Further Definition of Abstraction

Bentley Oakes, Ahsan Qamar, Pieter Mosterman, Hans Vangheluwe

February 2, 2015

Motivation

- To understand the many guises of abstraction
- To relate the notions of abstraction in many domains

What Do We Gain

- Enable correct communication between modellers about abstraction
- Enables one to enhance modelling tools to support correct reasoning about abstraction
  - Handle element approximations
  - Make explicit the assumptions made
  - Get insights into how effects of approximations and abstractions propagate during compositions

What is a Model?

- **Mapping feature** - A model is based on an original.
- **Reduction feature** - A model only reflects a (relevant) selection of the originals properties.
- **Pragmatic feature** - A model needs to usable in place of the original with respect to some purpose.

Herbert Stachowiak. *Allgemeine Modelltheorie*.
Springer-Verlag, Wien and New York, 1973.

What is Abstraction?

- **Mapping feature** - An abstracted model is based on a model
- **Reduction feature** - An abstraction includes a (relevant) selection of the original model's properties.
- **Pragmatic feature** - An abstraction needs to **be more usable** in place of the original.
  - Notion of 'difficulty/effort to reason about properties'

What is Difficulty?

- Two categories:
  - Cognitive difficulty or effort
    - Difficulty in (human) reasoning about a property
    - Example: Aggregating transistors to an AND gate to reason about behaviour
  - Technical/Computational difficulty
    - Difficulty in (computational) reasoning about a property
    - Example: Transforming statecharts to Petri net to check for deadlocks
- Example of both: Fourier transform of signal to find periodicity
- Reasoning about properties can be made less difficult, or made possible

Effect of Assumptions

- How effective user's reasoning is will be dependent upon the assumptions made in their abstraction
- Assumptions may make the abstraction no longer valid
  - For the same input, the abstraction and the system produce different behaviour or properties
    - Notion of tolerance and approximation fits in here
    - Again, behaviour and properties must be relevant
  - Loss of correspondence between the abstraction and system
- Therefore, it would be beneficial to make assumptions explicit in abstractions
  - Need to select 'relevant' assumptions

Difficulty Perception

- Abstraction user thinks that using an abstraction will reduce cognitive or computational difficulty/effort
- This perception of difficulty/effort may vary widely between users
- As well, this difficulty may change over time as new algorithms or methodologies become available

Three Types of Abstraction

- Physical world to physical world
  - Systems as models for other systems
  - Example: A thrown ball as a model of a fired cannon shell
  - An example of a metaphor
  - Reduction in cognitive effort to understand behaviour

Three Types of Abstraction

- Physical world to virtual world
  - Creation of models to reflect physical world
  - Example: Building a model of a resistor
  - Enables simulation of properties
  - Example: Can now determine behaviour in isolation or within a connected system
  - Note that reasoning is based on validity of model (assumptions made)

Three Types of Abstraction

- Virtual world to virtual world
  - Creating new versions of our models
  - Example: Changing formalisms or level of abstraction
  - Affects both cognitive effort and computational effort
    - Tradeoff could happen based on intent
    - Example: Transformation to Petri nets may produce unreadable model, but enables detection of deadlocks