RoboStar: Engineering Verified Software for Robotics

Jim Woodcock FREng

RoboStar — University of York, UK
www.cs.york.ac.uk/RoboStar

Ningbo, 10 May 2019

Thanks:
James Baxter, Ana Cavalcanti, Madiel Conserva, Simon Foster, Wei Li, Alvaro Miyazawa, Alexandre Mota, Pedro Ribeiro, Augusto Sampaio, Jon Timmis

ROYAL ACADEMY OF ENGINEERING

EPSRC
Engineering and Physical Sciences Research Council

Software Engineering for Robotics: why the interest?

UK's 8 Great Technologies

Robotics and Autonomous Systems — Policy Exchange

Innovate UK

> *"In the future, we will increasingly use RAS (Robotics and Autonomous Systems) to enhance almost every aspect of our lives."*
>
> *"Ensuring that RAS systems are able to make the right decisions, safely, effectively and efficiently in all likely circumstances requires a unique level of testing and functional validation."*

Full verification is beyond the state of the art.

Software Engineering of Robots: others are interested

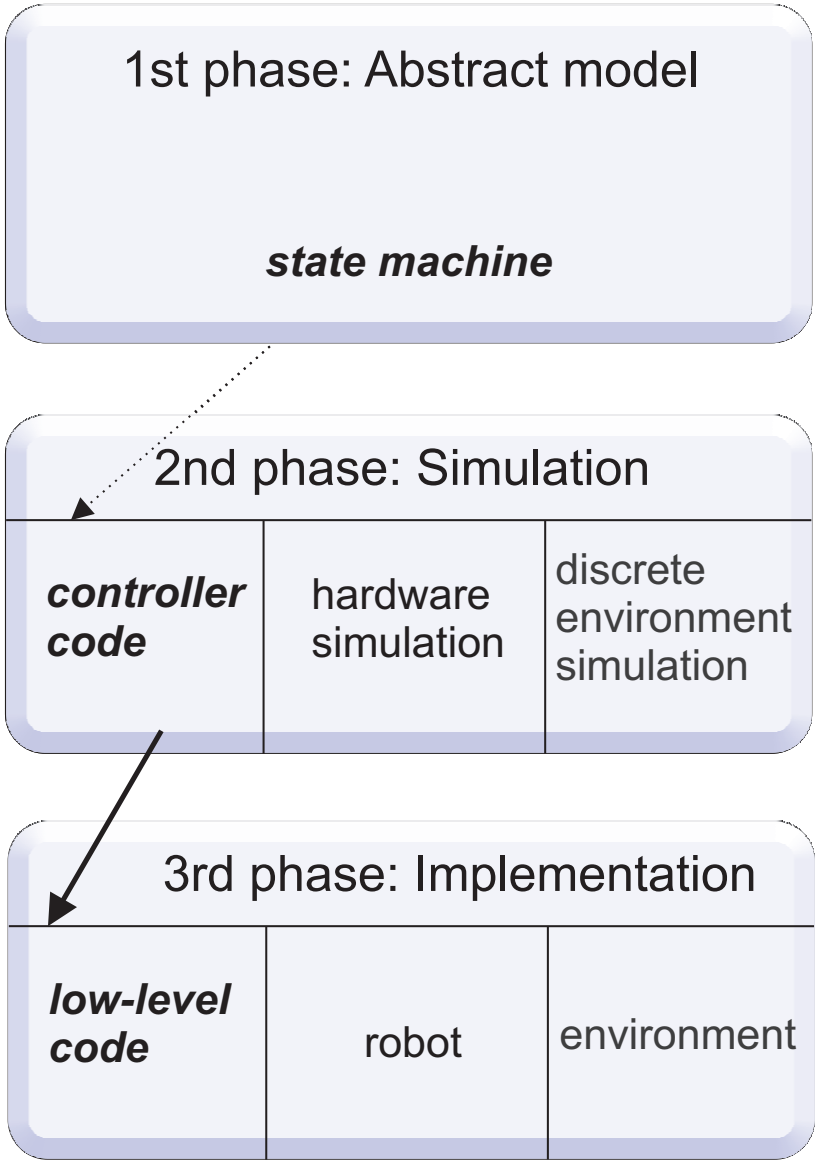
IROS 2016
special session

RoboChart in
IROS 2017

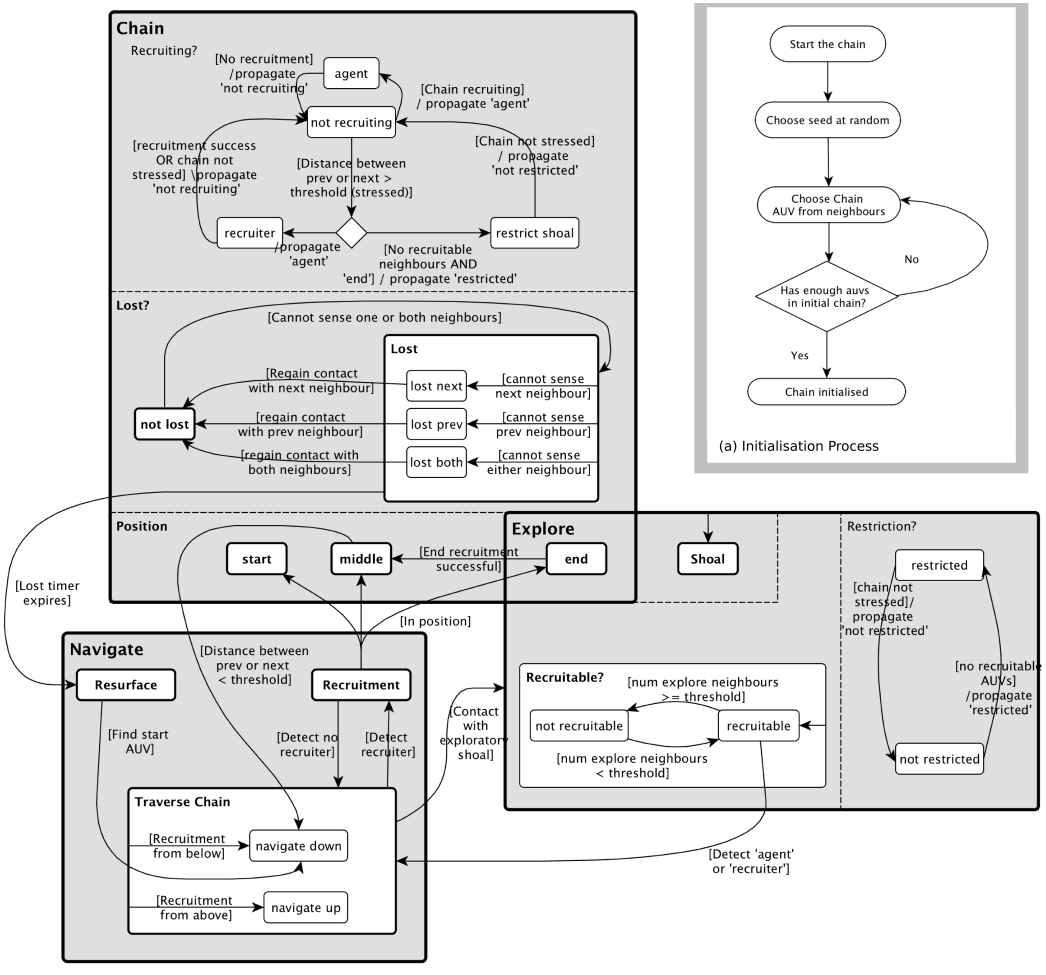
Our distinctive approach

- Complement existing techniques
- Use of domain-specific language
- Mathematics for validation and verification

Current approach to development

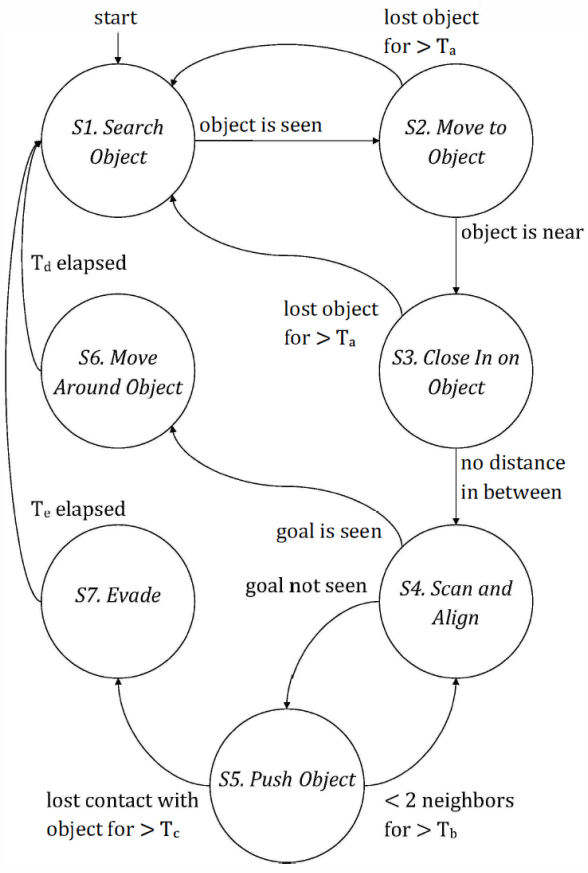


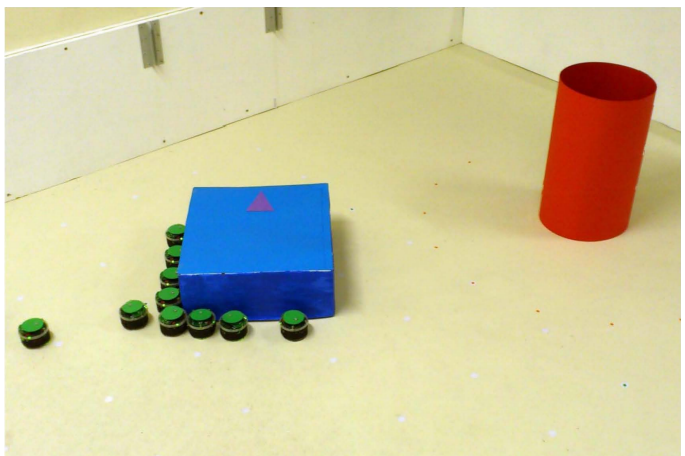
Example state machines



Becky Naylor, Mark Read, Jon Timmis, and Andy Tyrrell.
The Relay Chain: Communication between an Exploratory Underwater Shoal and a Surface Vehicle.
ALIFE 14: Proceedings of the 14th International Conference on the Synthesis and Simulation of Living Systems.

Example state machines

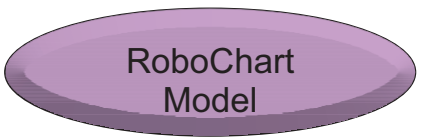




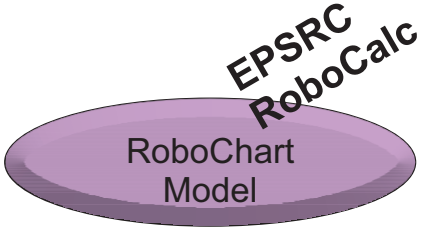
"A group of e-puck robots transporting an object (blue box) towards a goal (red cylinder)."

Jianing Chen, M. Gauci and R. Gross. "A strategy for transporting tall objects with a swarm of miniature mobile robots". In: Robotics and Automation (ICRA), 2013 IEEE International Conference on. 2013, pp. 863869.

