From Formalised State Machines to Implementations of Robotic Controllers

Wei Li¹, Alvaro Miyazawa², Pedro Ribeiro², Ana Cavalcanti², Jim Woodcock² and Jon Timmis¹

¹Department of Electronics, University of York, York, UK; ²Department of Computer Science, University of York, York, UK

1. Abstract

Safety is a major concern for autonomous robots, and the ability to provide evidence that a robotic system is safe can be demanding. Robotic controllers can be specified using state machines. However, these are typically developed in an ad hoc manner without formal semantics, which makes it difficult to analyse the controller. We present a state-machine based notation, RoboChart, together with a tool to automatically create code from the state machines, establishing a rigorous connection between specification and implementation. RoboChart has a formal semantics that allows for formal verification [1]. We demonstrate our approach using two case studies (self-organized aggregation and swarm taxis) in swarm robotics. This paves the way for the verification of controller of individual robots in the swarm as well as their resulting emergent behaviours.

2. RoboChart framework

Elements of RoboChart:
- Architecture for robotics
- State machine
- Interface
- Clock

Features of RoboChart:
- Graphical and textual modelling
- Formal semantics (CSP)
- Automatic code generation
- Platform independent

Automatic code generation:
- Model-View-Controller pattern
- Direct mapping from elements to entities in C++

3. Case study one: Aggregation

Behaviour: Robots aggregate into a single compact cluster as fast as possible.
Controller: two states; two events

Elements:
state machine: AggregationFSM; interface: Aggregationinterface; state: S1, S2; operation: MoveClockwise, RotateClockwise

Textual description:

<table>
<thead>
<tr>
<th>Transition</th>
<th>Source</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>S1</td>
<td>S2</td>
</tr>
<tr>
<td>T2</td>
<td>S2</td>
<td>S1</td>
</tr>
</tbody>
</table>

4. Case study two: Swarm taxis

Behaviour: Robots move towards a beacon while maintaining a coherent group
Controller: three states; one event; two conditions

5. Conclusions and future work

Summary:
- State-machine based robotic controllers can be modelled in RoboChart.
- Controller code can be automatically generated and integrated into different robotic platforms.
- Formal CSP semantics allows for the application of formal verification techniques[1].
- Gap between high-level reasoning and low-level implementation of robotic controller is reduced.

Future work:
- Verify individual controllers as well as their emergent swarm behaviour.
- Model probability and environmental stimuli.
- Generate code for implementation in physical robots.

Reference: