

Symbolic Reasoning Applied to Control Laws

April 2002

Richard Boulton, Ruth Hardy, Tom Kelsey, Ursula Martin

School of Computer Science
University of St Andrews

North Haugh
St Andrews
Fife KY16 9SS
Scotland, UK

Work funded by QinetiQ

Outline

- Background, objectives, and approach
- A bit of control theory
- Some properties of control systems and reasoning about them
- Composable properties
- Summary

Machine-Assisted Reasoning for Computational Mathematics

- Machine-assisted reasoning produces formal proofs in a rigorous logical system
 - e.g. HOL (Cambridge), PVS (SRI), ProofPower (Lemma 1)
 - applied by Intel, Motorola, QinetiQ to critical applications
- Our goal:
 - machine-assisted reasoning to enhance computational mathematics
 - use familiar informal systems (Maple, MATLAB, NAG) as normal
 - generate proof obligations for theorem prover to increase
 - * assurance
 - * scope

Some Approaches

Maple-PVS

- use computer algebra system to compute and experiment
- pass conjectured result to PVS theorem prover

proof obligation: computed result

Larch-AXIOM (sponsored by NAG Ltd)

- “smart comments”: annotate legacy numerics components with assertions
- use in verification, program synthesis, method selection

proof obligation: property of computed result

Verified table look-up

- theorem prover decides which case of symbolic table holds

proof obligation: satisfiability of symbolic constraints

Formal Methods for Control Engineering

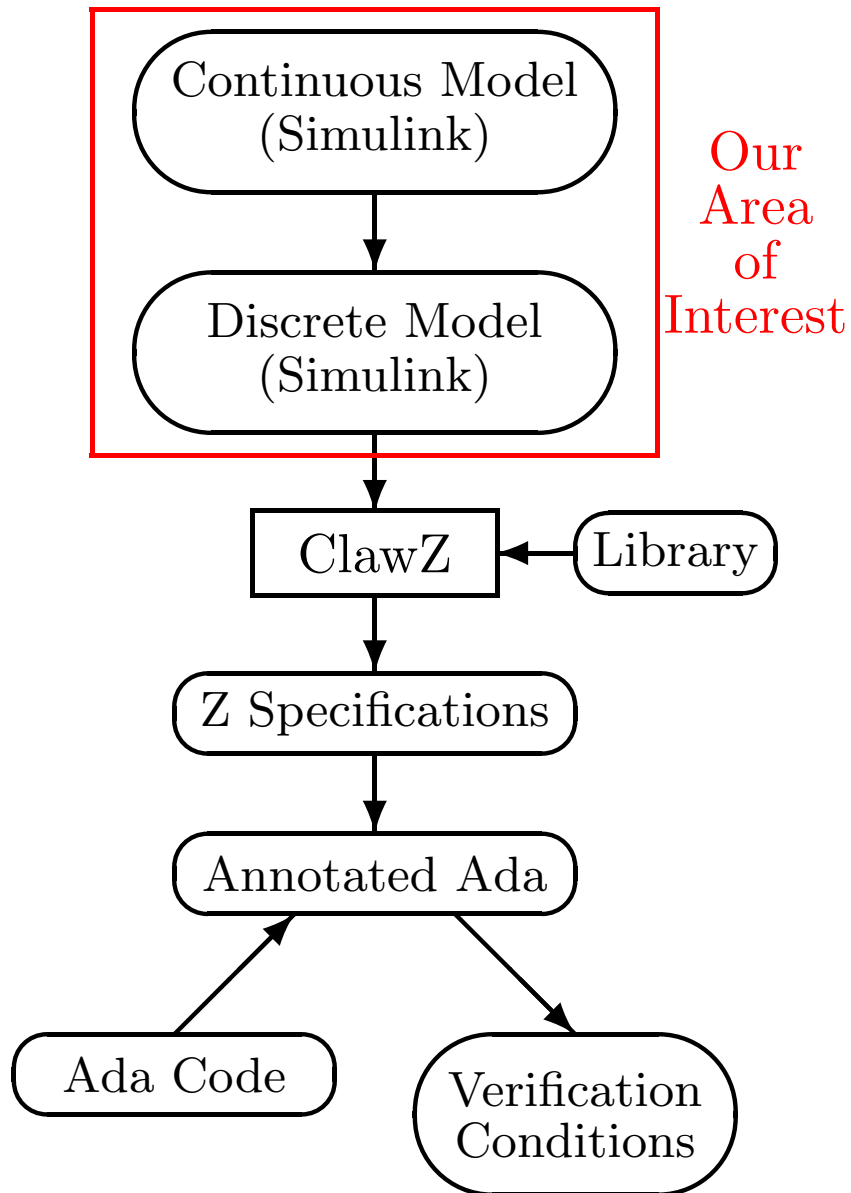
Work at QinetiQ

- Formal verification of critical systems
- ClawZ: Simulink models \rightarrow Z specifications
- Compliance Tool for annotated Ada code
 - verification conditions: establish using a proof assistant
- Want additional acceptance criteria

Variety of other approaches

- DARPA Mobies project (CMU, Berkeley, SRI, Ford, ...)
- Hybrid systems, model as state machine

ClawZ: Control Laws in Z



ClawZ covers correctness of SPARK Ada code with respect to discrete-time Simulink model.

But what can we say formally about the model(s)? ...

Objectives and Approach

Objective:

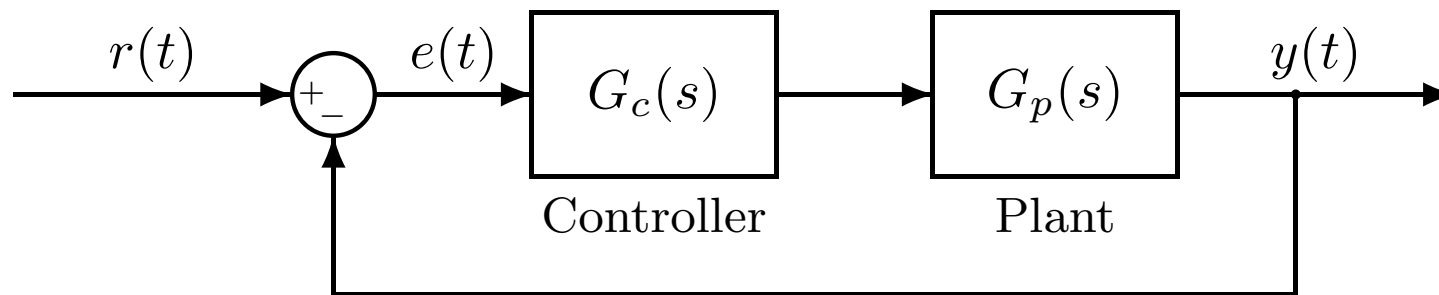
- Investigate how machine-assisted symbolic reasoning might be applied in control engineering

Approach:

- Determine possible proof obligations
 - e.g. bounds on values of amplitude and phase shift
- Study feasibility of machine-assisted reasoning
- Use combination of
 - MATLAB
 - computer algebra systems (e.g. Maple)
 - proof assistants (e.g. ProofPower)

Control Theory

Transfer function: Laplace transform of function of time



Open-loop transfer function (no feedback) (s complex):

$$G_c(s)G_p(s)$$

Closed-loop transfer function:

$$\frac{Y(s)}{R(s)} = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)}$$

Features to be Analysed

- Rise-time, damping, steady-state behaviour
- Stability: steady-state response decays or remains bounded
 - analyse using a step-function input
- Frequency response: gain (amplitude) and phase shift
 - analyse using a sinusoidal input
- Handling properties of system

