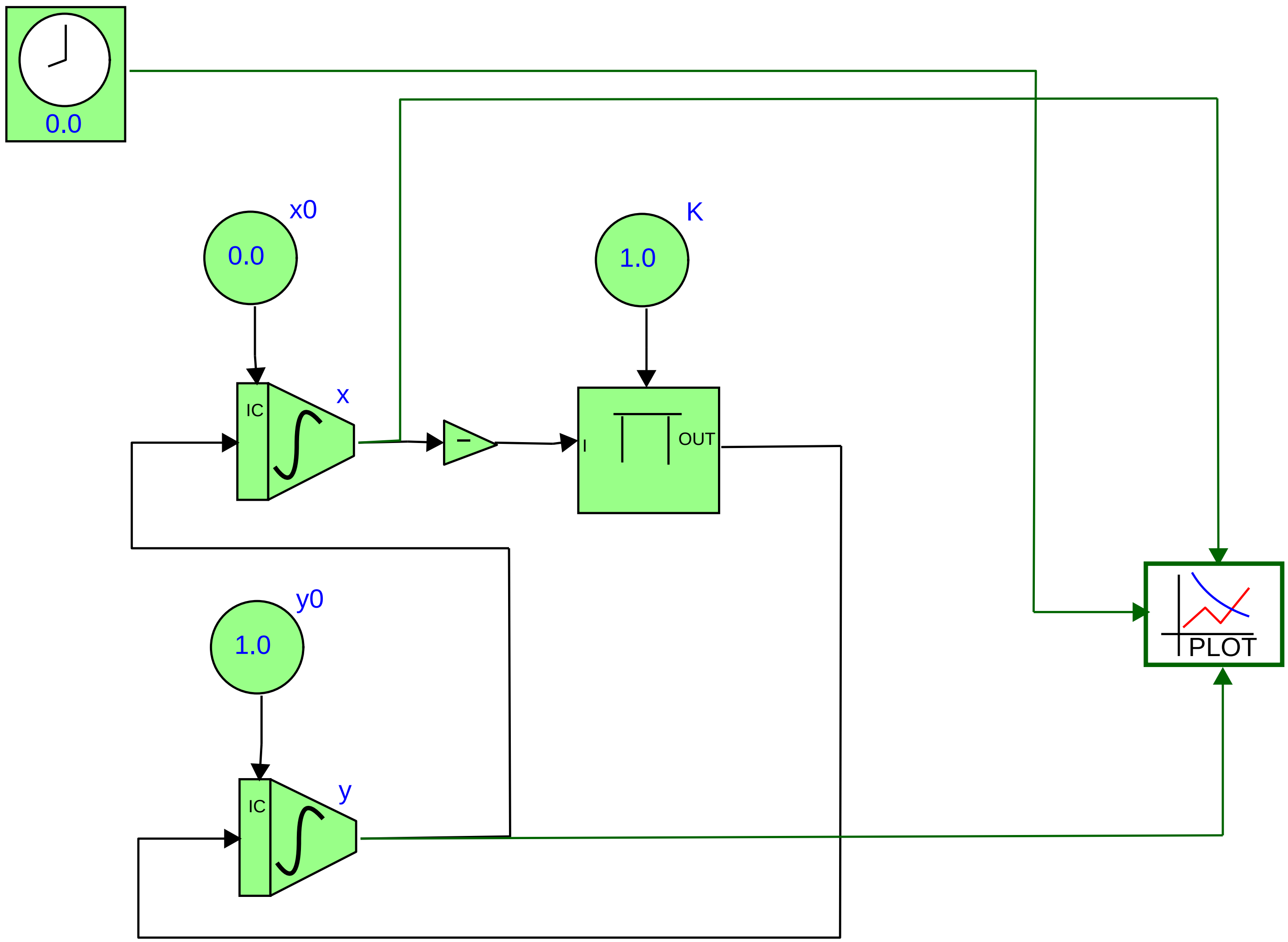




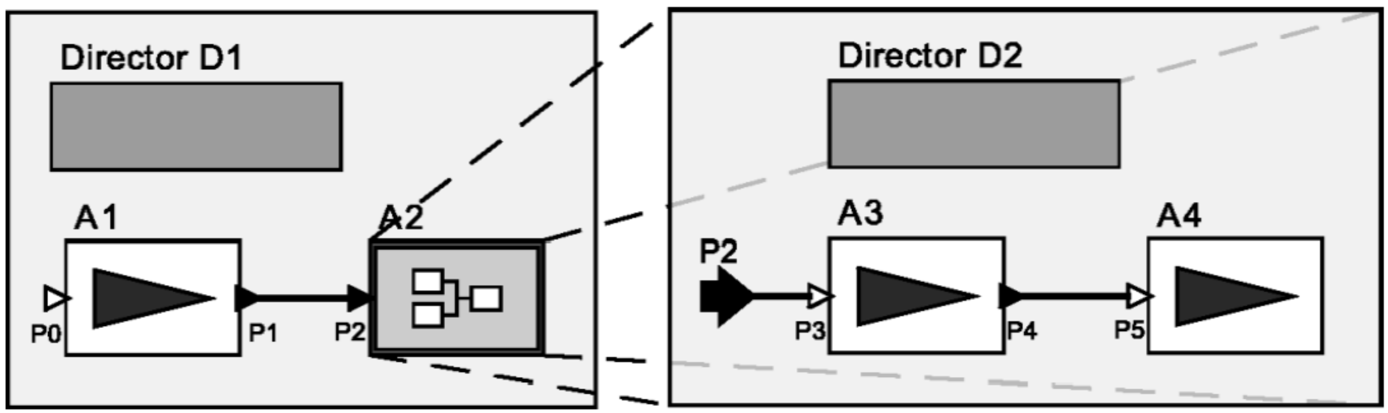
Multicomponent Specification

- Collections of *interacting* components
- *Compositional* modelling
  - *Modular* (interaction through ports only).
    Encapsulated. Allows for *hierarchical (de-)composition*.