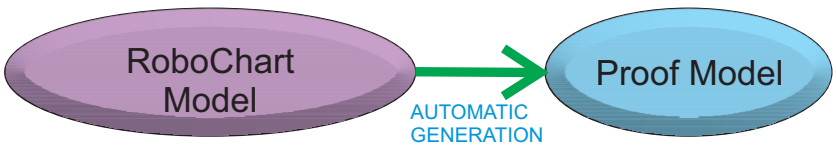
Our approach



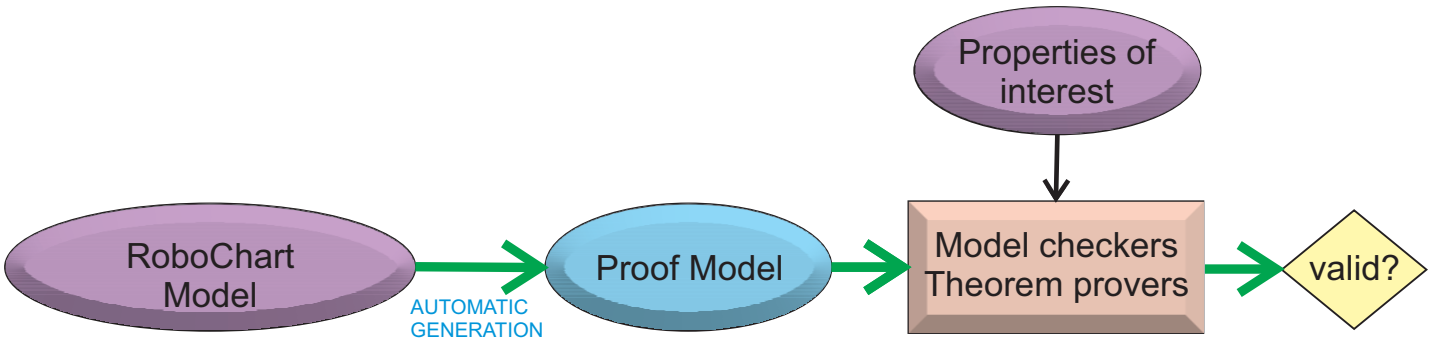
Our approach



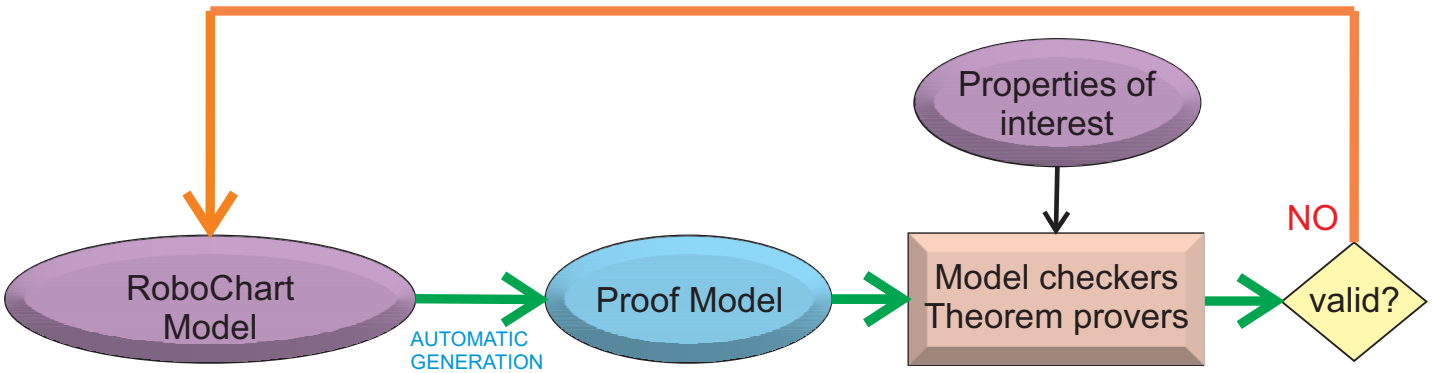
Our approach



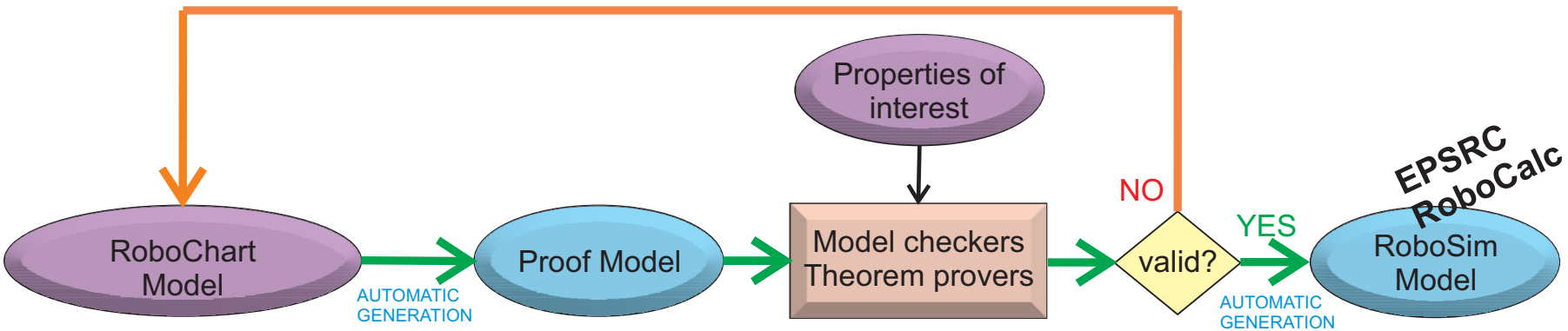
Our approach



