The Modelling and Simulation Process

1. History of Modelling and Simulation
2. Modelling and Simulation Concepts
3. Levels of Abstraction
4. Experimental Frame
5. Validation
6. Studying a mass-spring system
7. The Modelling and Simulation Process

Modelling and simulation: past

(1950–): Numerical simulations: numerical analysis, statistical analysis,
simulation languages (CSSL, discrete-event world views).
focus: performance, accuracy

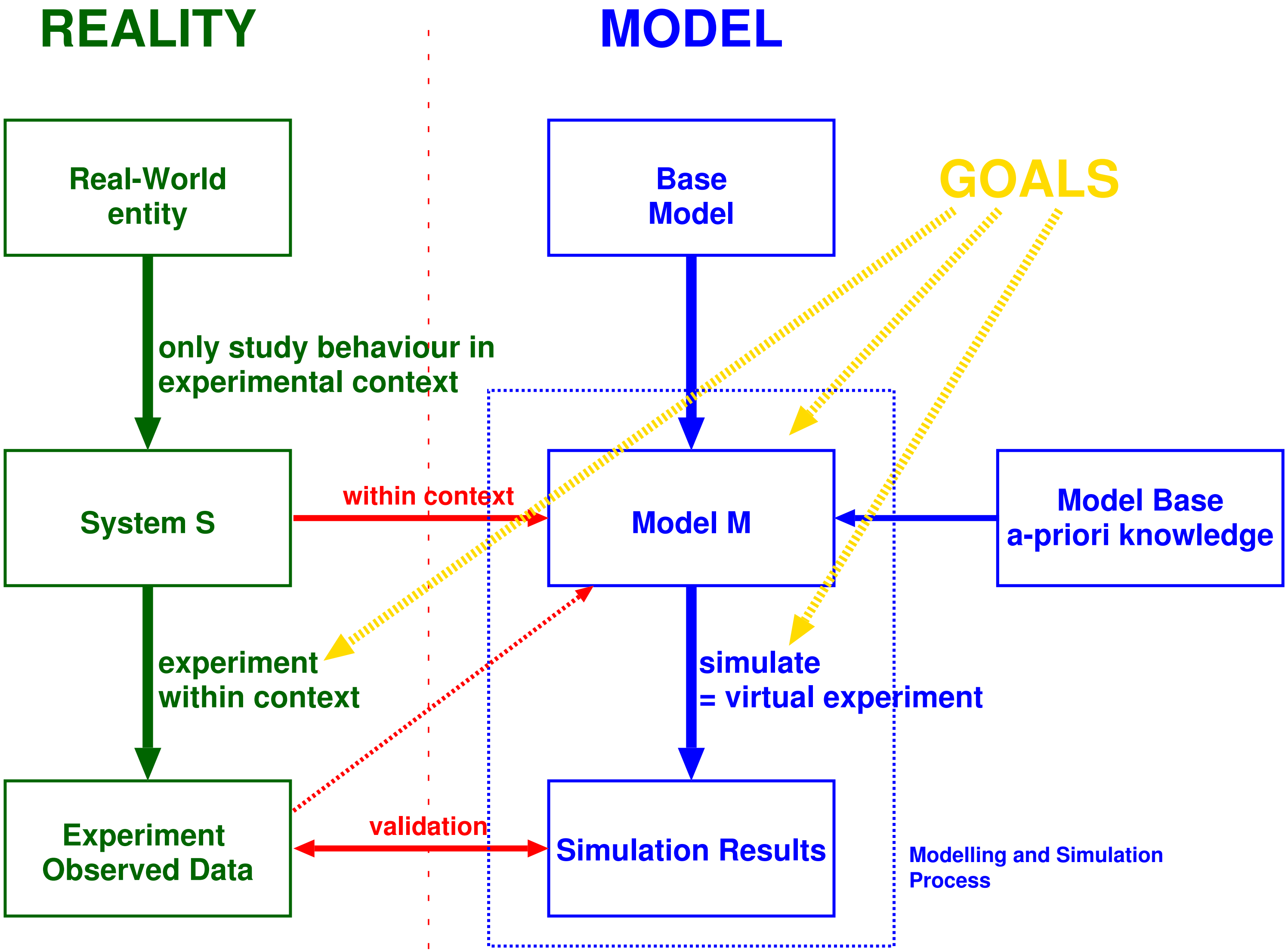
(1981–): Artificial Intelligence: model = knowledge representation
Use AI techniques in modelling, AI uses simulation ("deep" knowledge)
focus: knowledge

(1988–): Object-oriented modelling and simulation
focus: object orientation, later "agents", non-causal modelling

Modelling and simulation: past, present, future

(1993–): Multi-formalism, Multi-paradigm (2001 –)

1. Do it right (optimally) the first time (market pressure)
2. Complex systems: **multi-formalism**
3. Hybrid: continuous-discrete, hardware/software
4. **Exchange** (between humans/tools) and **re-use** (validated model)
5. User focus: do not expect user to know details (software: glueing of components), need for **tools**



