Software Engineering of Robots

Ana Cavalcanti
Jon Timmis, Jim Woodcock
Wei Li, Alvaro Miyazawa, Pedro Ribeiro
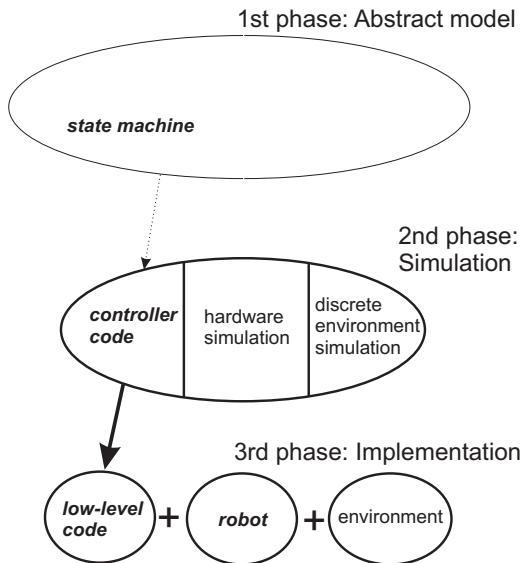
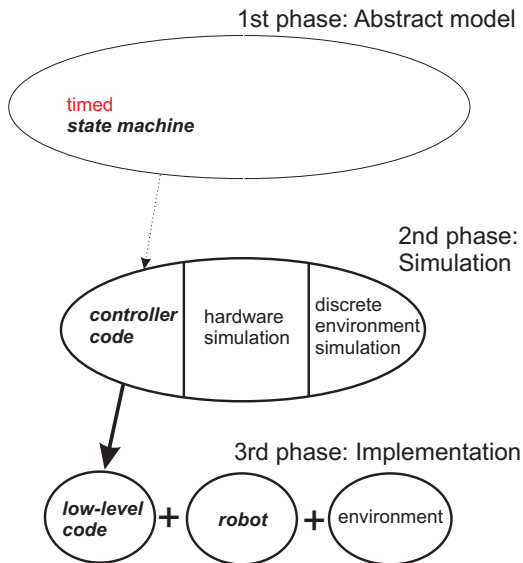University of York

