Personal Introduction / Position Statement

Frank Zeyda

Teesside University (UK)

4th December 2015 @ RoboCheck York

Personal Background

PhD at Teesside University (Reversible Computations in B)

EPSRC Projects at York 2007–2014 (post-doc)

1. "Programming from Control Laws" EP/E025366/1
2. "High-integrity Java Applications using Circus" EP/H017461/1

Current Position

Senior Lecturer in Computer Science
School of Computing, Teesside University (UK)

Contact Details

Email: f.zeyda@tees.ac.uk
Phone: +44 (0)1642 342784
Website: `https://www.scm.tees.ac.uk/users/f.zeyda/`

↪ Collaboration with York on "**A Calculus for Software Engineering of Mobile and Autonomous Robots**" (RoboCalc) EP/M025756/1

Personal Background

PhD at Teesside University (Reversible Computations in B)

Teesside University

EPSRC Projects at York 2007–2014 (post-doc)

1. "Programming from Control Laws" EP/E025366/1
2. "High-integrity Java Applications using Circus" EP/H017461/1

Current Position

Senior Lecturer in Computer Science
School of Computing, Teesside University (UK)

Contact Details

Email: f.zeyda@tees.ac.uk
Phone: +44 (0)1642 342784
Website: `https://www.scm.tees.ac.uk/users/f.zeyda/`

↪ Collaboration with York on "**A Calculus for Software Engineering of Mobile and Autonomous Robots**" (RoboCalc) EP/M025756/1

Personal Background

PhD at Teesside University (Reversible Computations in B)

EPSRC Projects at York 2007–2014 (post-doc)

1. "Programming from Control Laws" EP/E025366/1
2. "High-integrity Java Applications using Circus" EP/H017461/1

Current Position

Senior Lecturer in Computer Science
School of Computing, Teesside University (UK)

Contact Details

Email: f.zeyda@tees.ac.uk
Phone: +44 (0)1642 342784
Website: `https://www.scm.tees.ac.uk/users/f.zeyda/`

↪ Collaboration with York on "**A Calculus for Software Engineering of Mobile and Autonomous Robots**" (RoboCalc) EP/M025756/1

Personal Background

PhD at Teesside University (Reversible Computations in B)

EPSRC Projects at York 2007–2014 (post-doc)

1. “Programming from Control Laws” EP/E025366/1
2. “High-integrity Java Applications using Circus” EP/H017461/1

Current Position

Senior Lecturer in Computer Science
School of Computing, Teesside University (UK)

Contact Details

Email: f.zeyda@tees.ac.uk
Phone: +44 (0)1642 342784
Website: `https://www.scm.tees.ac.uk/users/f.zeyda/`

↪ Collaboration with York on “**A Calculus for Software Engineering of Mobile and Autonomous Robots**” (RoboCalc) EP/M025756/1

Personal Background

PhD at Teesside University (Reversible Computations in B)

EPSRC Projects at York 2007–2014 (post-doc)

1. "Programming from Control Laws" EP/E025366/1
2. "High-integrity Java Applications using Circus" EP/H017461/1

Current Position

Senior Lecturer in Computer Science
School of Computing, Teesside University (UK)

Contact Details

Email: f.zeyda@tees.ac.uk
Phone: +44 (0)1642 342784
Website: `https://www.scm.tees.ac.uk/users/f.zeyda/`

↪ Collaboration with York on "**A Calculus for Software Engineering of Mobile and Autonomous Robots**" (RoboCalc) EP/M025756/1

Personal Background

PhD at Teesside University (Reversible Computations in B)

EPSRC Projects at York 2007–2014 (post-doc)

1. "Programming from Control Laws" EP/E025366/1
2. "High-integrity Java Applications using Circus" EP/H017461/1

Current Position

Senior Lecturer in Computer Science
School of Computing, Teesside University (UK)

Contact Details

Email: f.zeyda@tees.ac.uk
Phone: +44 (0)1642 342784
Website: `https://www.scm.tees.ac.uk/users/f.zeyda/`

↪ Collaboration with York on "**A Calculus for Software Engineering of Mobile and Autonomous Robots**" (RoboCalc) EP/M025756/1

Research Interests

1. Verification via mathematical proof;
2. Refinement techniques and strategies;
3. Semantic foundations of languages;
4. Unifying Theories [of Programming] (UTP);
5. Theory mechanisation in theorem provers.

Research Interests (cont'd)

Why are these attractive for autonomous robots?

- Proof may address complexity issues due to state explosion;
  ↪ including support for infinite state spaces.
- Refinement is a transformational technique that facilitates piece-wise and step-wise verification:
  1. **piece-wise** verification embodies compositional reasoning;
  2. **step-wise** verification embodies incremental (agent) design.
- Unifying theories can provide a sound justification for the
  ↪ combination of verification methods and formalisms.
- Mechanisation is useful to strengthen certification evidence.

Research Interests (cont'd)

Why are these attractive for autonomous robots?

- Proof may address complexity issues due to state explosion;
  ↪ including support for infinite state spaces.
- Refinement is a transformational technique that facilitates piece-wise and step-wise verification:
  1. **piece-wise** verification embodies compositional reasoning;
  2. **step-wise** verification embodies incremental (agent) design.
- Unifying theories can provide a sound justification for the
  ↪ combination of verification methods and formalisms.
- Mechanisation is useful to strengthen certification evidence.

Research Interests (cont'd)

Why are these attractive for autonomous robots?