REALITY

MODEL

GOALS

Real-World entity

only study behaviour in experimental context

System S

experiment within context

Experiment Observed Data

Base Model

within context

Model M

Model Base a-priori knowledge

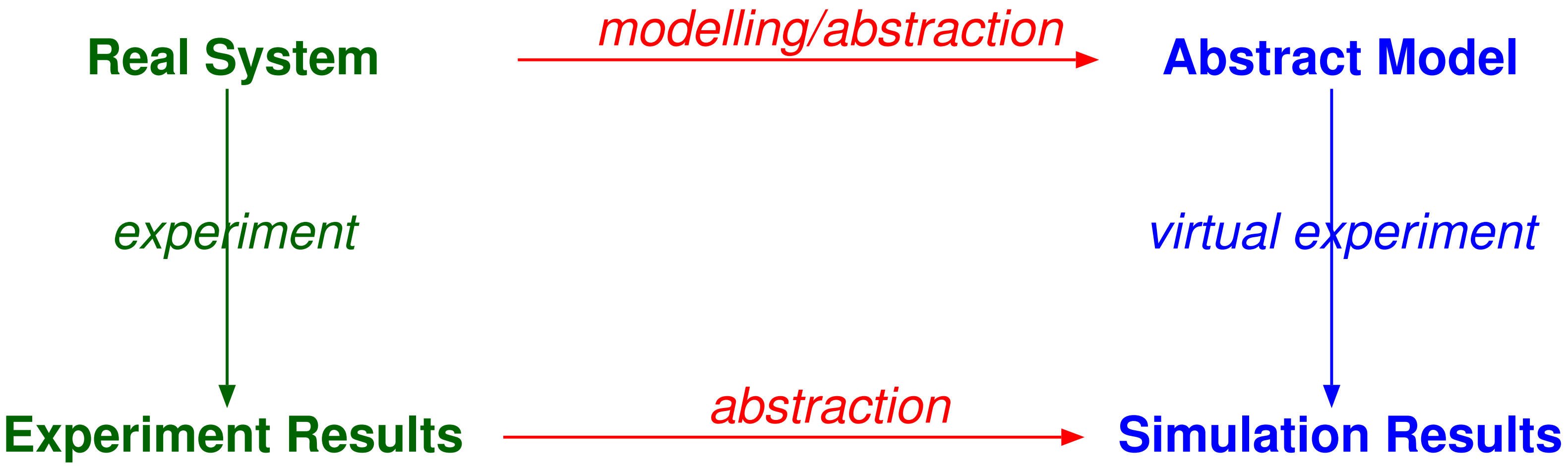
simulate = virtual experiment

validation

Simulation Results

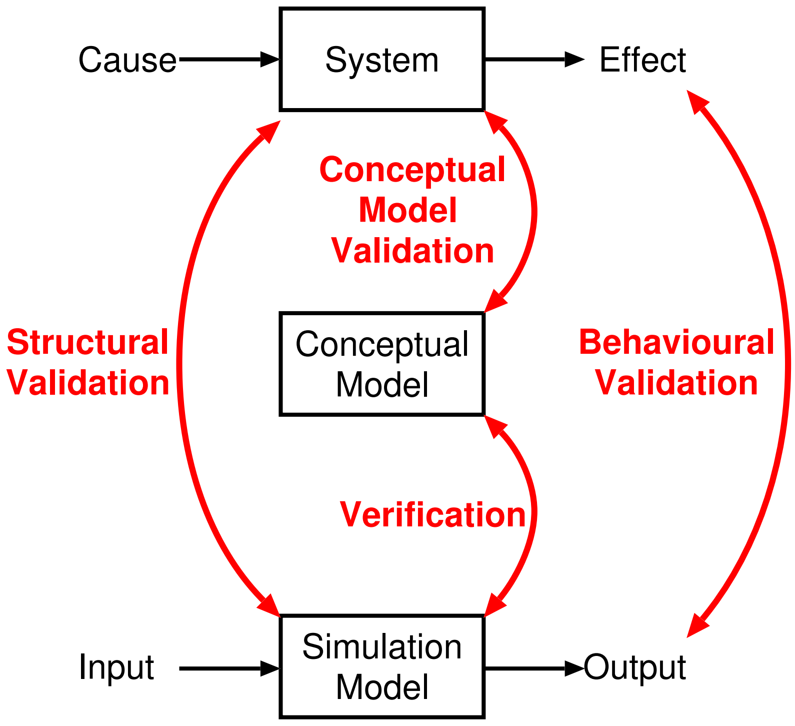
Modelling and Simulation Process

Behaviour (homo)morphism

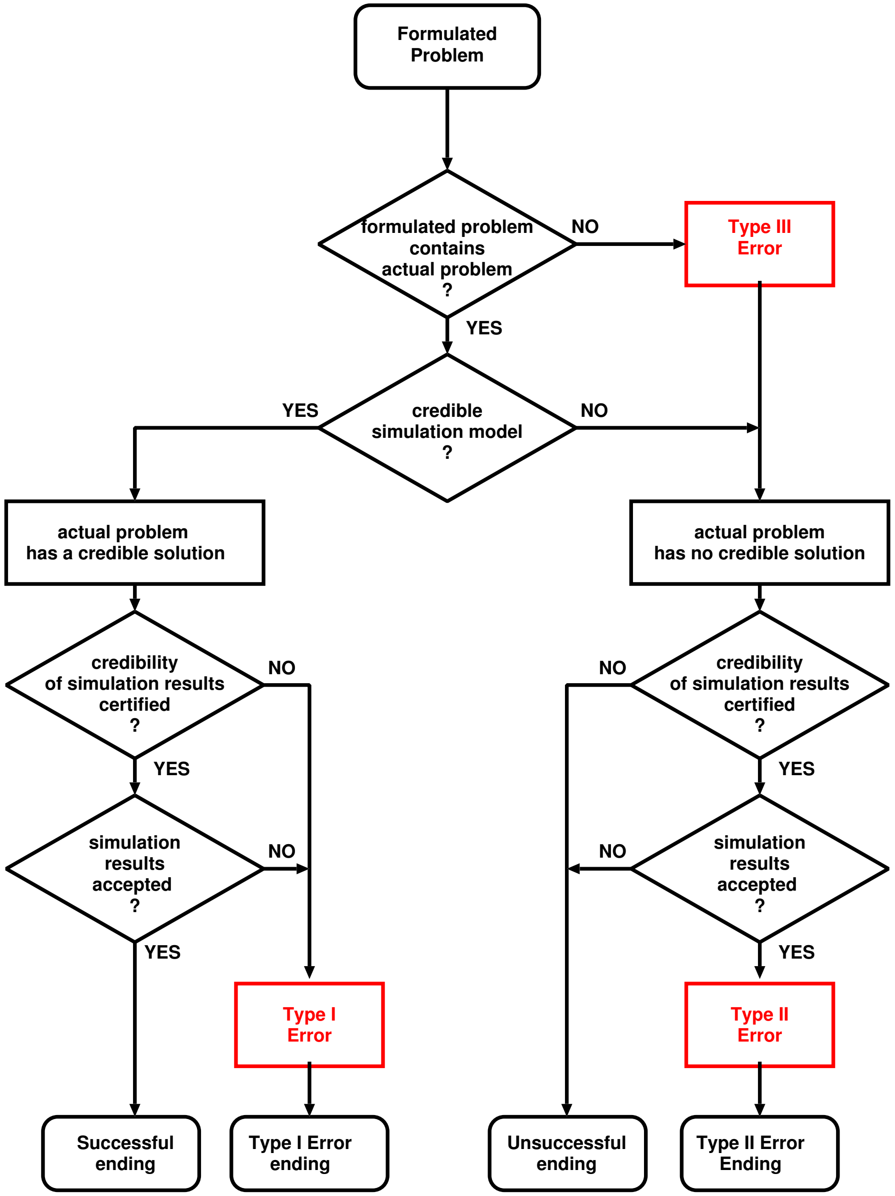


$$\begin{tikzcd}
\textbf{Real System} \arrow[r, "\textit{modelling/abstraction}"] \arrow[d, "\textit{experiment}"] & \textbf{Abstract Model} \arrow[d, "\textit{virtual experiment}"] \\
\textbf{Experiment Results} \arrow[r, "\textit{abstraction}"] & \textbf{Simulation Results}
\end{tikzcd}$$

Verification and Validation



Carl Popper: Falsification, Confidence



Formulated Problem

formulated problem contains actual problem ?

NO

Type III Error

YES

credible simulation model ?

YES

NO

actual problem has a credible solution

actual problem has no credible solution

credibility of simulation results certified ?

NO

YES

simulation results accepted ?

NO

YES

Type I Error

Successful ending

Type I Error ending

credibility of simulation results certified ?

NO

YES

simulation results accepted ?