December 2015

Overview

- One of UK eight great technologies: robotics and autonomous systems.
- £13 billion global market predicted for 2025
- Safety: numerous applications of concern
- Autonomous vehicles
- Home automation
- Full verification is beyond the state of the art
- Among other concerns: verification of controller software
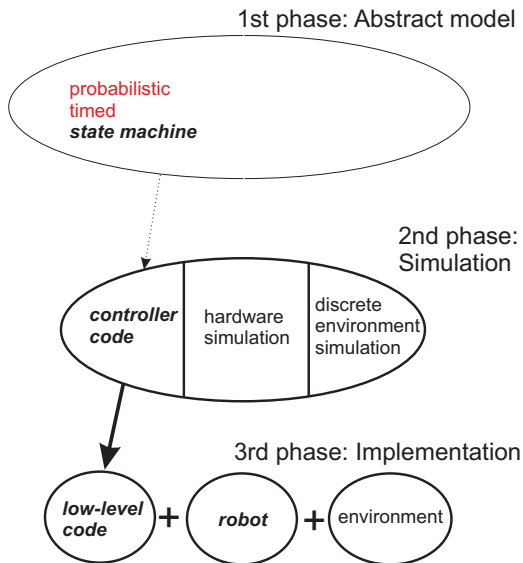- Models enable validation of requirements
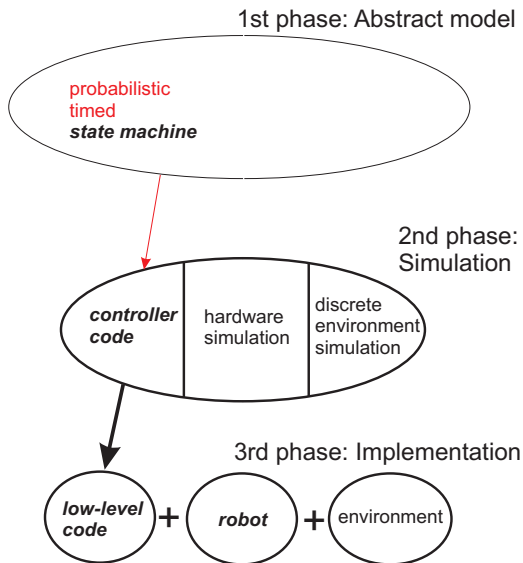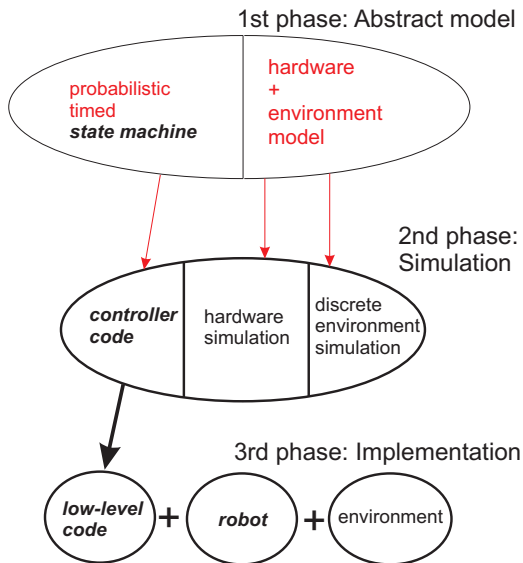
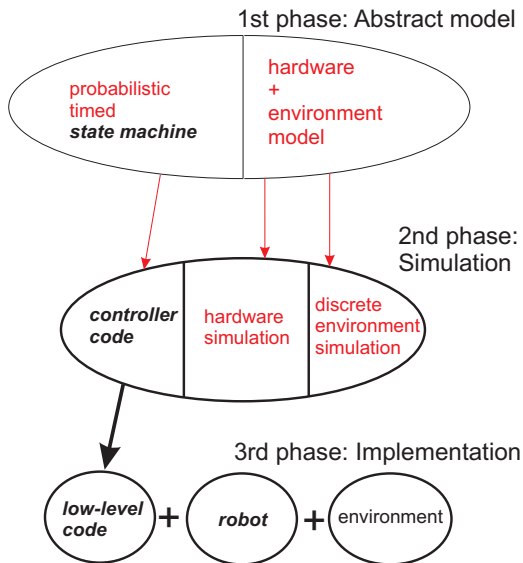# Current approach to development

What do we want to do?

1st phase: Abstract model

timed
state machine

2nd phase: Simulation

controller code | hardware simulation | discrete environment simulation

3rd phase: Implementation

low-level code + *robot* + environment

What do we want to do?

1st phase: Abstract model

probabilistic
timed
state machine

2nd phase:
Simulation

*controller
code* | hardware
simulation | discrete
environment
simulation

3rd phase: Implementation

*low-level
code* + *robot* + environment

What do we want to do?

1st phase: Abstract model

probabilistic
timed
state machine

2nd phase:
Simulation

controller code | hardware simulation | discrete environment simulation

3rd phase: Implementation

low-level code + *robot* + environment

What do we want to do?

1st phase: Abstract model

probabilistic timed *state machine*

hardware + environment model

2nd phase: Simulation

controller code

hardware simulation

discrete environment simulation
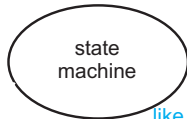
3rd phase: Implementation

low-level code + *robot* + environment

What do we want to do?

