Reasoning about Linear Systems

Rob Arthan

Lemma 1 Ltd./Dept. of Computer Science. Queen Mary, London

With

Ursula Martin, Paulo Oliva, Erik Arne Mathiesen

Dept. of Computer Science. Queen Mary, London

ARG, Cambridge

13th November, 2007

References and Links

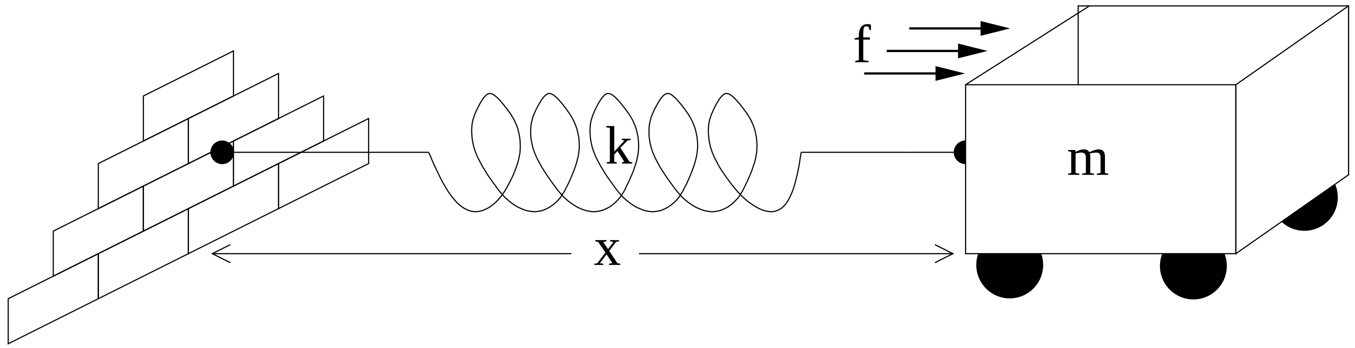
- *Reasoning About Linear Systems*, this paper, SEFM 2007.
- *Hoare Logic in the Abstract*, Oliva, Mathiesen & Martin, CSL 2006.
- *Formal Methods for Control Engineering: a Validate Decision Procedure for Nichols Plot Analysis*, Hardy, Ph. D. Thesis, St. Andrews, 2006 (also *Applications of Real Number Theorem Proving in PVS*, Gottliebsen, Hardy, Lightfoot & Martin, to appear).
- *A Hoare Logic for Single-Input Single-Output Continuous-Time Control Systems*, Boulton, Hardy & Martin, HSCC, 2003.
- *ClawZ: Control Laws in Z*, Arthan, Caseley, O'Halloran & Smith, ICFEM 2000.
- *ClawZ — The Semantics of Simulink Diagrams*, Jones, Lemma 1 Report, 1999.
- See 'Papers' and 'ClawZ' pages at `http://www.lemma-one.com`

Backgrounds and Aims

- Apply computational logic to specification and verification of systems with continuous data/time.
- Applications domains, e.g., avionics control systems
- Existing tools (e.g., Simulink):
  - Good for: numerical calculation, simulation
  - No help with: logical reasoning, verification of properties
- Reuse ideas from software specification and verification
- Modular, scalable approach building on proven ideas

Models of Control Systems I

- A physical system:



- Laws of mechanics give an equational model:

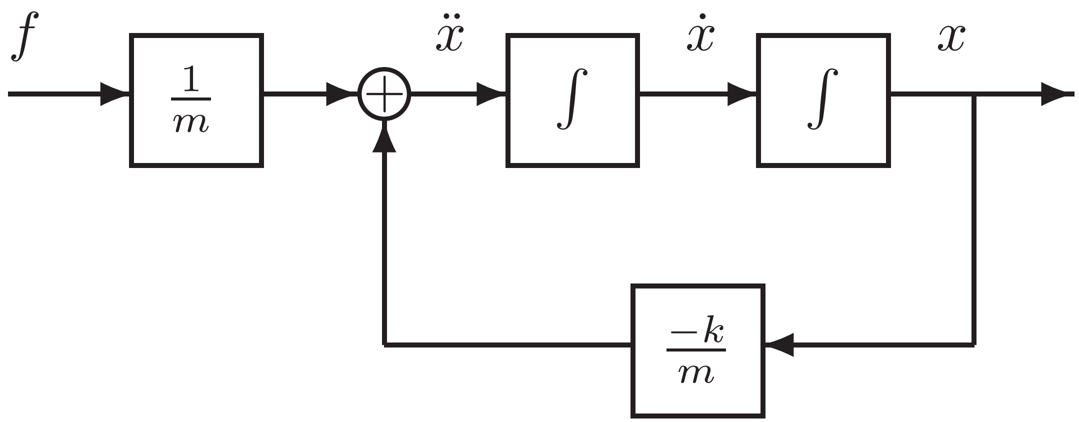
$$m\ddot{x}(t) + kx(t) - f = 0$$

Models of Control Systems II

- Equational model has no intensionality:

$$m\ddot{x}(t) + kx(t) - f = 0$$

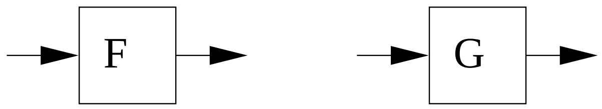
- Block diagram model gives structure and intensionality:



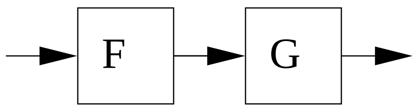
- Can be viewed in many ways: e.g., as a design for an analogue computer

New Systems From Old — Structured Block Diagrams

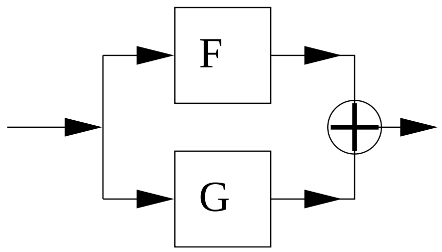
- Starting from existing systems:



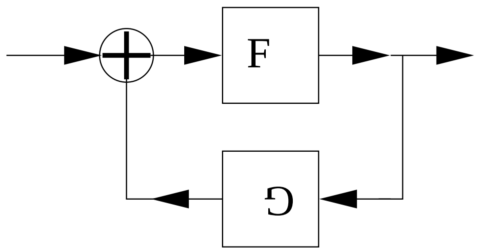
- Construct new systems using standard constructions:



Sequence



Sum

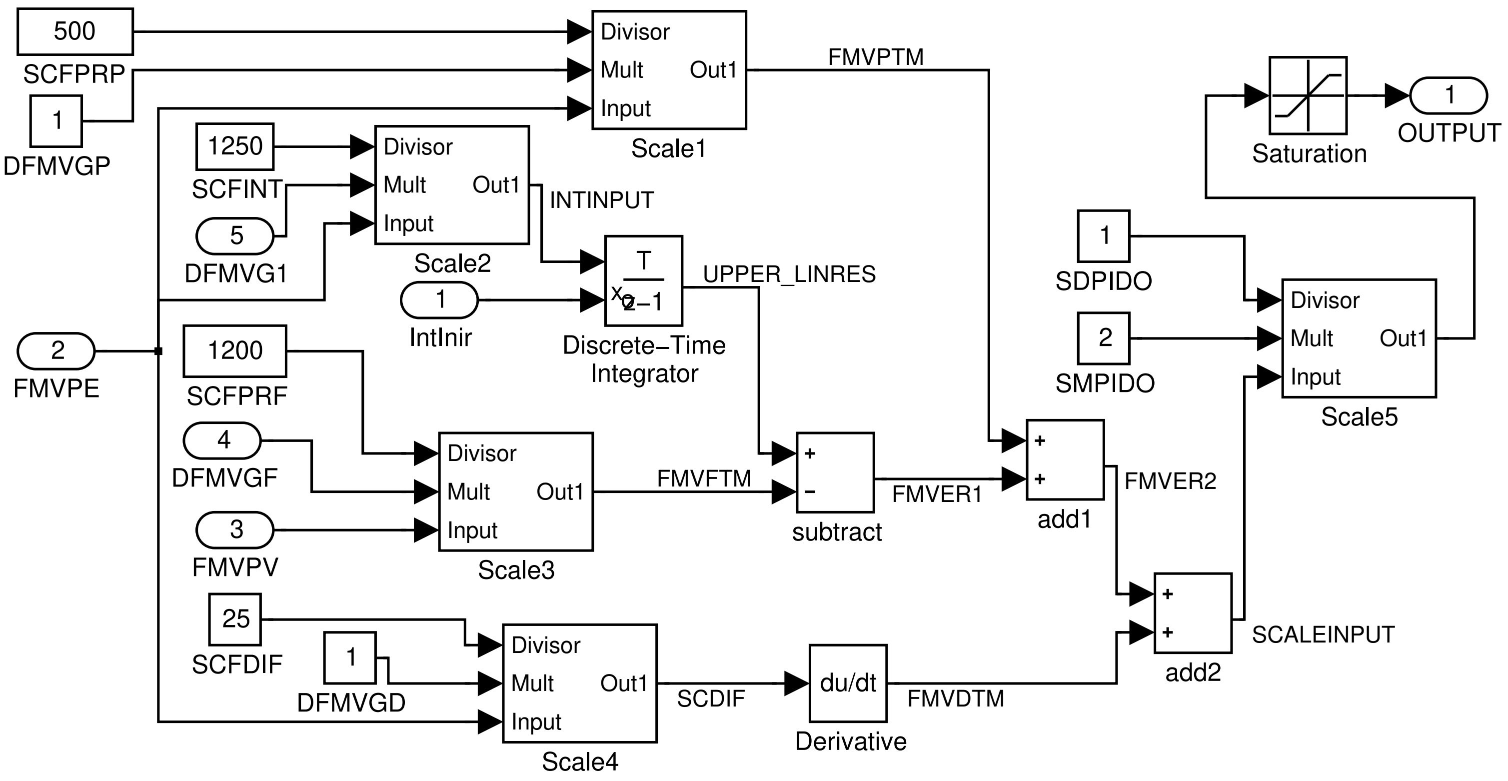


Feedback Loop

- Modular approach is good for scalability

Unstructured Diagrams — Signal Flow Graphs

Real-life diagrams are often less structured:



Linear Algebra

- A (real) vector space V is:
  - A commutative group: $(\mathbf{v}, \mathbf{w}) \mapsto \mathbf{v} + \mathbf{w}$, $\mathbf{v} \mapsto -\mathbf{v}$,
  - acted on by elements of $\mathbb{R}$, $(\lambda, \mathbf{v}) \mapsto \lambda\mathbf{v}$,
  - where $\lambda \in \mathbb{R}$ acts as a homomorphism: $\lambda(\mathbf{v} + \mathbf{w}) = \lambda\mathbf{v} + \lambda\mathbf{w}$
  - and $\_ \times \_$ on $\mathbb{R}$ is composition of actions: $(\lambda\mu)\mathbf{v} = \lambda(\mu\mathbf{v})$
- Linear transformations, $f : V \to W$:
  - satisfy $f(\mathbf{v} + \mathbf{w}) = f(\mathbf{v}) + f(\mathbf{w})$ and $f(\lambda\mathbf{v}) = \lambda f(\mathbf{v})$,
  - have kernels, $\mathsf{ker}(f) = \{\mathbf{v} \mid f(\mathbf{v}) = \mathbf{0}\}$
  - and factor as: $f = (g : V \twoheadrightarrow V/\mathsf{ker}(f)); (h : V/\mathsf{ker}(f) \rightarrowtail W)$

Linear Systems

- Semantic value of a system is its input/output relation
- Many possible mathematical domains to model signals
- We consider **linear systems**. I.e., systems where:
  - signals are elements of vector spaces over $\mathbb{R}$
  - blocks are linear transformations between spaces
  - edges represent equational constraints
- semantic value of a diagram is an **additive relation**

Additive Relations

- A relation $r : V \leftrightarrow W$ between vector spaces is **additive** iff. r is a subspace of the product space $V \times W$

  E.g., a linear transformation or its inverse, or a composition:

$$f_1; f_2^{-1}; f_3; f_4^{-1}; \ldots$$

- Like a linear transformation, an additive relation has a kernel:

$$\mathsf{ker}(r) = \{v : V \mid v \;\underline{r}\; 0\}$$

  A uniform measure of information loss.

- An additive relation also has an indeterminacy:

$$\mathsf{ind}(r) = \{w : W \mid 0 \;\underline{r}\; w\} = \mathsf{ker}(r^{-1})$$

  A uniform measure of non-determinism.