- Proof may address complexity issues due to state explosion;
  ↪ including support for infinite state spaces.
- Refinement is a transformational technique that facilitates piece-wise and step-wise verification:
  1. **piece-wise** verification embodies compositional reasoning;
  2. **step-wise** verification embodies incremental (agent) design.
- Unifying theories can provide a sound justification for the
  ↪ combination of verification methods and formalisms.
- Mechanisation is useful to strengthen certification evidence.

Research Interests (cont'd)

Why are these attractive for autonomous robots?

- Proof may address complexity issues due to state explosion;
  ↪ including support for infinite state spaces.
- Refinement is a transformational technique that facilitates piece-wise and step-wise verification:
  1. **piece-wise** verification embodies compositional reasoning;
  2. **step-wise** verification embodies incremental (agent) design.
- Unifying theories can provide a sound justification for the
  ↪ combination of verification methods and formalisms.
- Mechanisation is useful to strengthen certification evidence.

Research Interests (cont'd)

Why are these attractive for autonomous robots?

- Proof may address complexity issues due to state explosion;
  ↪ including support for infinite state spaces.
- Refinement is a transformational technique that facilitates piece-wise and step-wise verification:
  1. **piece-wise** verification embodies compositional reasoning;
  2. **step-wise** verification embodies incremental (agent) design.
- Unifying theories can provide a sound justification for the
  ↪ combination of verification methods and formalisms.
- Mechanisation is useful to strengthen certification evidence.

Research Interests (cont'd)

Why are these attractive for autonomous robots?

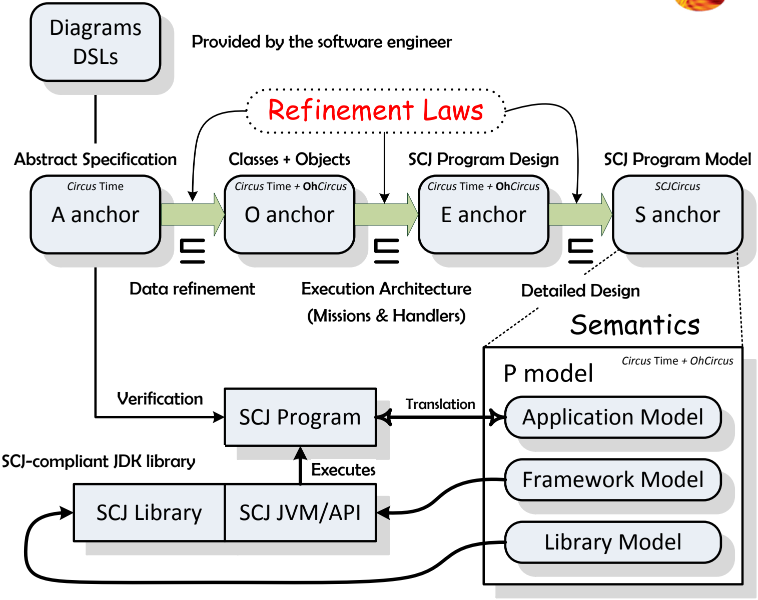
- Proof may address complexity issues due to state explosion;
  ↪ including support for infinite state spaces.
- Refinement is a transformational technique that facilitates piece-wise and step-wise verification:
  1. **piece-wise** verification embodies compositional reasoning;
  2. **step-wise** verification embodies incremental (agent) design.
- Unifying theories can provide a sound justification for the
  ↪ combination of verification methods and formalisms.
- Mechanisation is useful to strengthen certification evidence.

Research Interests (cont'd)

Why are these attractive for autonomous robots?

- Proof may address complexity issues due to state explosion;
  ↪ including support for infinite state spaces.
- Refinement is a transformational technique that facilitates piece-wise and step-wise verification:
  1. **piece-wise** verification embodies compositional reasoning;
  2. **step-wise** verification embodies incremental (agent) design.
- Unifying theories can provide a sound justification for the
  ↪ combination of verification methods and formalisms.
- Mechanisation is useful to strengthen certification evidence.

Refinement Example: Safety-Critical Java



A Personal View on Challenges

Teesside University

- Can we provide a formal methodology to justify abstractions and validate their suitability?
  - Perhaps use ideas from abstract interpretation and refinement.
- How do we ensure that high-level safety properties are preserved by the actual (physical) robot?
- Can we create compositional and incremental approaches to decompose and modularise verification effort?
- Can we provide a sound justification for combining evidence from model-checking, simulation and theorem proving?
  - The Unifying Theories of Programming may aid this endeavour.
- How do we deal with the complexity human behaviour?
  ↪ Perhaps take inspiration from rely-guarantee approaches...
- If absolute safety is out of reach, how to we formally quantitate a notion of sufficiently safe?

Contribution to Autonomous Robotics

The following is anticipated:

- Collaboration on the RoboCalc project (EP/M025756/1);
- EPSRC proposal next year to tackle open challenges (previous slide);
  ↪ A repository of challenges and problems would be useful.
- Contribute to the semantic integration and mechanisation of relevant **languages** and **logics** for autonomous agents and robots;
- Extension of the Isabelle/UTP prove tool to that end;
- Interaction and collaboration with the academic community in the area of autonomous robotics.

Being relatively new to this field, it would be great to interact, work and collaborate with you guys. ☺

Contribution to Autonomous Robotics

The following is anticipated:

- Collaboration on the RoboCalc project (EP/M025756/1);
- EPSRC proposal next year to tackle open challenges (previous slide);
  ↪ A repository of challenges and problems would be useful.
- Contribute to the semantic integration and mechanisation of relevant **languages** and **logics** for autonomous agents and robots;
- Extension of the Isabelle/UTP prove tool to that end;
- Interaction and collaboration with the academic community in the area of autonomous robotics.

Being relatively new to this field, it would be great to interact, work and collaborate with you guys. ☺