NO

YES

Type II Error

Unsuccessful ending

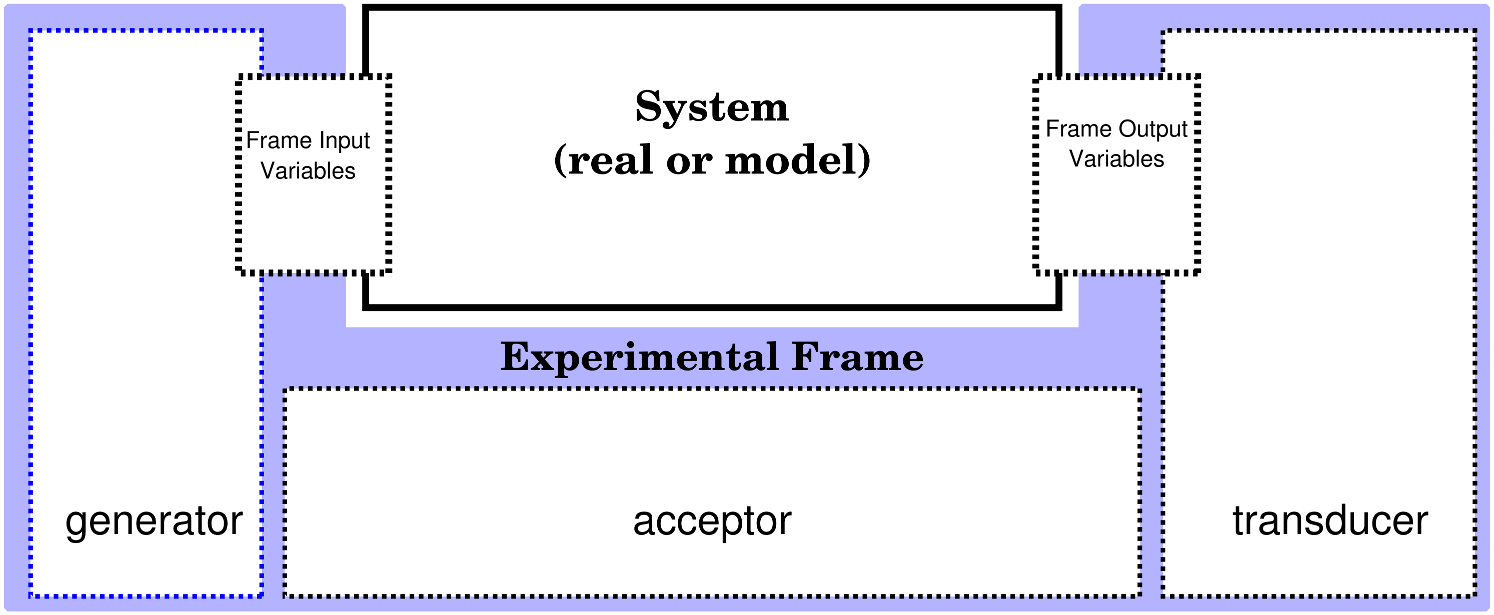
Type II Error Ending

System, Base Model, Lumped Model

$$D_{BaseModel} \equiv D_{RealSystem}$$

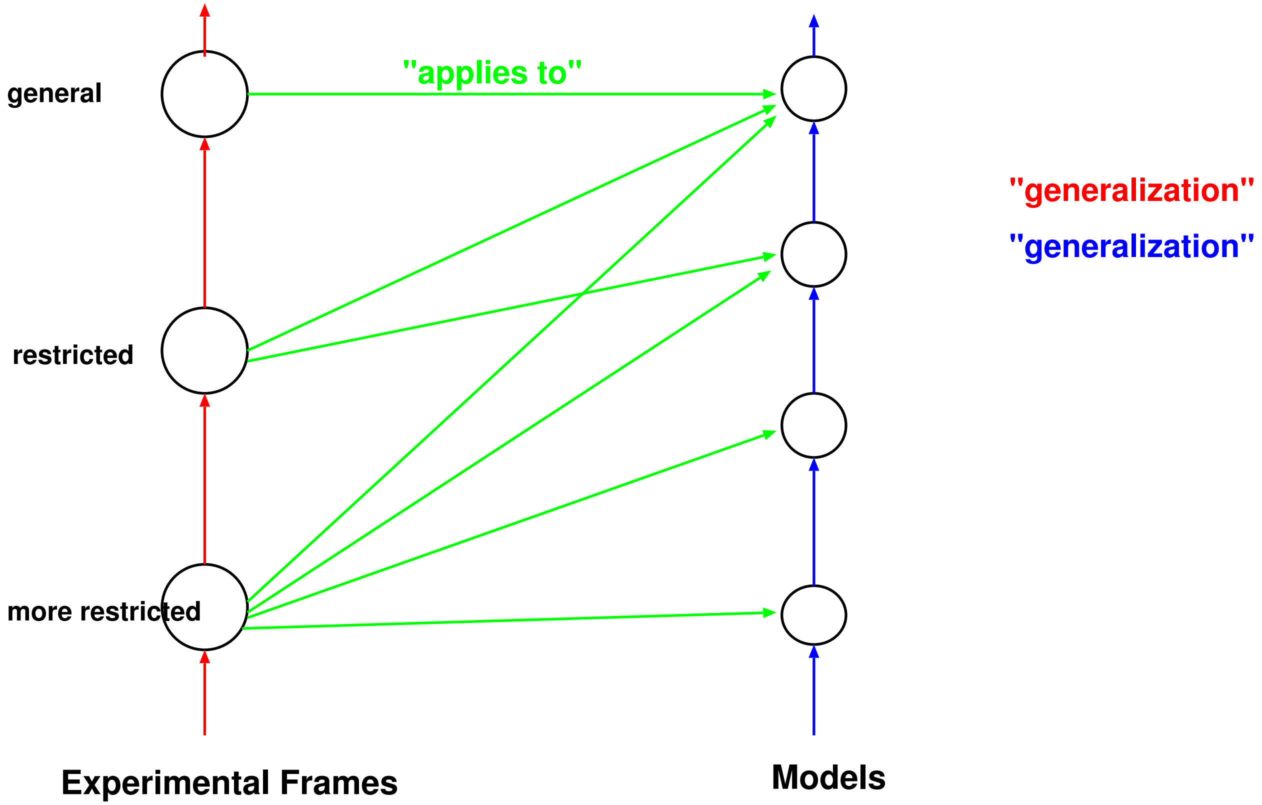
$$D_{LumpedModel}||E \equiv D_{RealSystem}||E$$

Experimental Frame Structure



∼ Programming Language Types, Pre/Post-conditions

Models and matching Experimental Frames



Experimental Frame and Validity

Replicative Validity ($\equiv$: within accuracy bounds):

$$D_{LumpedModel} || E \equiv D_{BaseModel} || E$$

Predictive Validity:

$$F_{LumpedModel} || E \subseteq F_{BaseModel} || E$$

Structural Validity (morphism $\overset{\triangle}{=}$):