What do we want to do?
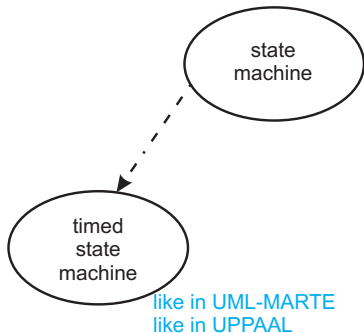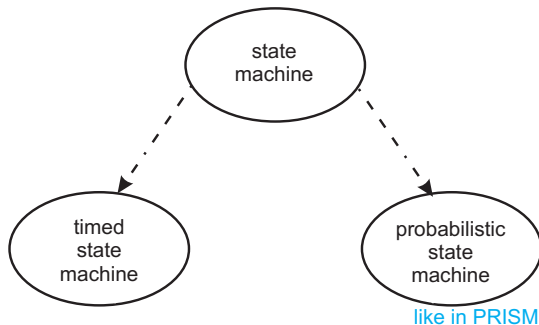
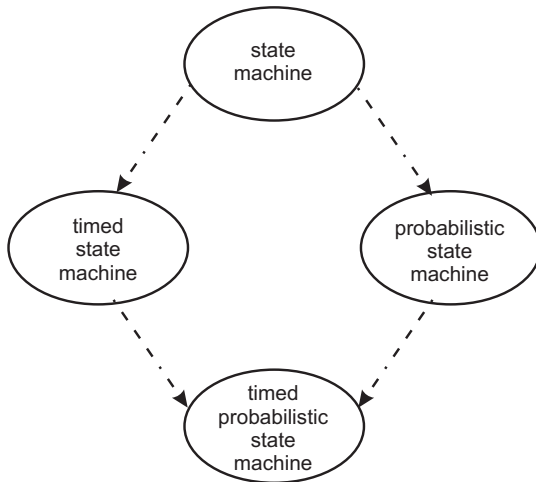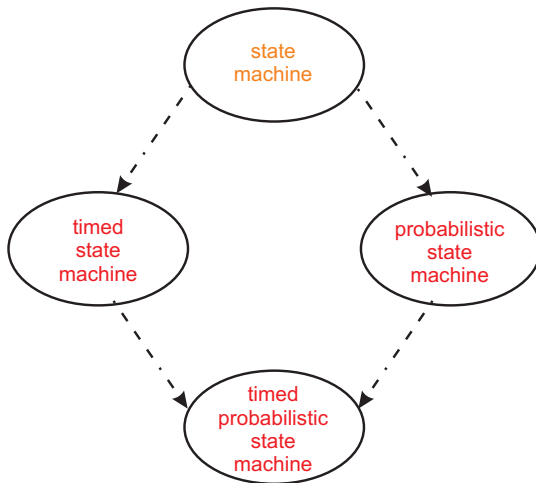# How do we want to do it?
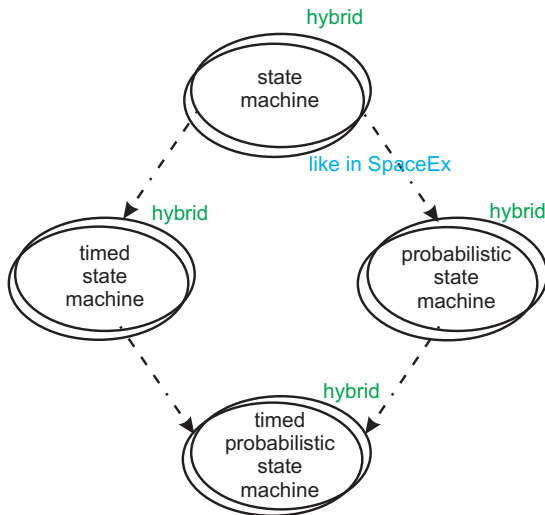
state
machine
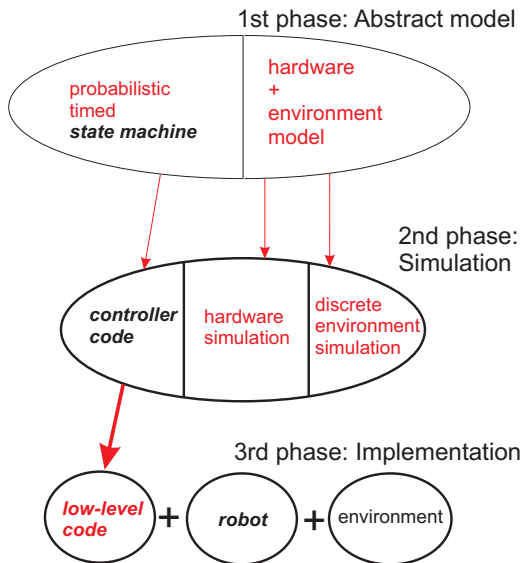
like in UML (subset)