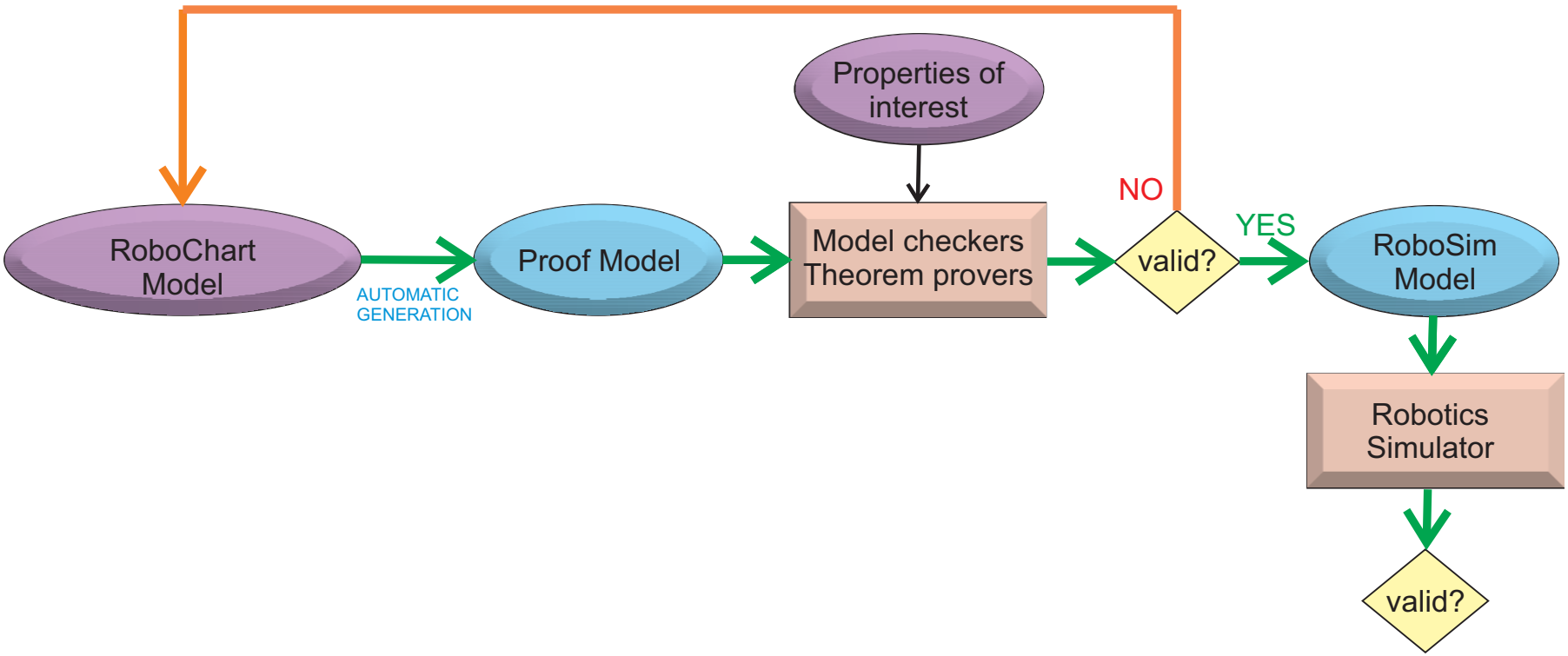
Our approach



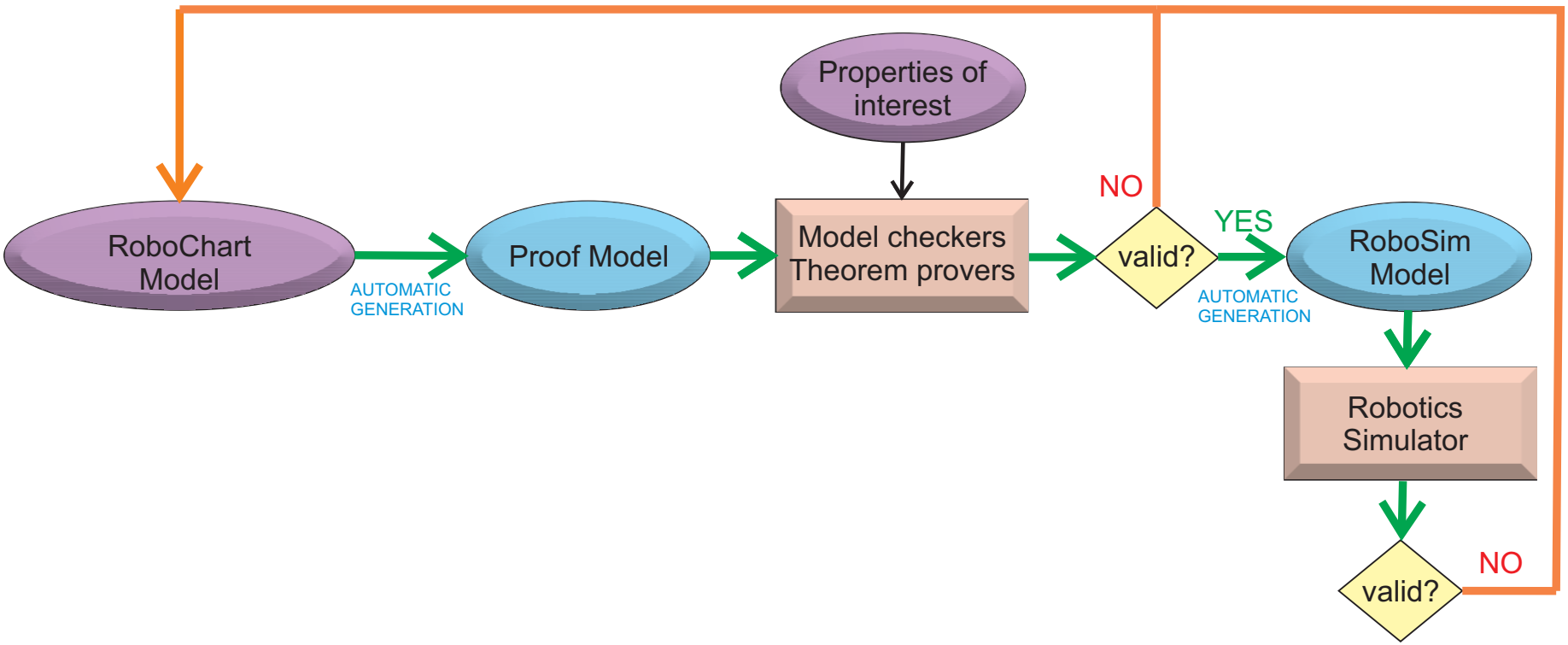
Our approach



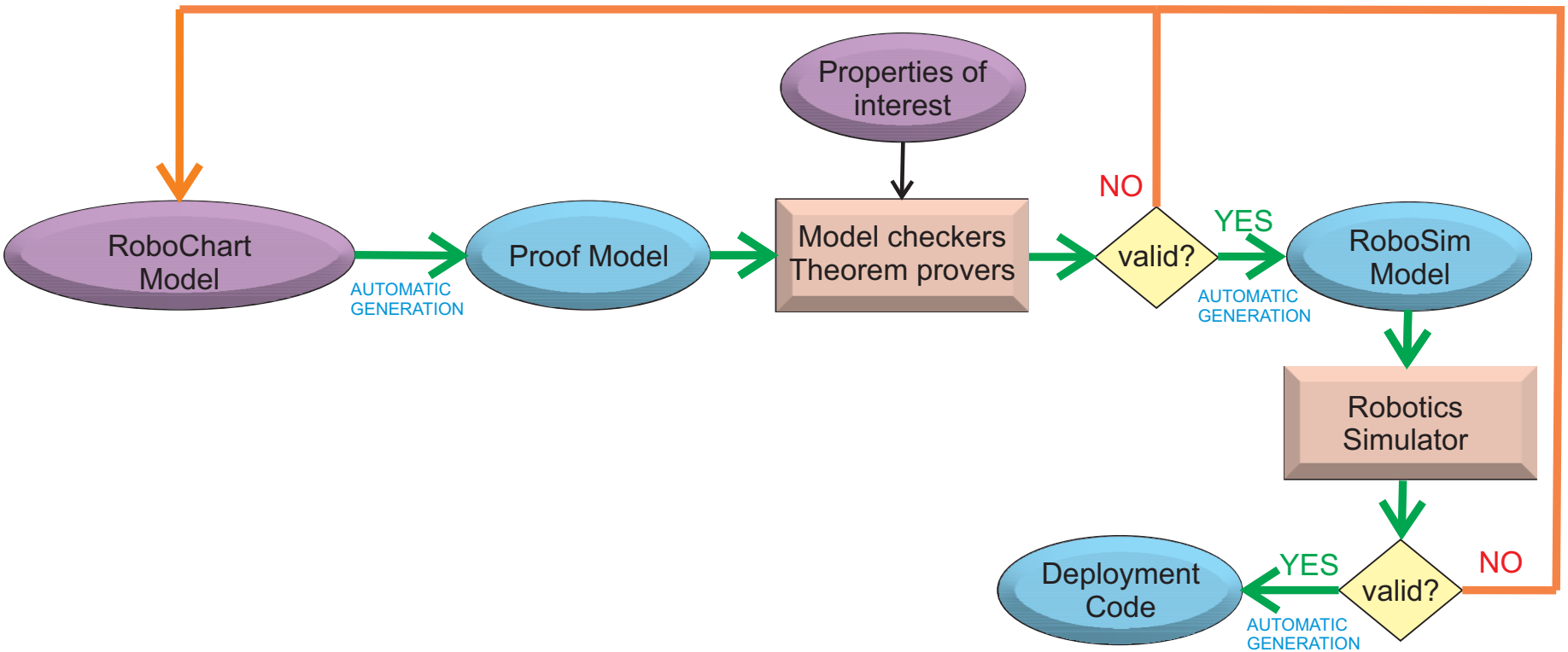
Our approach



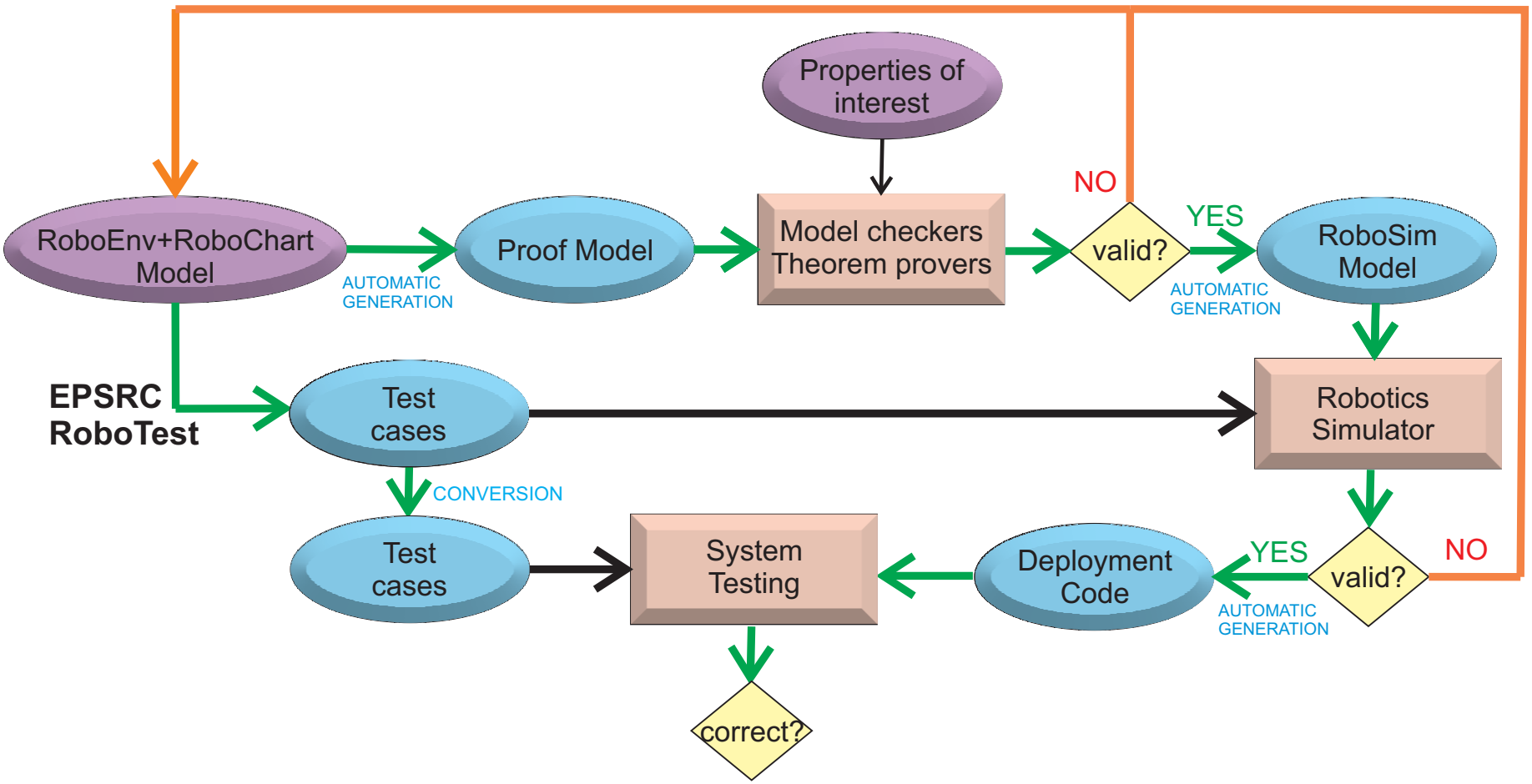
Our approach



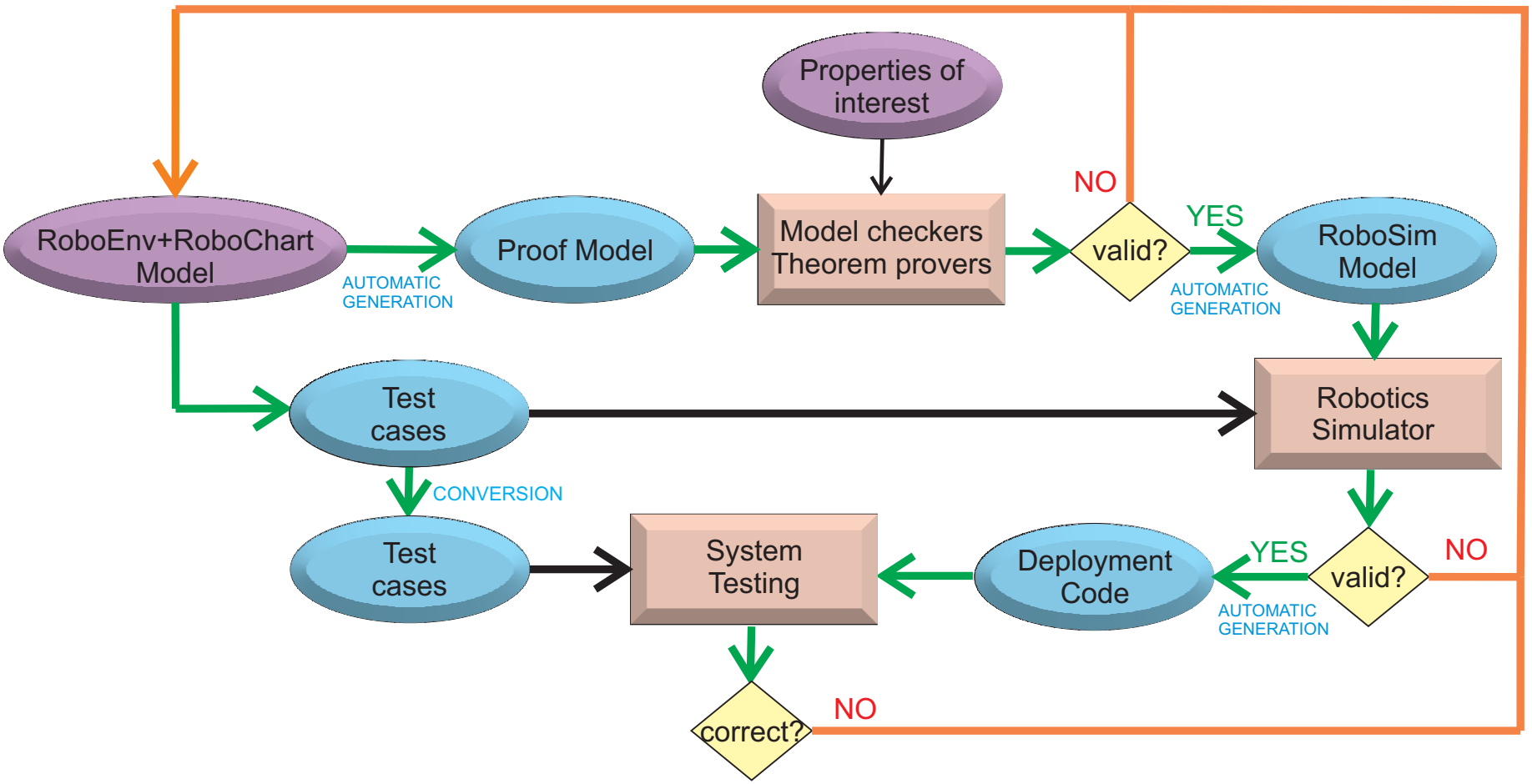
Our approach



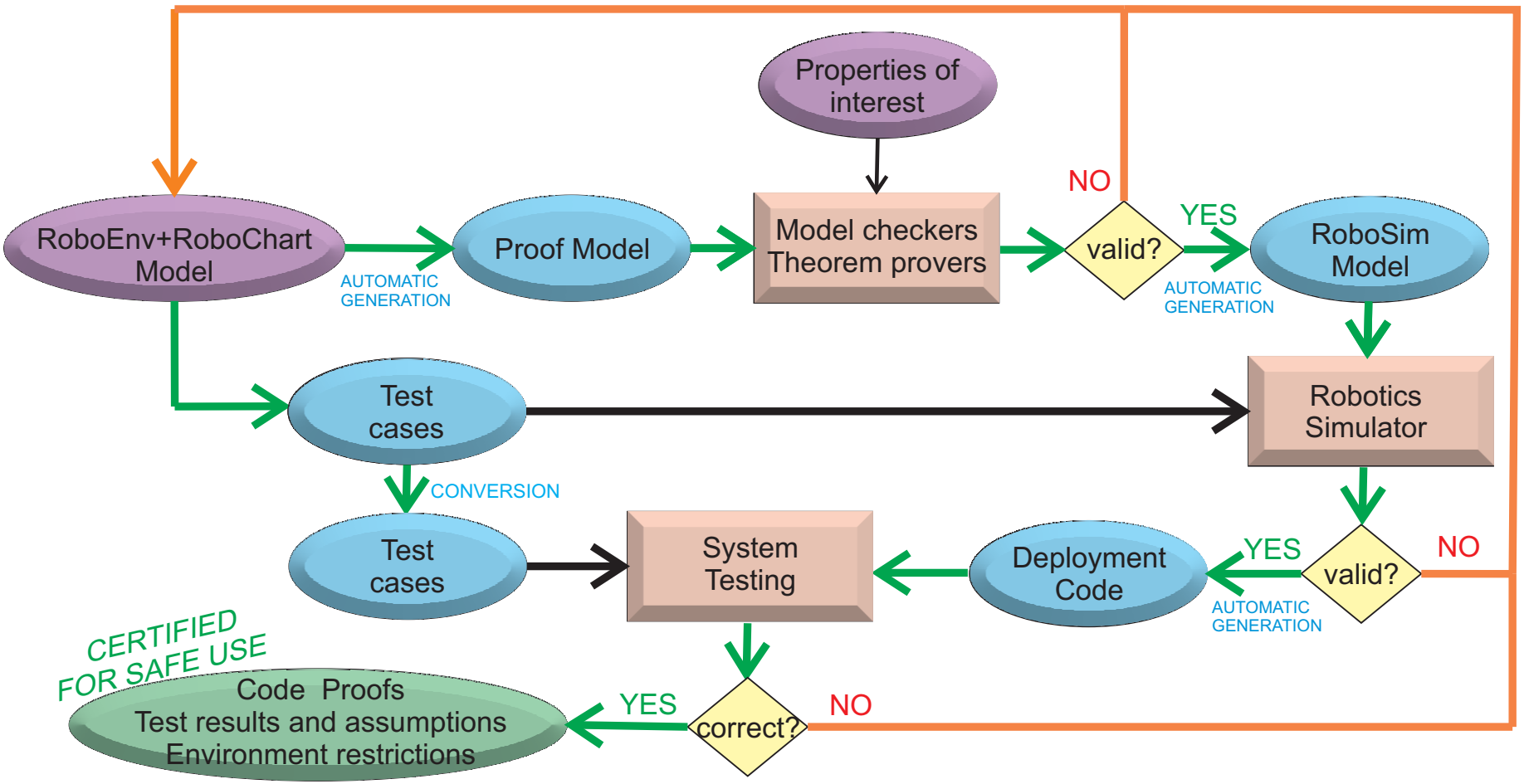
Our vision


Properties of interest
RoboEnv+RoboChart Model
AUTOMATIC GENERATION
Proof Model
Model checkers Theorem provers
NO
valid?
YES
AUTOMATIC GENERATION
RoboSim Model
EPSRC RoboTest
Test cases
Robotics Simulator
CONVERSION
Test cases
System Testing
Deployment Code
YES
valid?
NO
AUTOMATIC GENERATION
correct?

Our vision


Properties of interest
RoboEnv+RoboChart Model
AUTOMATIC GENERATION
Proof Model
Model checkers Theorem provers
NO
valid?
YES
AUTOMATIC GENERATION
RoboSim Model
Test cases
Robotics Simulator
CONVERSION
Test cases
System Testing
Deployment Code
YES
valid?
NO
AUTOMATIC GENERATION
correct?
NO