$$LumpedModel || E \overset{\triangle}{=} BaseModel || E$$

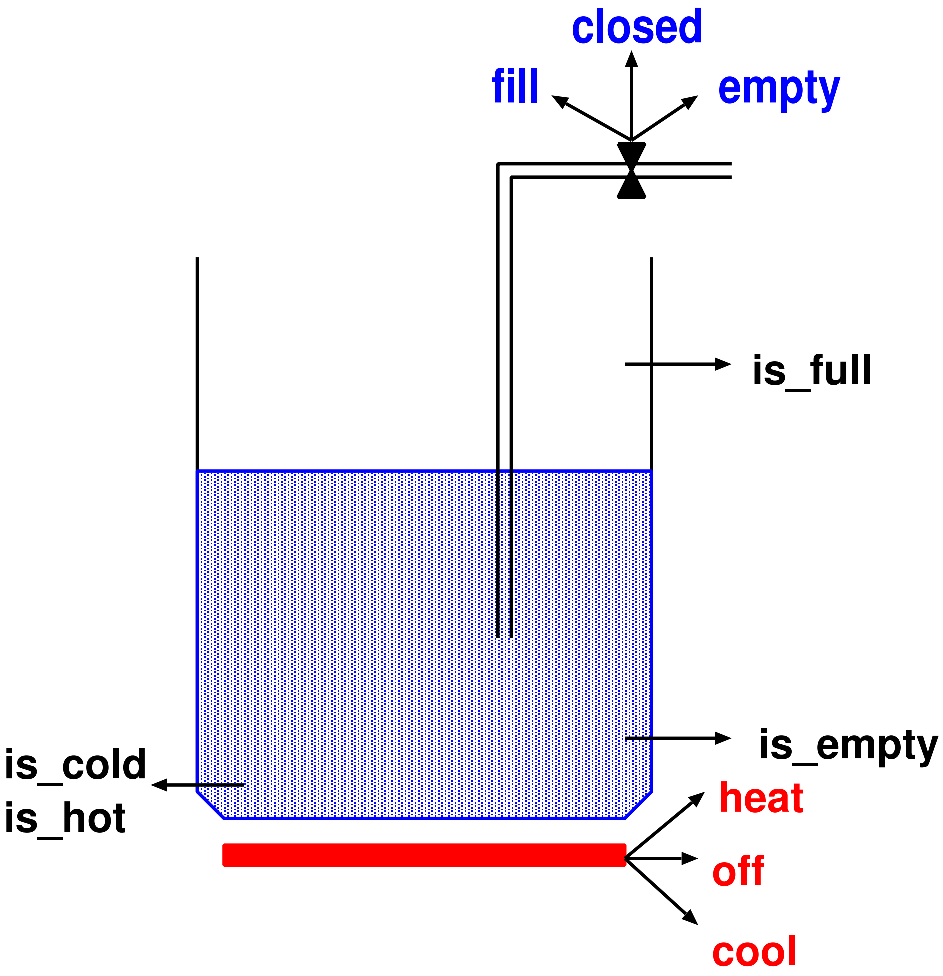
Simulator Verification:

$$D_{Simulator} \equiv D_{LumpedModel}$$

Modelling (and Simulation) Choices

1. System Boundaries and Constraints: Experimental Frame (EF)
2. Level of Abstraction
3. Formalism(s)
4. Level of Accuracy

System under study: T, l controlled liquid



System Boundaries (Experimental Frame)

- Inputs: liquid flow rate, heating/cooling rate
- Outputs: observed level, temperature
- Contraints: no overflow/underflow, one phase only (no boiling)

Abstraction: detailed (continuous) view, ALG + ODE formalism

Inputs (discontinuous → hybrid model):

- Emptying, filling flow rate ϕ
- Temperature of inflowing liquid T_{in}
- Rate of adding/removing heat W

Parameters:

- Cross-section surface of vessel A
- Specific heat of liquid c
- Density of liquid ρ

State variables:

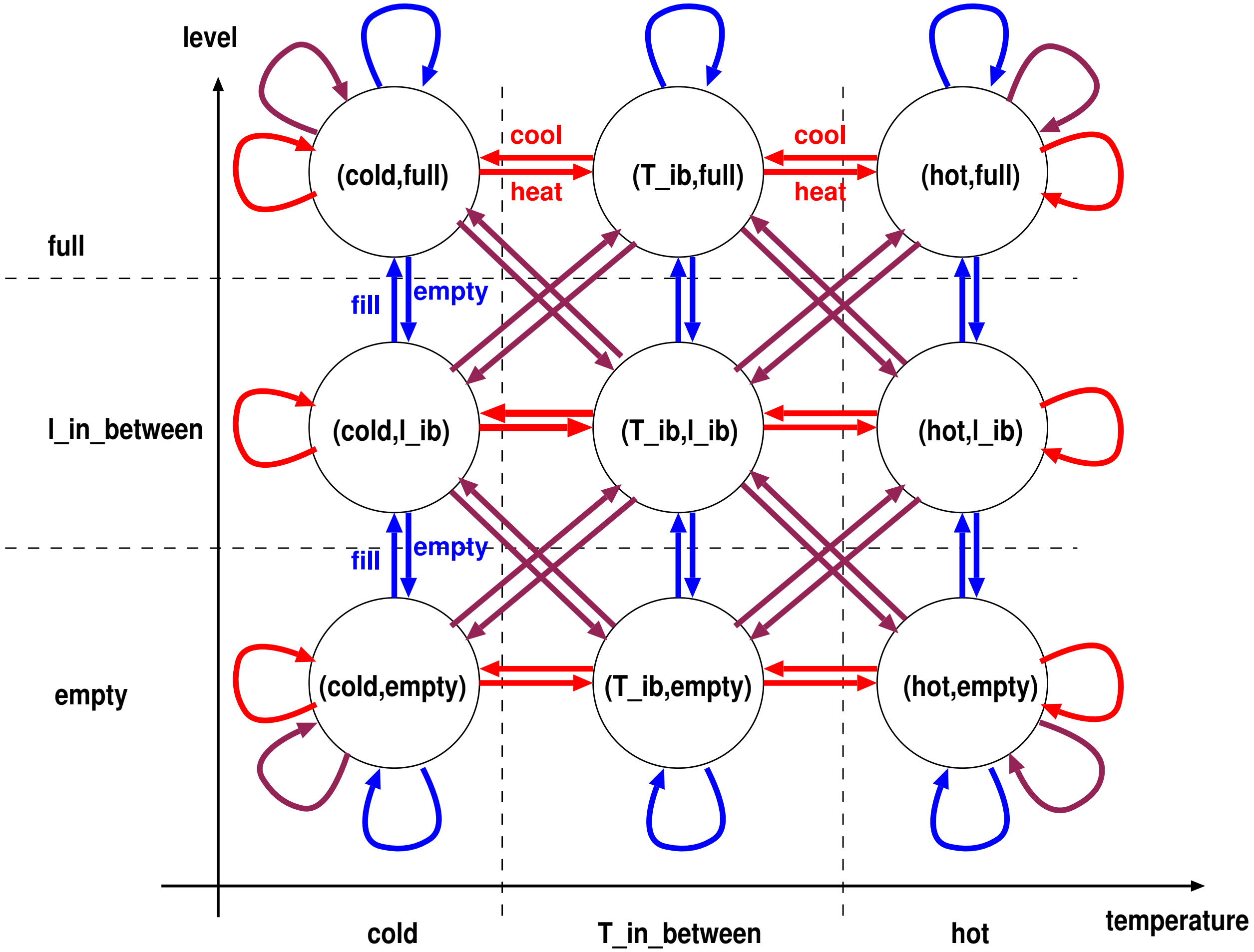
- Temperature T
- Level of liquid l

Outputs (sensors):