How do we want to do it?

How do we want to do it?
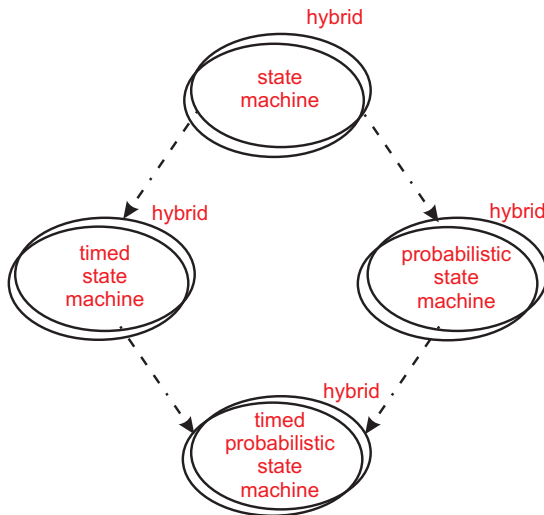
How do we want to do it?
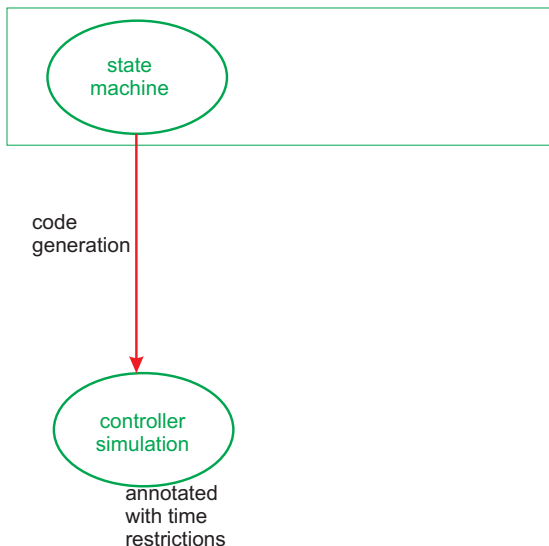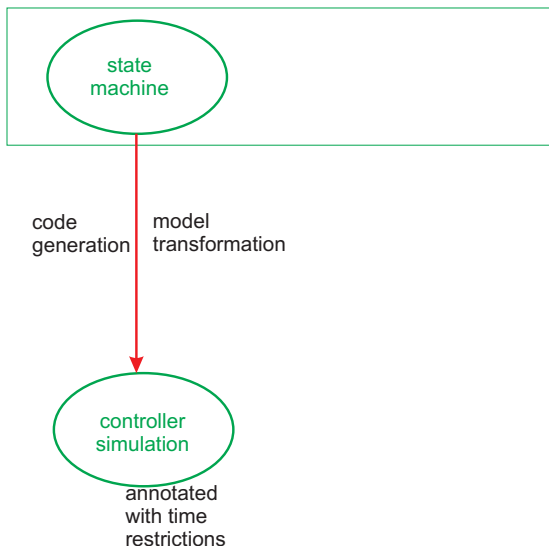
How do we want to do it?