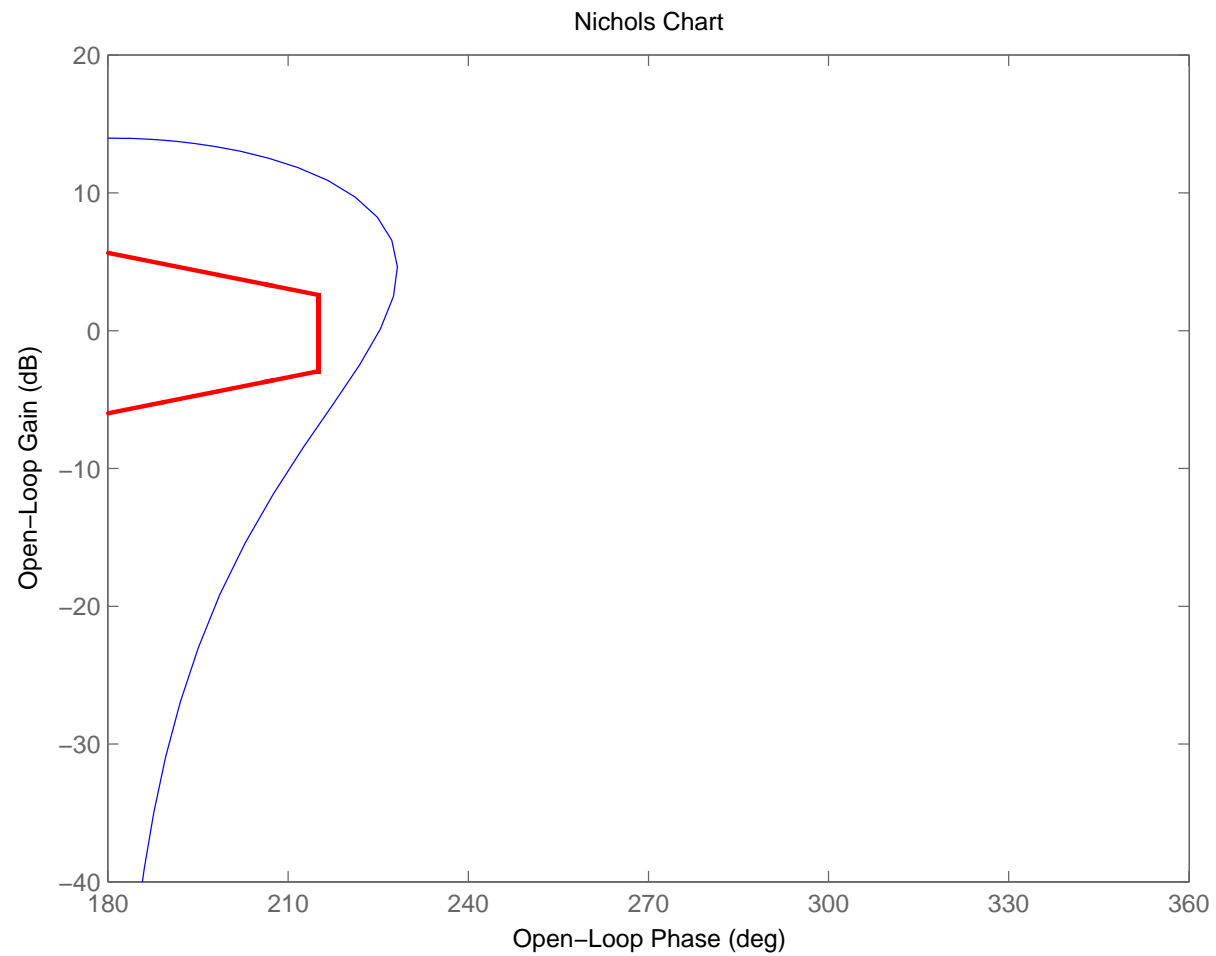
Stability

- Determined by position of roots of denominator of closed-loop transfer function
- Conditions for absolute stability:
 - Continuous: all roots must lie in left half of complex plane
 - Discrete: all roots must lie within the unit circle
- Do not need to find the roots of the polynomials
 - Can determine from coefficients whether conditions hold
- Relationship between stability conditions for continuous-time and discrete-time models
 - We have investigated this symbolically in terms of controller parameters

Frequency Response

- For linear systems, sinusoidal input yields sinusoidal output
- Analyse amplitude (gain) and phase shift of output w.r.t. input
- Bode diagram: separate plots of gain vs freq. and phase vs freq.
- Nichols plot: single plot of gain vs phase as frequency varies
- Logarithmic scale used for amplitude and frequency
- Bounds on shape and path of plots used to ensure good handling
- We want to represent these bounds symbolically

Example: Nichols Plot



Exploiting Composability

- Real-world control systems are large
 - Transfer function for whole system may involve high-order polynomials
- Where possible, look for properties that are composable
 - i.e. property of system can be obtained from properties of components
- Gain and phase appear to be two such properties
 - Gains are multiplicative and phases are additive
- Construct some kind of calculus around these?
- ClawZ might provide infrastructure to mechanise this

Summary

- Symbolic representations for stability criteria and bounds on frequency response
- Mainly using a computer algebra system (Maple)
- Aim to check properties by mechanised formal proof
- Control theory uses complex numbers
 - but proof obligations may not have to involve them
- Adding assertions to Simulink and/or ClawZ
 - ClawZ infrastructure allows alternative Z specs
- Seeking composable properties where possible