A Taylor Function Calculus for Hybrid System Analysis

Validation in Coq

P. Collins[1] M. Niqui[1] N. Revol[2]

[1]CWI

[2]INRIA, LIP, Université de Lyon

3rd NSV Workshop, July 15, 2010

Outline

Motivation

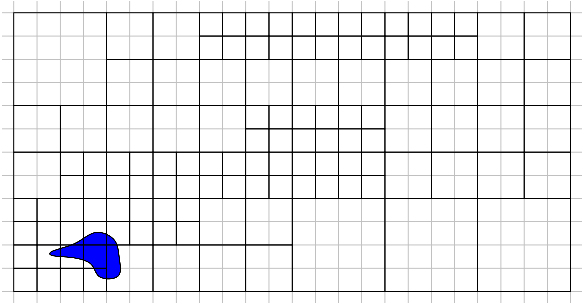
Verification of Hybrid Systems.

- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`

Motivation

Verification of Hybrid Systems.

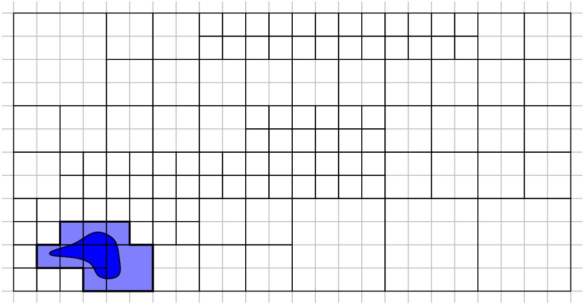
- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`
- Example: Performing reachability analysis.



Motivation

Verification of Hybrid Systems.

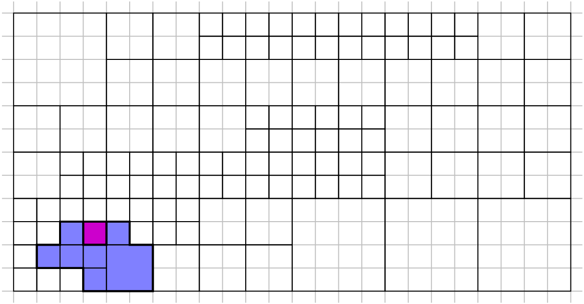
- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`
- Example: Performing reachability analysis.



Motivation

Verification of Hybrid Systems.

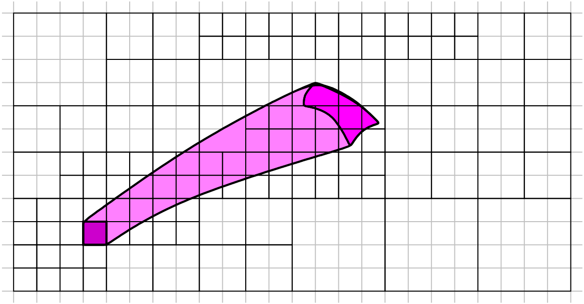
- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`
- Example: Performing reachability analysis.



Motivation

Verification of Hybrid Systems.

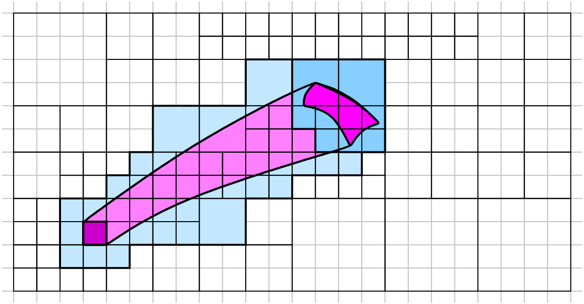
- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`
- Example: Performing reachability analysis.



Motivation

Verification of Hybrid Systems.

- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`
- Example: Performing reachability analysis.

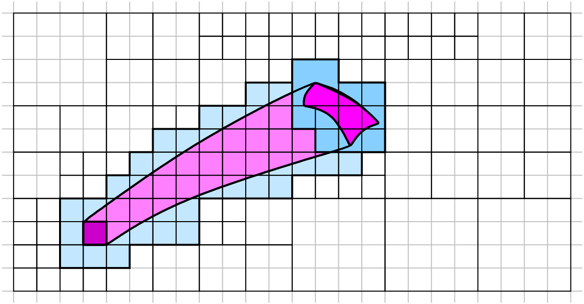


Motivation

Verification of Hybrid Systems.

- Ariadne: Tool for analysis of nonlinear hybrid systems.

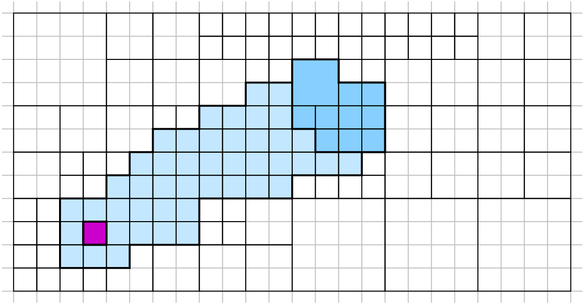
  http://trac.parades.rm.cnr.it/ariadne/
- Example: Performing reachability analysis.