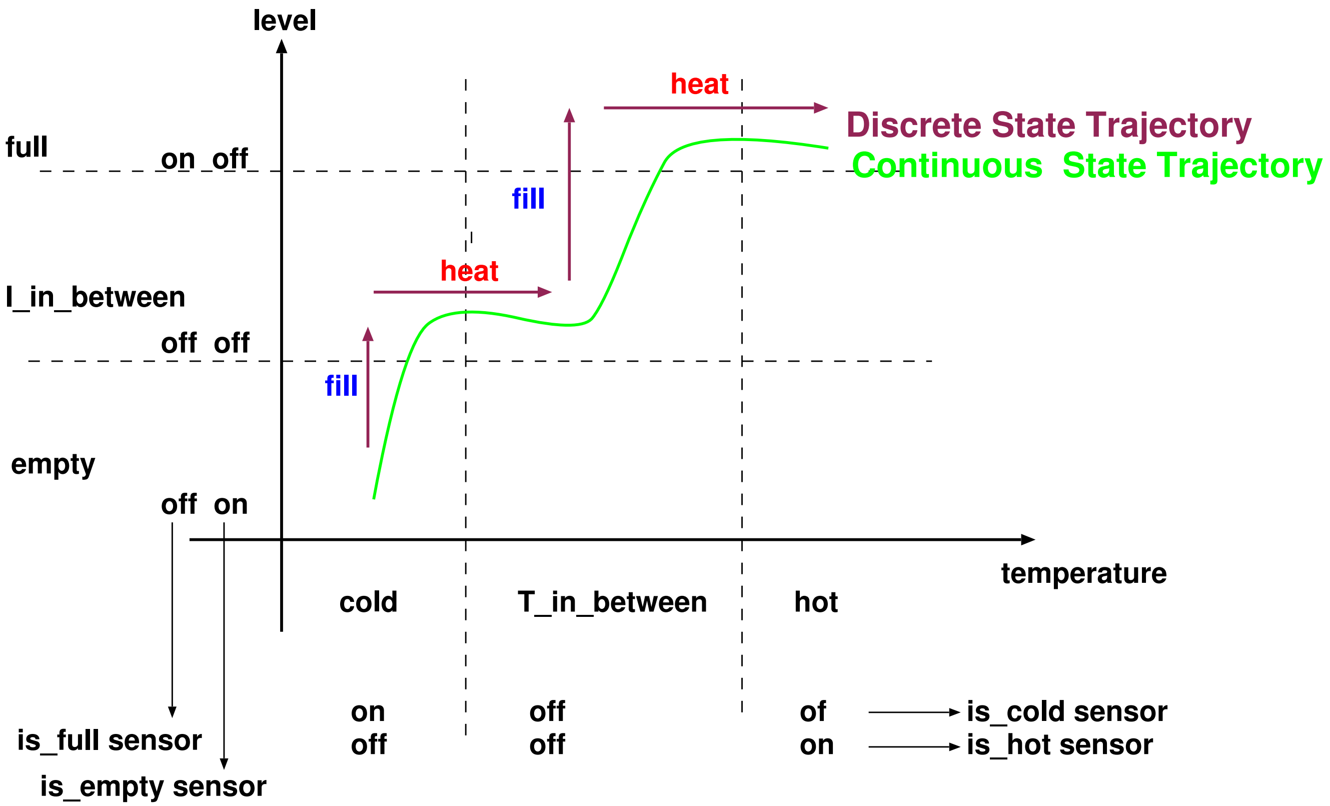
- $is\_low, is\_high, is\_cold, is\_hot$

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

Abstraction: high-level (discrete) view, FSA formalism



Levels of abstraction: trajectories (behaviour)



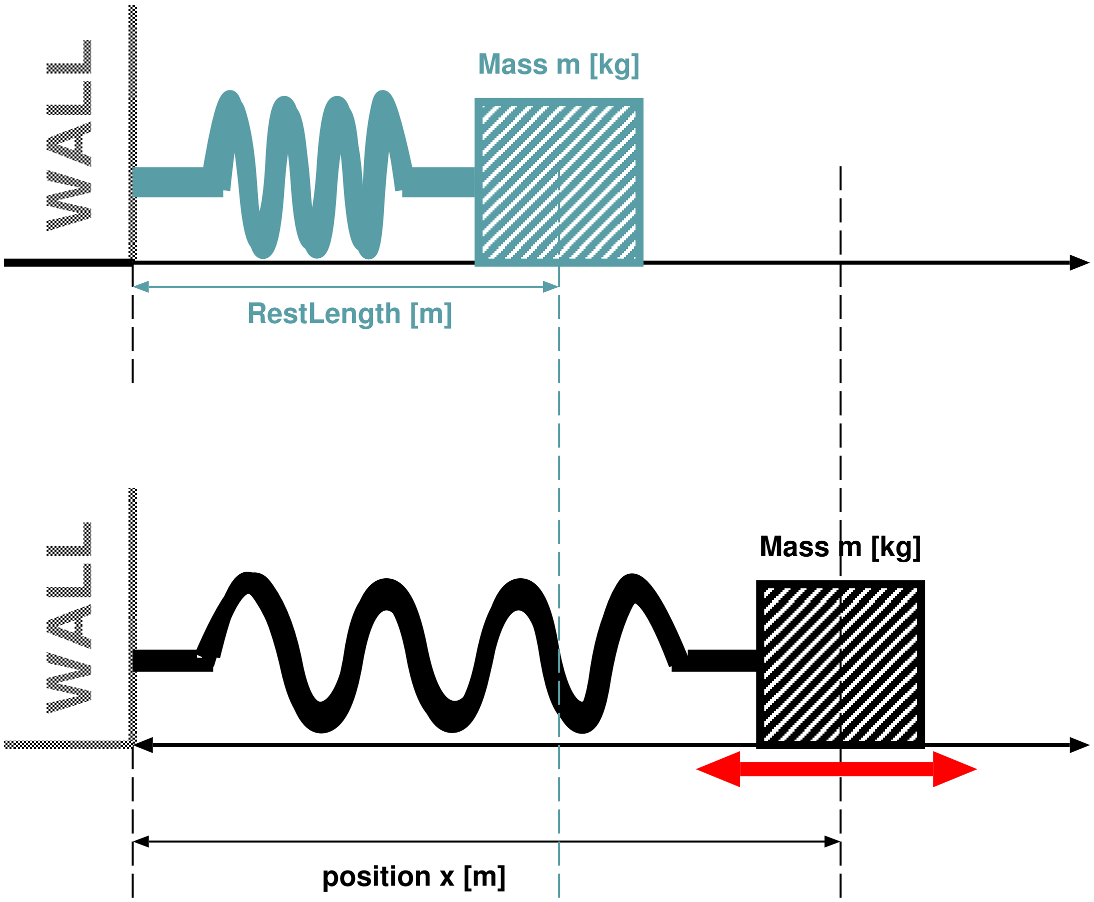
Levels of accuracy

- Depends on “equality” metric (definition of accuracy)
- Depends on choice of formalism
- Depends on choice of numerical approximation

Levels of abstraction: behaviour morphism

$$\begin{tikzcd}
\textbf{model} & \textbf{M\_t} \arrow[r, "\textbf{abstraction}"'] \arrow[d, "\textbf{simulation}"'] & \textbf{M\_d} \arrow[d] \\
\textbf{trajectory} & \textbf{traj\_t} \arrow[r] & \textbf{traj\_d} \\
 & \textbf{detailed (technical) level} & \textbf{abstract (decision) level}
\end{tikzcd}$$