  - *non-modular* (direct interaction between components).
    Not encapsulated. “global” variable access. Direct interaction through transition function

Causal Block Diagram



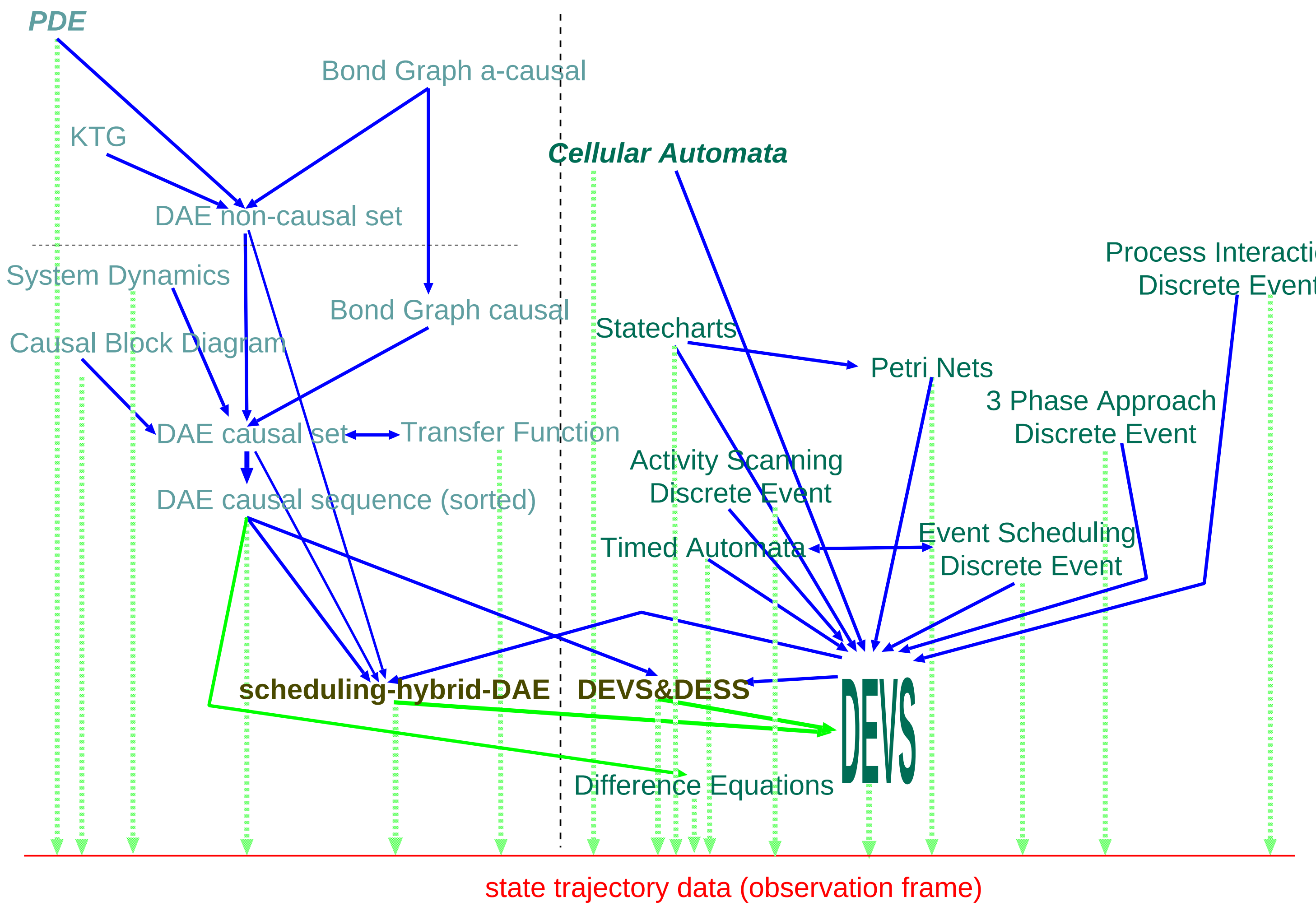
Multi-formalism / Heterogeneous MoC (Ptolemy)