Motivation

Verification of Hybrid Systems.

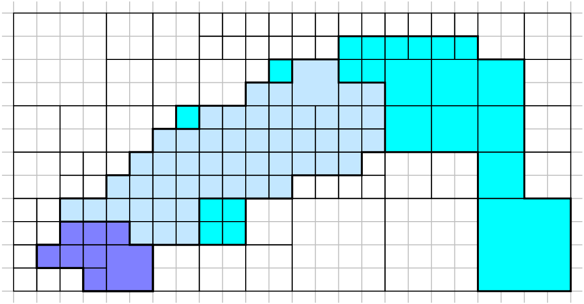
- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`
- Example: Performing reachability analysis.



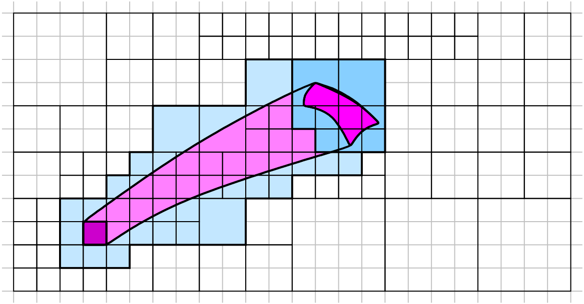
Motivation

Verification of Hybrid Systems.

- Ariadne: Tool for analysis of nonlinear hybrid systems.
  `http://trac.parades.rm.cnr.it/ariadne/`
- Example: Performing reachability analysis.



Motivation



Main steps:

- Computing the flow of a differential equations $\dot{x} = f(x)$
- Computing an outer-approximation of an enclosure on a grid.

These operations must be performed *rigorously* and *efficiently*.

Ariadne

- implemented in (C++)
- kernel is generic!, there are theories for floats, reals and continuous functions.
- theories still work without full implementation.

To get *validated* results:

- verify the kernel
  - primitives for function calculus
  - algorithms for reachability analysis
- verify the result of each calculation.

We use the Coq system.

Ariadne

- implemented in (C++)
- kernel is generic!, there are theories for floats, reals and continuous functions.
- theories still work without full implementation.

To get *validated* results:

- verify the kernel
  - primitives for function calculus
  - algorithms for reachability analysis
- verify the result of each calculation.

We use the Coq system.

Ariadne

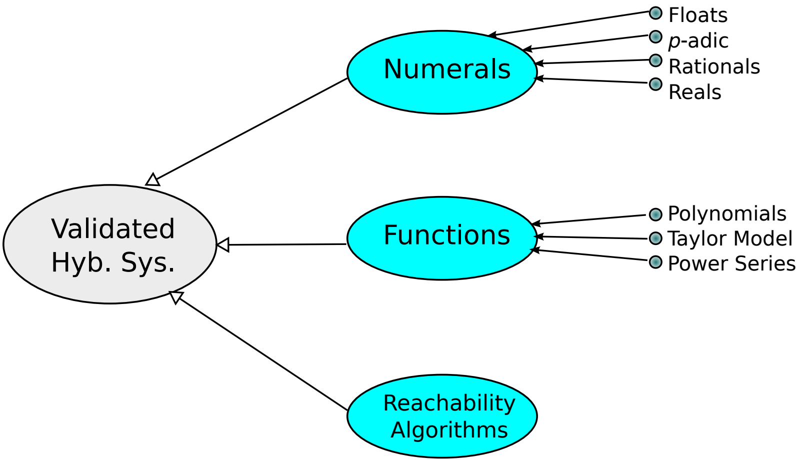
- implemented in (C++)
- kernel is generic!, there are theories for floats, reals and continuous functions.
- theories still work without full implementation.

To get *validated* results:

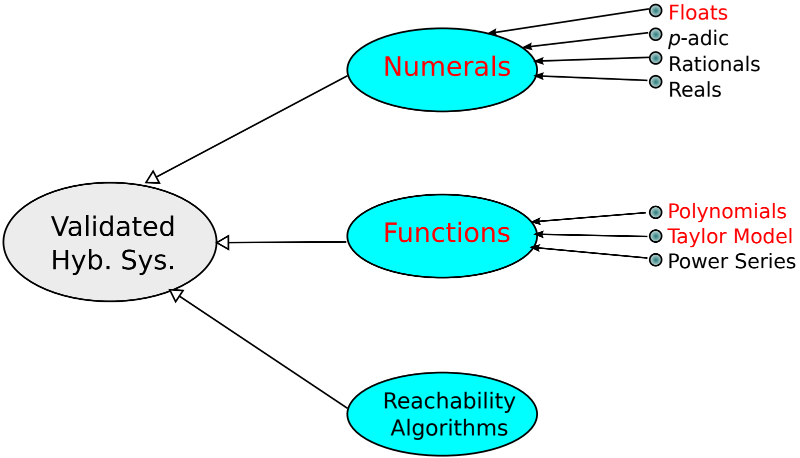
- verify the kernel
  - primitives for function calculus
  - algorithms for reachability analysis
- verify the result of each calculation.

We use the Coq system.