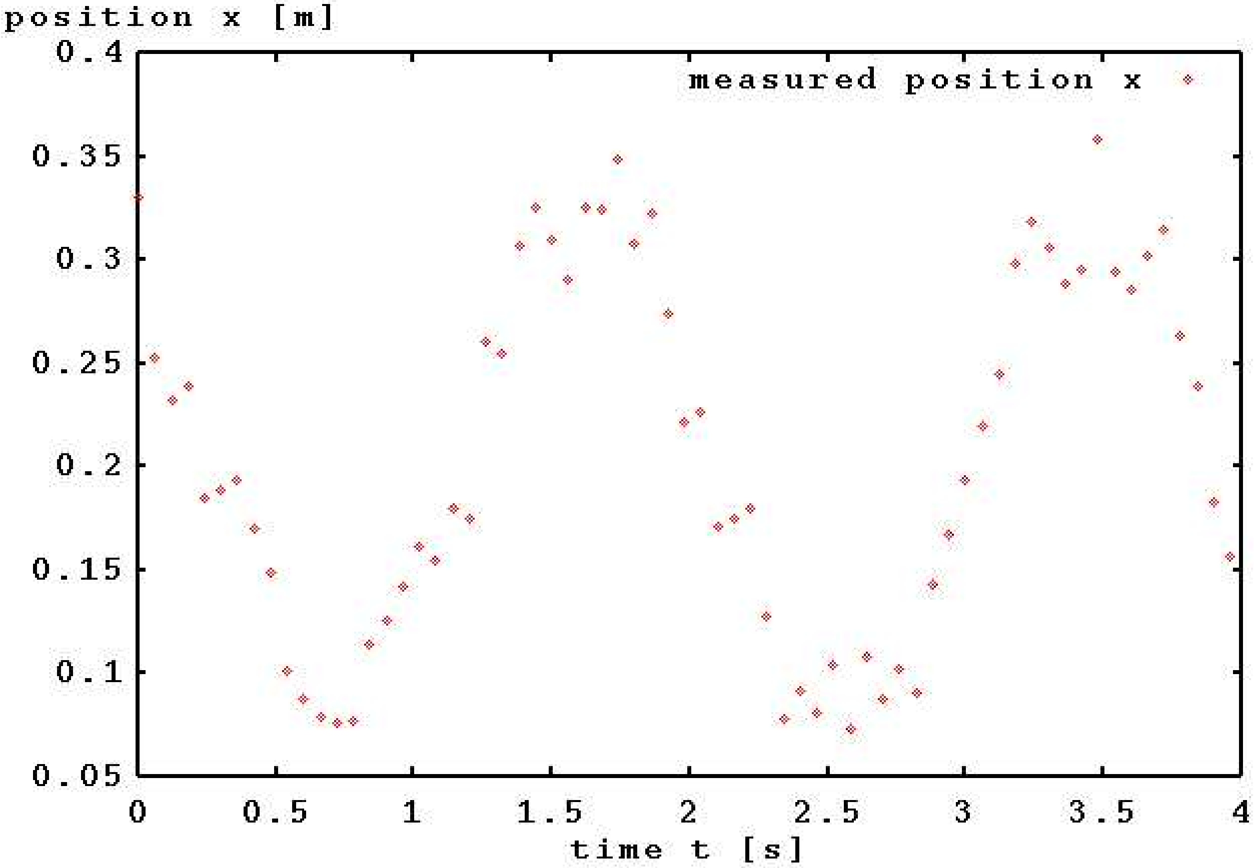
A Modelling and Simulation Exercise: the Mass-Spring system



Knowledge Sources

- A Priori Knowledge: Laws of Physics
- Goals, Intentions: Predict trajectory given Initial Conditions, “optimise” behaviour, . . .
  1. Analysis
  2. Design
  3. Control
- Measurement Data

Measured Data



Experimental Frame

- Room Temperature, normal humidity, . . .
- Frictionless, Ideal Spring, . . .
- Apply deviation from rest position
- Observe position as function of time

Structure Characterisation

- $n-1$-order polynomial will perfectly fit n data points
- Ideal Spring: *Feature* = maximum amplitude constant
- Spring with Damping: *Feature* = amplitute decreases

$\Rightarrow$ Ideal Spring

Building the model from a-priori knowledge

Newton's Law

$$F = M\frac{d^2\Delta x}{dt}$$

Ideal Spring

$$F = -K\Delta x$$

$$\downarrow$$

$$\frac{d^2\Delta x}{dt^2} = -\frac{K}{M}\Delta x$$

Model representation

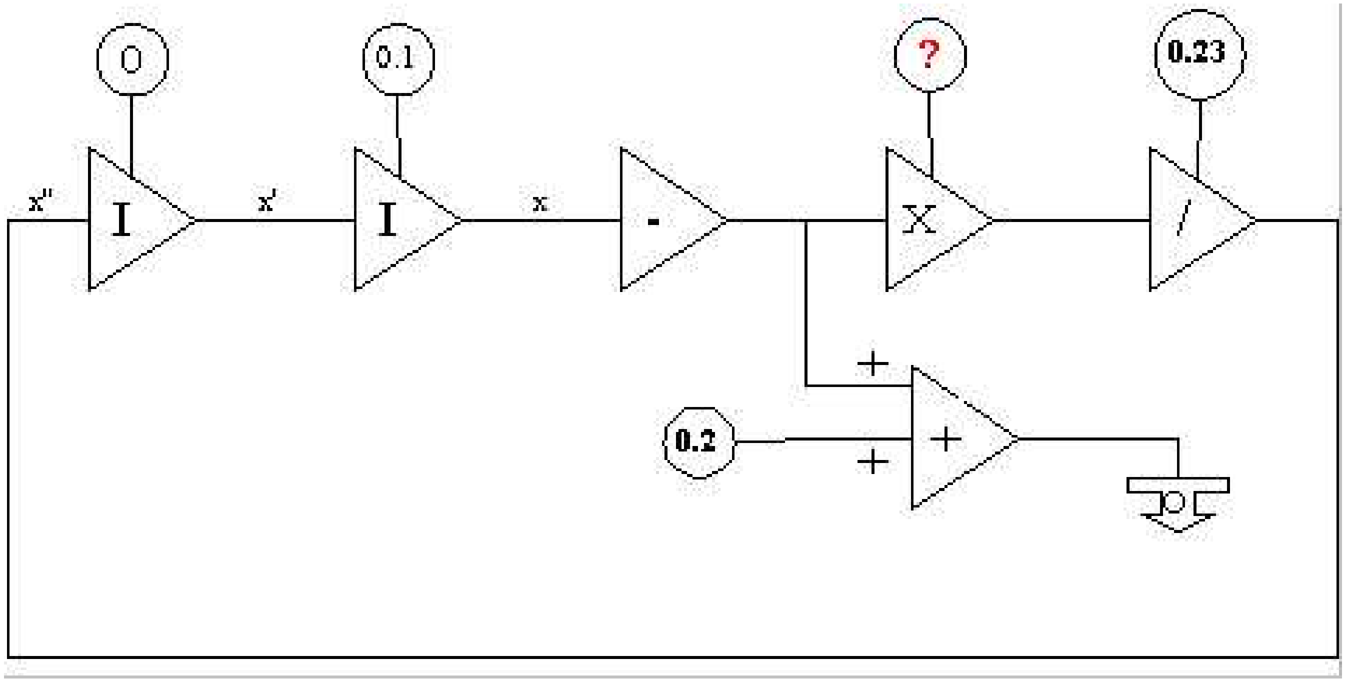
```
CLASS Spring "Ideal Spring": DAEmodel :=
{
 OBJ F_left: ForceTerminal,
 OBJ F_right: ForceTerminal,

 OBJ RestLength: LengthParameter,
 OBJ SpringConstant: SCParameter,

 OBJ x: LengthState,
 OBJ v: SpeedState,

 F_left - F_right = - SpringConstant * (x - RestLength),
 DERIV([ x, [t,] ]) = v,

 EF_assert( x - RestLenght < RestLength/100),
},
```

From Model to Simulation

Block-diagrams

analog computers, Continuous System Modelling Program (CSMP)