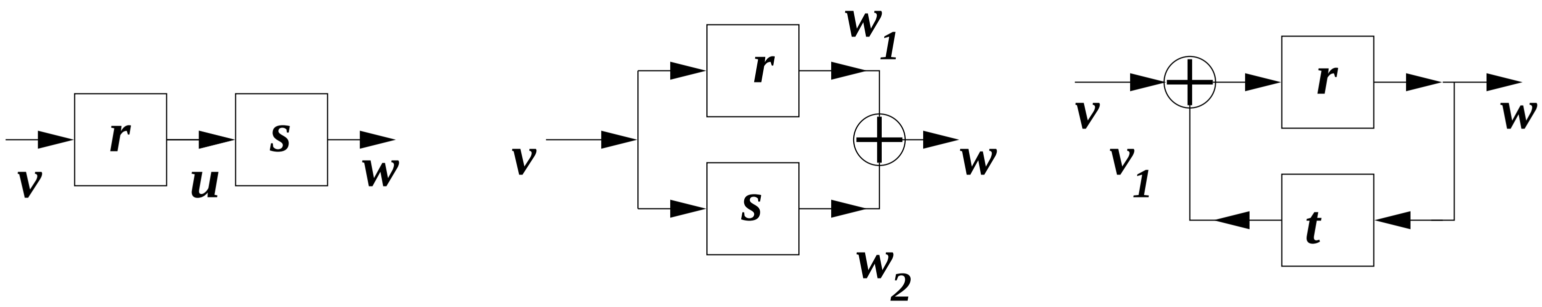
New Additive Relations From Old

Diagram constructors correspond to operators on additive relations:

Sequence: $v\ \underline{(r;s)}\ w$ iff. $\exists u \bullet v\ \underline{r}\ u \wedge u\ \underline{r}\ w$

Sum: $v\ \underline{r+s}\ w$ iff. $\exists w_1, w_2 \bullet v\ \underline{r}\ w_1 \wedge v\ \underline{s}\ w_2 \wedge w = w_1 + w_2$

Loop: $v\ \underline{\mathsf{loop}(r,t)}\ w$, iff. $\exists v_1 \bullet w\ \underline{t}\ v_1 \wedge (v + v_1)\ \underline{r}\ w$



WARNING: sum is associative, commutative and has a 0, but:

$$\exists r, s, t \bullet r + s = t + s \not\Rightarrow r = t$$

$$\exists r \bullet r - r \neq 0$$

Feedback Loops and Relational Inverse

- The feedback loop is definable in terms of relational inverse:

$$\mathsf{loop}(r,t) = (r^{-1} - t)^{-1},$$

- ...and vice versa: if $r : V \leftrightarrow V$ is additive:

$$r^{-1} = (1_V^{-1} - (1_V - r))^{-1} = \mathsf{loop}(1_V, 1_V - r)$$

And, in general, if $r : V \leftrightarrow W$, r^{-1} is the composite:

$$\begin{array}{rcl} (0, 1_W) & : & W \to V \times W; \\ \mathsf{loop}(1_{V\times W}, 1_{V\times W} - (\pi_1; r; (0, 1_W))) & : & V \times W \leftrightarrow V \times W; \\ \pi_1 & : & V \times W \to V \end{array}$$

where $\pi_1 : V \times W \to V$ and $\pi_2 : V \times W \to W$ are the projections.

Completeness of Structured Block Diagrams

Assume given all linear transformations as basic blocks.

Theorem 1 *Every additive relation is the semantic value of some structured block diagram.*

Proof. An additive relation $r : V \leftrightarrow W$, factors as $f^{-1}; g; h^{-1}$ where for some A, B, $f : A \rightarrowtail V$, $g : A \twoheadrightarrow B$, and $h : W \twoheadrightarrow B$.

Corollary 2 *Every signal flow graph is equivalent to some structured block diagram.*

Proof. The semantic value of an unstructured diagram is an additive relation. Apply the theorem to that additive relation.

Cf. `while`-programs are Turing complete.

Weakest pre-conditions

- As usual, for $r : X \leftrightarrow Y$, $B \subseteq Y$, define

$$\mathsf{wp}(r, B) := \{x : \mathsf{dom}(r) \mid \forall y : Y \bullet x \ \underline{r}\ y \Rightarrow y \in B\}$$

- W.P. for a function is very simple: if $f : X \rightarrow Y$,

$$\mathsf{wp}(f, B) = Bf^{-1}$$

- W.P. for additive relations are nearly as simple:

$$\mathsf{wp}(r, B) = B_0 r^{-1}$$

  where

$$B_0 = \{b : B \mid b + \mathsf{ind}(r) \subseteq B\}$$

The Hoare Logic

- Linear combination rule:

$$\frac{\{A\} \quad r \quad \{B\} \qquad \{A\} \quad s \quad \{B_1\}}{\{A\} \quad \beta r + \gamma s \quad \{\beta B + \gamma B_1\}}$$

- Inverse rule:

$$\frac{\{A\} \quad r \quad \{B\}}{\{B\} \quad r^{-1} \quad \{A + \mathsf{ker}(r)\}} \; B \subseteq Ar$$