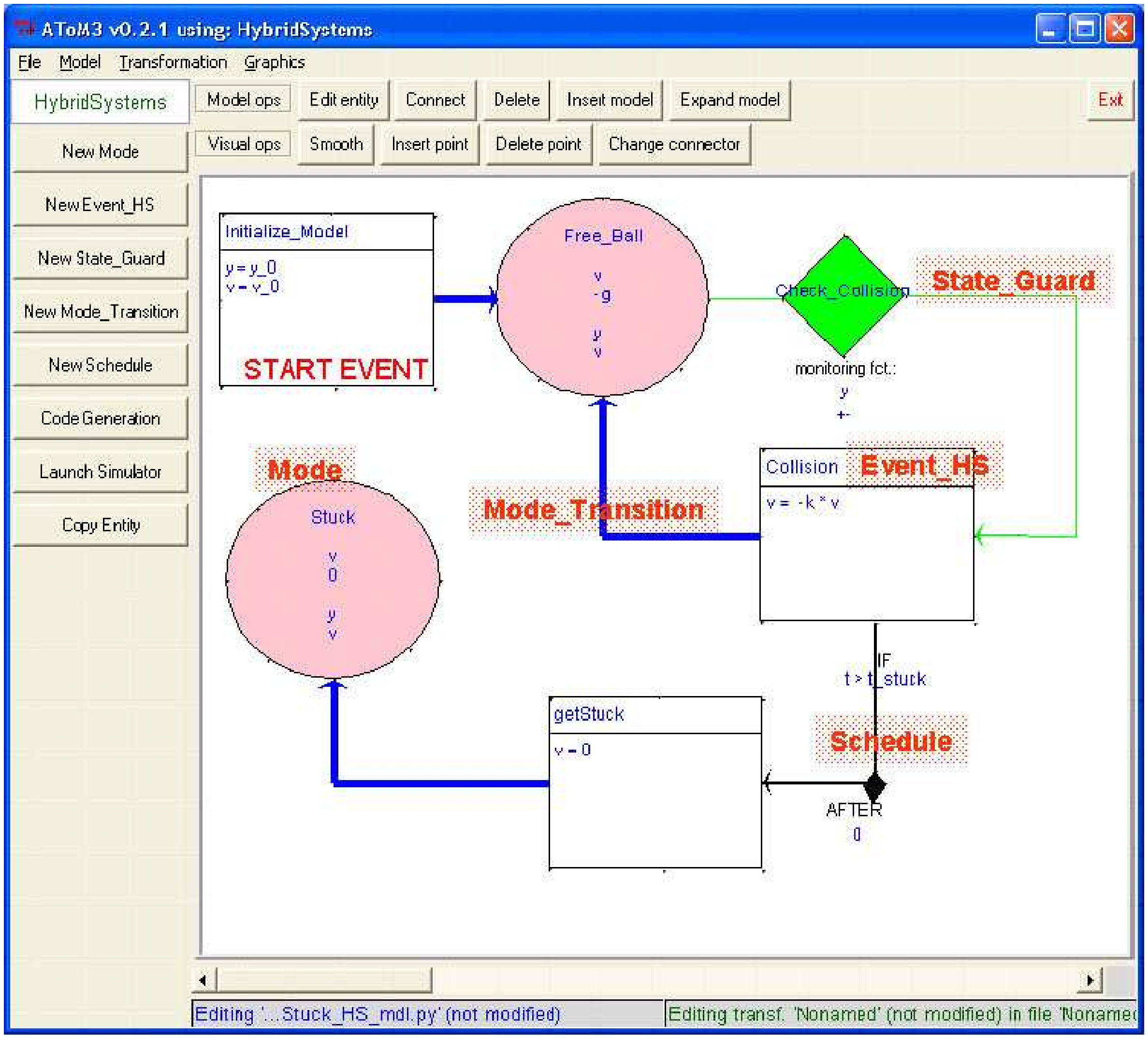
solution:

- co-simulation
- formalism transformation (using graph transformation)

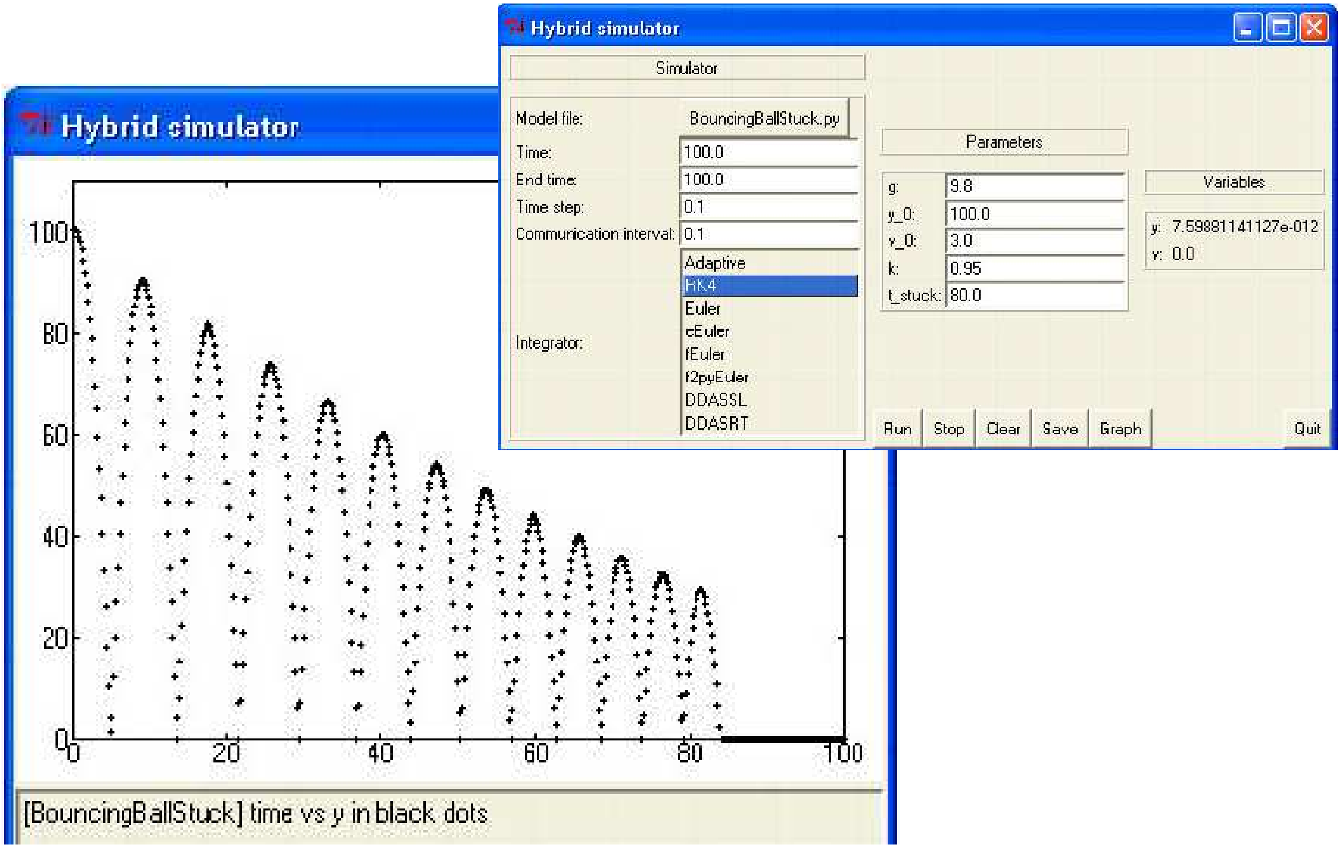
Transform to common Formalism



Hybrid Simulation



Simulation Trace



A Zoo of Formalisms