- From (algebraic) equation to Block Diagram
- Higher order differential equations

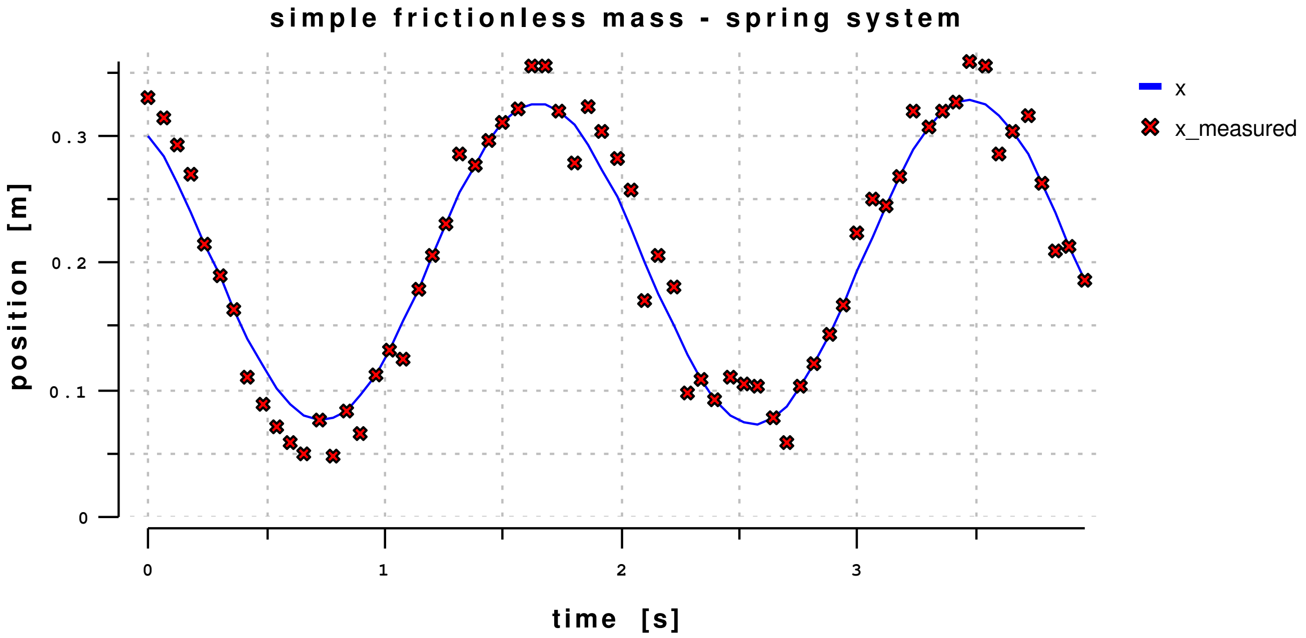


Time-slicing Simulator

Experimentation

1. Model
2. Parameters (constant for each simulation run)
3. Initial Conditions
4. Input (file, interactive, real system)
5. Output (file, plot, real system)
6. Solver Configuration
7. Experiment type
   (simulation, optimization, parameter estimation $=$ model calibration)

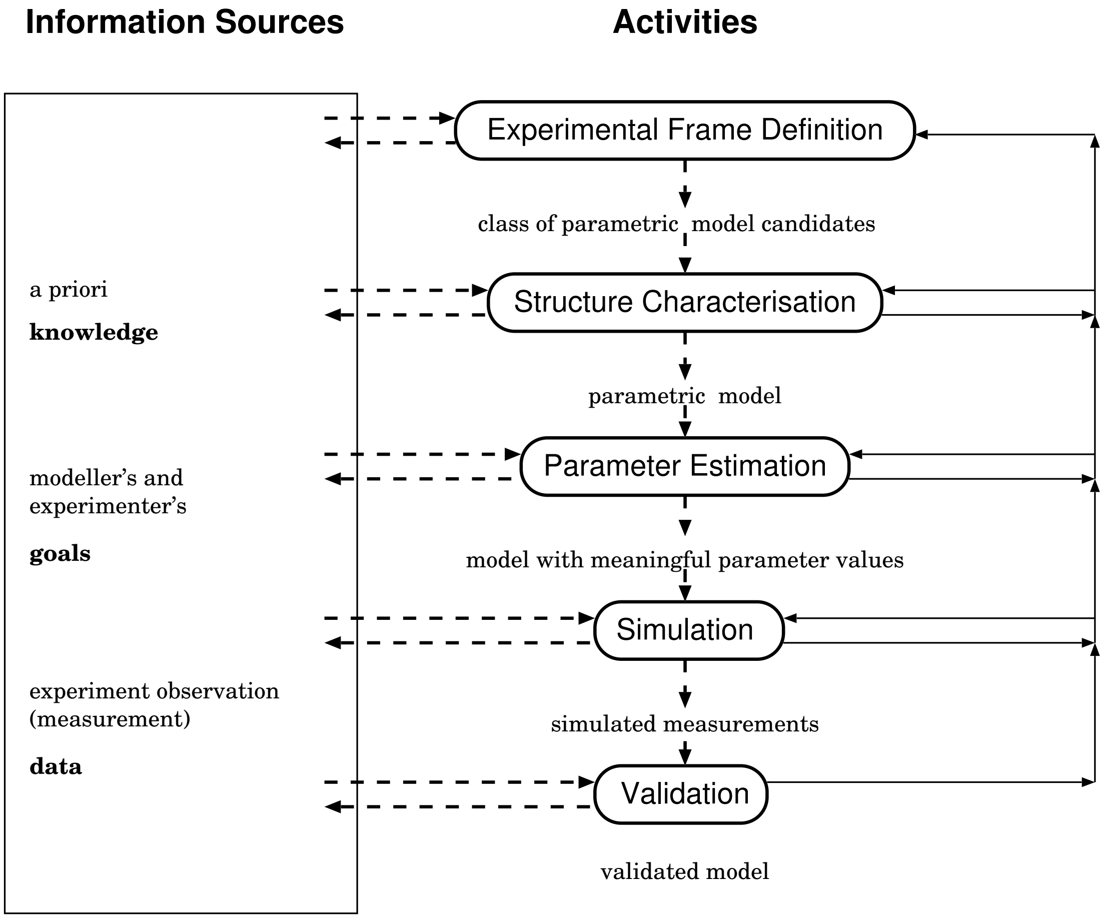
Model Calibration: Parameter Fit

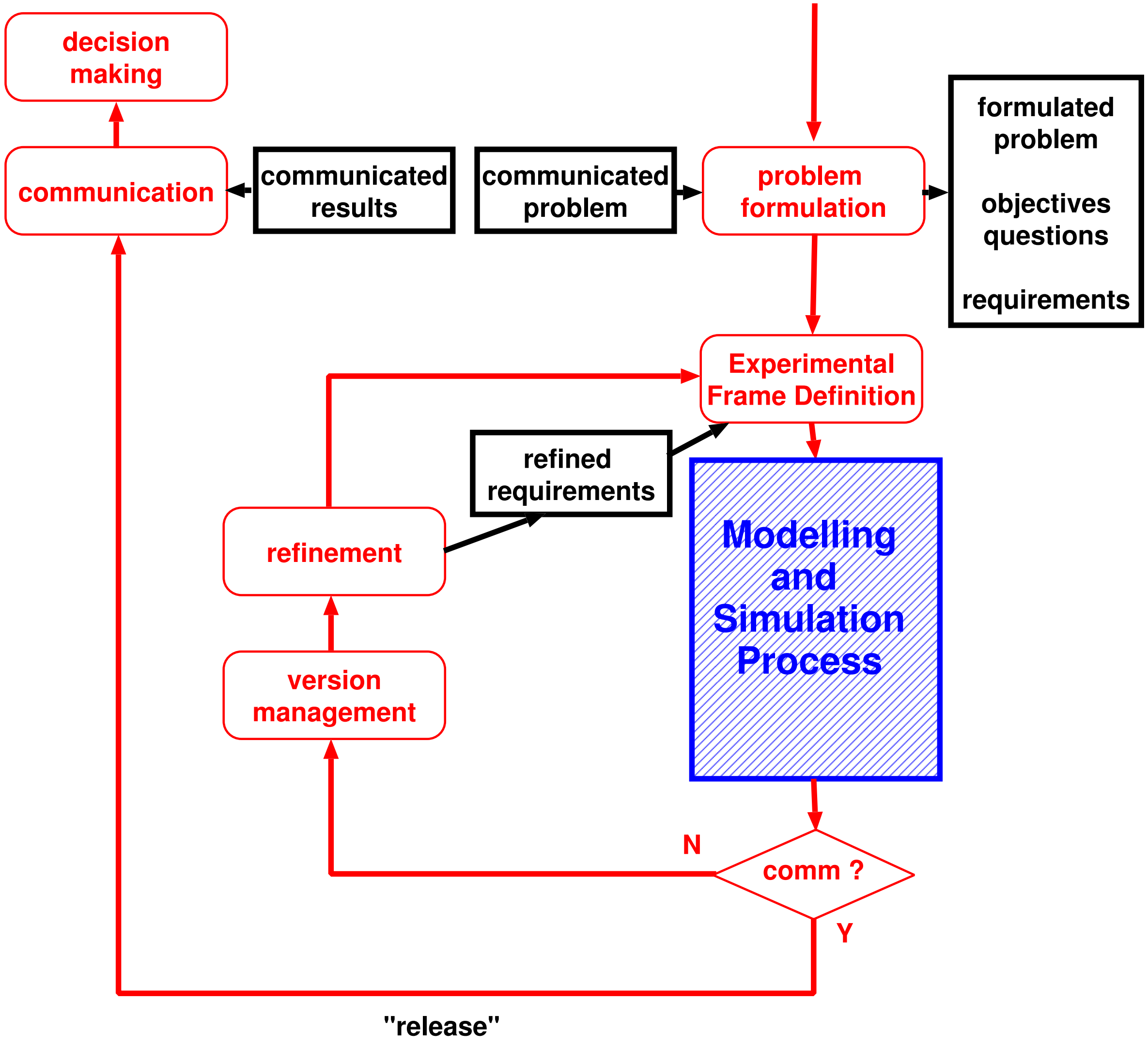


From Here On . . .

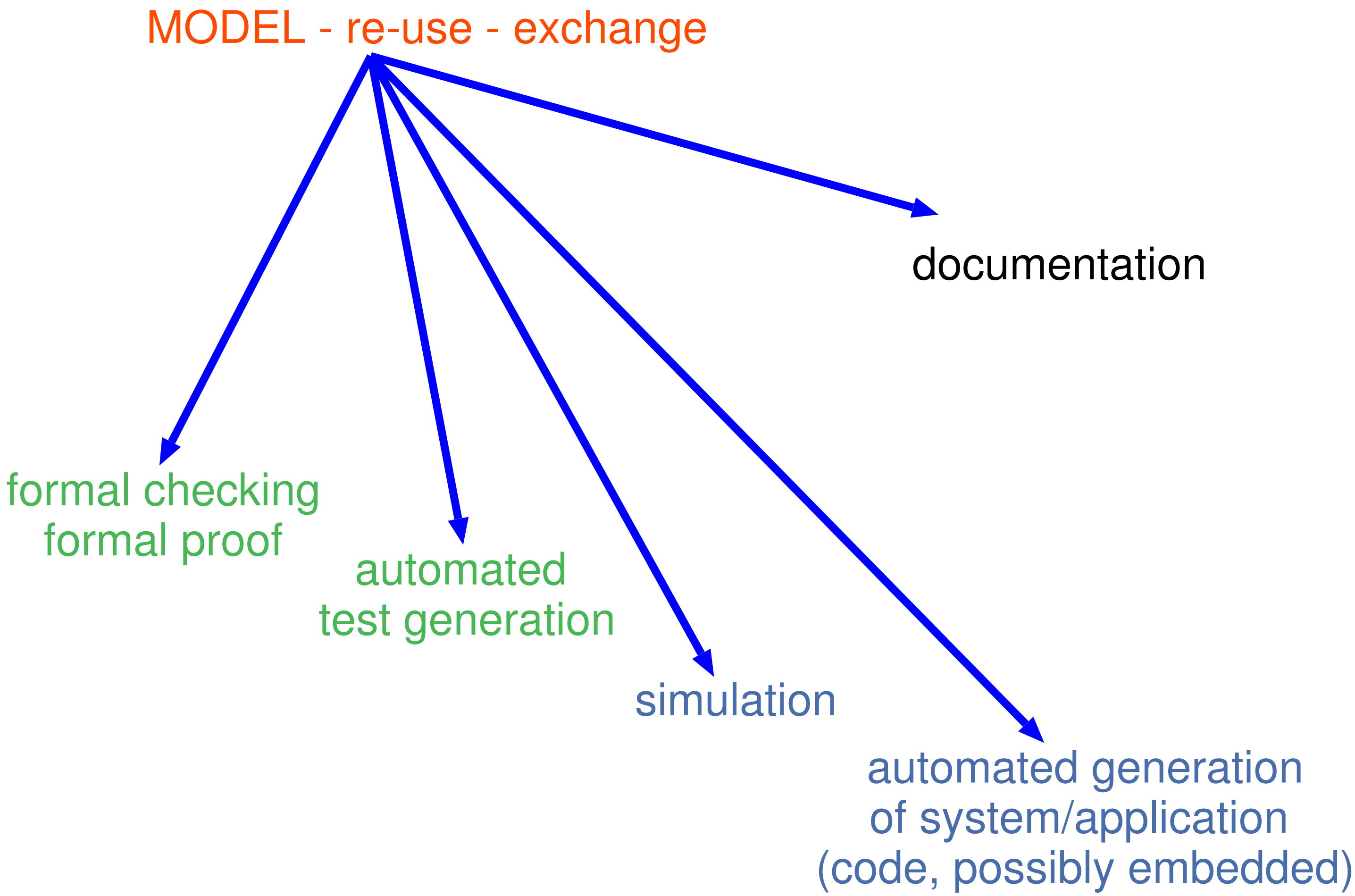
- Virtual Experiments: simulation, optimisation, what-if, . . .
- Validation/Falsification

The Modelling and Simulation *Process*




decision making
communication
communicated results
communicated problem
problem formulation
formulated problem
objectives
questions
requirements
Experimental Frame Definition
refined requirements
refinement
Modelling and Simulation Process
version management
N
comm ?
Y
"release"

Model uses



MODEL - re-use - exchange

documentation

formal checking
formal proof

automated
test generation

simulation

automated generation
of system/application
(code, possibly embedded)