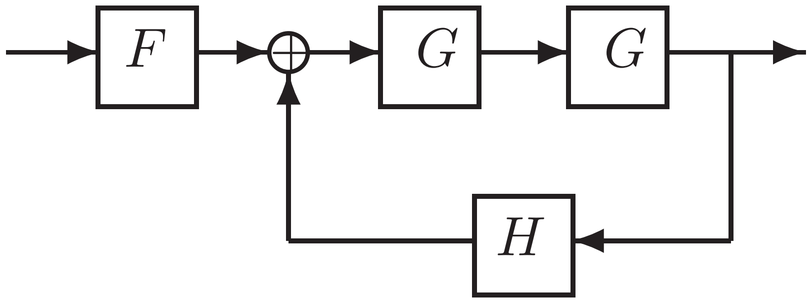
- Sequence rule:

$$\frac{\{A\} \quad s \quad \{B\} \qquad \{B\} \quad t \quad \{C\}}{\{A\} \quad s; t \quad \{C\}}$$

- Loop rule derivable from inverse rule (see paper).

State Space Representation

- Taking u, $\dot{u}$ and $\ddot{u}$ as states to represent a signal, u, the spring and cart system is a 3-dimensional linear system:



where:

$$F = \begin{pmatrix} \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & \frac{1}{m} \end{pmatrix} \quad G = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \quad H = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -\frac{k}{m} & 0 & 0 \end{pmatrix}$$

Specification and Proof Automation

- Envisage assertions expressed say in first-order real arithmetic, possibly restricting to linear arithmetic.
- Language is expressive but decidable
- Weakest pre-conditions can be calculated automatically.
- Automatically prove properties: E.g.,

$$\{\ddot{x} > c\} \quad \mathsf{SpringAndCart} \quad \{f > (m + k)c\}$$

Specification Language Issues

- Trade off between expressiveness and complexity/possibility of decision procedures
- Full first-order theory of vector spaces reduces to the theory of a real closed field (Solovay)
- Real closed field decision procedure only just practical with current state of the art (MacLaughlin, Harrison)
- There are "linear" theories of linear algebra that reduce to the theory of linear real arithmetic (Solovay, Arthan)
- Linear arithmetic with rational coefficients is widely implemented (e.g., in all the HOLs)
- But what about more general/expressive fields of reals?

Generality of Hodes-Fourier-Motzkin Procedure

- Recall the Fourier-Motzkin elimination step: if some $a_{i1} \neq 0$,

$$a_{11}x_1 + \ldots + a_{1n}x_n < b_1 \wedge \ldots \wedge a_{m1}x_1 + \ldots + a_{mn}x_n < b_m$$

$$\text{iff}$$

$$L_1 < x_1 \wedge L_2 < x_1 \wedge \ldots \wedge x_1 < U_1 \wedge x_1 < U_2 \wedge \ldots$$

$$\text{iff}$$

$$L_1 < U_1 \wedge L_1 < U_2 \wedge \ldots \wedge L_2 < U_1 \wedge L_2 < U_2 \wedge \ldots$$

- Works over *any* ordered field with decidable ground formulae
- Engineers will want $\sqrt{2}$ and likely more: e, π, etc.
- What are the options?

Zero-th Order Real Number Theorem Proving

- $\mathbb{Q}$, $\mathbb{Q}[\sqrt{p/q} \mid p, q \in \mathbb{N}, q \neq 0]$ — easy
- $\mathcal{A} = \mathbb{Q}[\mathsf{RealRoots}(f) \mid f \in \mathbb{Q}[X]]$
  the ring of all algebraic numbers – doable, hard-ish
- $\mathcal{A} = \mathcal{A}[\mathsf{RealRoots}(f) \mid f \in \mathcal{A}[X]]$ — doable, harder
- $\mathbb{Q}[e]$, $\mathbb{Q}[\pi]$ — doable, hard-ish
- $\mathcal{A}[e]$, $\mathcal{A}[\pi]$ — doable, harder still
- $\mathbb{Q}[e, \pi]$ — very difficult open question: Schanuel's conjecture

Concluding Remarks & Current/Future Work

- Hoare logic + algebraic structure is a promising combination
- Easier than logics for programming in some ways
- Inference rules provide structure and understanding of proofs
- Decision procedures give high level of automation when wanted
- Reconcile with *Hoare Logic in the Abstract* (with Oliva & Martin)
- Looking into implementation issues
- Metatheory of linear algebra (with Solovay & Harrison)