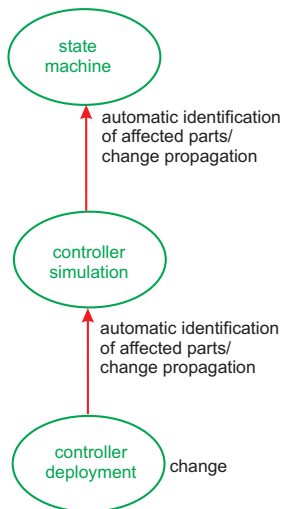# How do we want to do it?

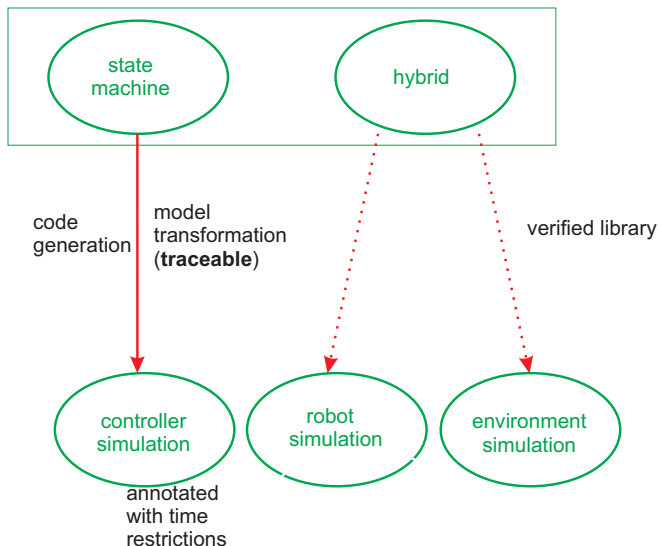# How do we want to do it?

And now to simulations and programs

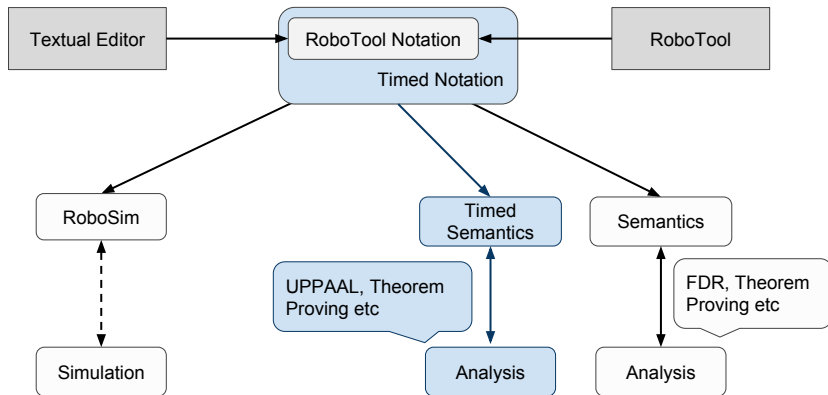# And now to simulations and programs

Traceability is very important.

And now to simulations and programs

In more detail

Plan of work

Five themes

- Core
- Time
- Probability: refinement is very challenging
- Environment:
    - no need for new simulation abstractions
    - code generation is very challenging

    abstractions
- Integration
    - theory of integration
    - simulation library
    - major case study: boats and drones

Plan of work

Five themes

- Core
- Time
- Probability: refinement is very challenging
- Environment:
    - no need for new simulation abstractions
    - code generation is very challenging

    abstractions
- Integration
    - theory of integration
    - simulation library
    - major case study: boats and drones

Conclusions

A lot to do

- Theory: UTP
- Practice: new languages (formal, diagrammatic, API), their implementations
- Verification: compositional, scalable, traceable

Our distinctive vision

- Notations akin to those already used
- Sound integration
- Full life cycle

The theory is that of cyber-physical systems.