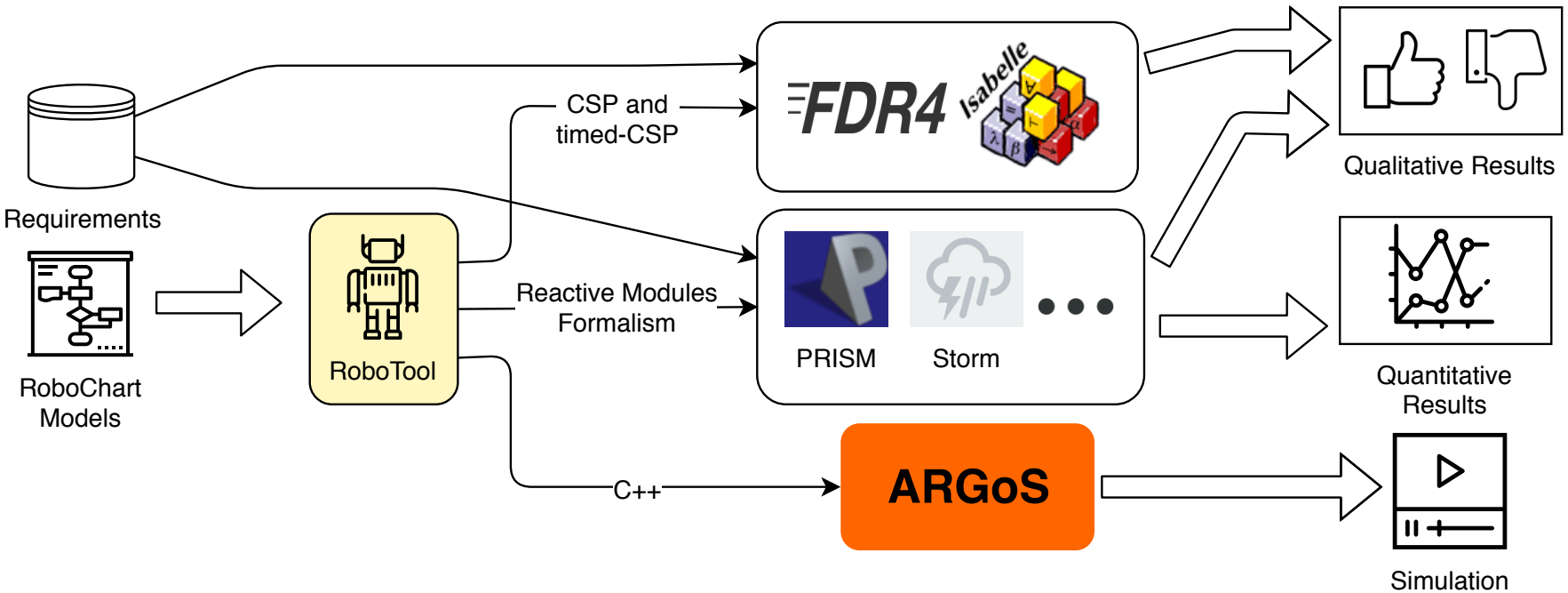
Our vision


Properties of interest
RoboEnv+RoboChart Model
AUTOMATIC GENERATION
Proof Model
Model checkers Theorem provers
NO
valid?
YES
AUTOMATIC GENERATION
RoboSim Model
Test cases
Robotics Simulator
CONVERSION
Test cases
System Testing
Deployment Code
YES
valid?
NO
AUTOMATIC GENERATION
CERTIFIED FOR SAFE USE
Code Proofs
Test results and assumptions
Environment restrictions
YES
correct?
NO

RoboChart

- Domain-specific modelling language for roboticists
- Based on UML, but restricted...
  - State machines
  - Component model
  - Physical robot explicitly modelled
- ...and enriched
  - real-time properties
  - probabilistic & stochastic properties
- Simplified compositional semantics and automated reasoning
  - Model checking bounded state
  - Theorem proving unbounded/infinite state
- Process algebraic semantics
- RoboTool: modelling, well formedness, ...

RoboTool


Requirements
RoboChart Models
RoboTool
CSP and timed-CSP
Reactive Modules Formalism
C++
FDR4
Isabelle
PRISM
Storm
ARGoS
Qualitative Results
Quantitative Results
Simulation

RoboTool: validation

- Examples modelled in RoboTool
  - Chemical detector
  - Transporter
  - Foraging
  - Swarm algorithms
  - Vacuum cleaner
  - Environmental sensing
- Generated semantics used for verification
- Large state-space for simple state-machines
- Compression functions highly effective

Chemical Detector

- Search for chemical spills
- Avoid obstacles
- Approach
- Drop flag

Property Checking

Properties

- Deadlock and livelock freedom
- Nondeterminism
- Refinement:
  - comparison of models, verification of requirements

Example requirements for chemical detector

1. Every gas reading should lead to resume/stop/turn.
2. Every command to move the robot (resume/stop/turn) leads to a reaction by the robot, before another command is issued.
3. If there is no gas, the chemical detector does not terminate.

Mathematical Models

- Formalised in a mathematical notation
- Covers core and timed notations
- Supports:
  - Automatic property checking using model checking
  - Semi-automatic property checking using theorem prover
- Model checking can prove requirements 1 and 3
- Requirement 2 fails, but provides counterexample

Robot Assurance Cases

Robot technology for critical systems

- decision making undesired real-word consequences
  - autonomous vehicles, surgical robot assistants, homecare robots, nuclear reactor robots
- how can we develop trust and regulatory acceptance?
- assurance cases with GSN and SACM
- foundational issues and automated support
- assurance case patterns
- comprehensive automated assurance framework
- supported by automated integrated formal methods
- certification tasks
  - cyber-physical systems, autonomous robots + digital twins

So, what next?

Summary

- Core notation design
- Mathematical definition
- Tool support
- Verified simulations

Currently

- Case studies: driverless pod, sandwich maker, ...
- Automatic translation from RoboChart to RoboSim
- Physical models for platforms
- Models for the environment
- Test generation
- ...

So, what next?

A lot to do

- Computer vision, artificial intelligence, human-robot interaction, ethics, ...
- Security, real-time analysis
- Software Engineering principles

Our distinctive vision

- Familiar notations
- Sound integration
- Full life-cycle

The theory is that of cyber-physical systems.