Validating Ariadne



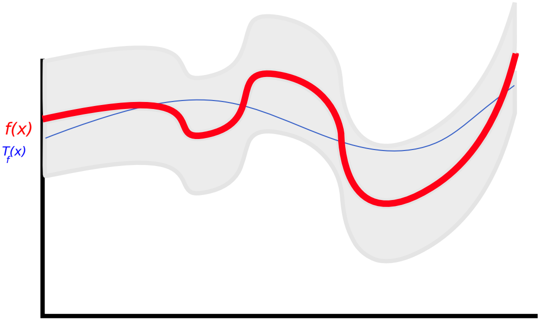
Validating Ariadne



Taylor Models (TM)

Approximate functions using their Taylor expansion.

$f \colon [-1, 1] \longrightarrow \mathbb{R}$ approximated by $T_f \colon [-1, 1] \longrightarrow \mathbb{R}$

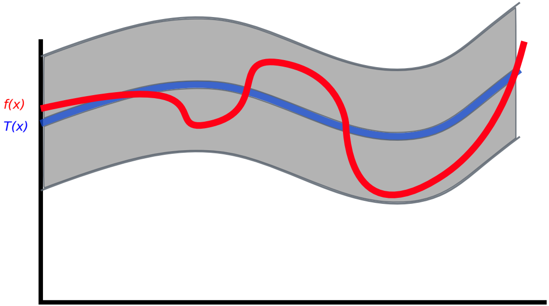


Taylor Models (TM)

Approximate functions using their Taylor expansion.

$f\colon [-1,1] \longrightarrow \mathbb{R}$ approximated by $T_f\colon [-1,1] \longrightarrow \mathbb{R}$

Each polynomial T approximates a family of functions.



Verifying TM

1. TM using constructive reals in Coq Zumkeller
2. TM using rational intervals in PVS Cháves–Daumas
3. Chebyshev models in HOL+Coq ... Joldes, Mayero

We implement a *basic calculus* of Taylor models with coefficients from an *abstract data-type* **F**.

Verifying TM

1. TM using constructive reals in Coq   Zumkeller
2. TM using rational intervals in PVS   Cháves–Daumas
3. Chebyshev models in HOL+Coq ...   Joldes, Mayero

We implement a basic calculus of Taylor models with coefficients from an *abstract data-type* **F**.

F: the minimum interface with respect to which we have a basic Taylor model calculus.

It covers Floats (various base/precision), arbitrary precision, exact etc.

Numerals

$\mathbf{F}$: Type

Constant:

$$0_{\mathbf{F}} : \mathbf{F}$$

Operations:

$- : \mathbf{F} \longrightarrow \mathbf{F}$	*opposite*
$\lvert \_ \rvert : \mathbf{F} \longrightarrow \mathbf{F}$	*abs. value*
$\oplus_u, \oplus_d, \oplus_n : \mathbf{F} \to \mathbf{F} \longrightarrow \mathbf{F}$	*rounded addition*
$\otimes_u, \otimes_d, \otimes_n : \mathbf{F} \to \mathbf{F} \longrightarrow \mathbf{F}$	*rounded multiplication*

Injection

$$\bar{\ } : \mathbf{F} \longrightarrow \mathbb{R}$$

Numerals

Axioms:

- $\overline{0_{\mathbf{F}}} = 0$
- $\forall z,\ \ \overline{-z} = -\overline{z}$
- $\forall z,\ \ \overline{|z|} = |\overline{z}|$

- $\forall z_0 z_1,\ \ |\overline{z_0 \oplus_n z_1} - (\overline{z_0} + \overline{z_1})| \leq |\overline{z_0 \oplus_u z_1} - (\overline{z_0} + \overline{z_1})|$
- $\forall z_0 z_1,\ \ |\overline{z_0 \oplus_n z_1} - (\overline{z_0} + \overline{z_1})| \leq |(\overline{z_0} + \overline{z_1}) - \overline{z_0 \oplus_d z_1}|$
- $\forall z_0 z_1,\ \ \overline{z_0 \oplus_d z_1} \leq \overline{z_0} + \overline{z_1} \leq \overline{z_0 \oplus_u z_1}$

- $\forall z_0 z_1,\ \ |\overline{z_0 \otimes_n z_1} - (\overline{z_0} \cdot \overline{z_1})| \leq |\overline{z_0 \otimes_u z_1} - (\overline{z_0} \cdot \overline{z_1})|$
- $\forall z_0 z_1,\ \ |\overline{z_0 \otimes_n z_1} - (\overline{z_0} \cdot \overline{z_1})| \leq |(\overline{z_0} \cdot \overline{z_1}) - \overline{z_0 \otimes_d z_1}|$
- $\forall z_0 z_1,\ \ \overline{z_0 \otimes_d z_1} \leq \overline{z_0} \cdot \overline{z_1} \leq \overline{z_0 \otimes_u z_1}$

Axiomatisation

- Formalised in Coq
- Instantiations not needed in our work
- Possible instances
  - Coq's (unbound) Floats Daumas–Rideau–Théry, Boldo
  - Subsets of $\mathbb{F}_{32}$, $\mathbb{F}_{64}$ (normalised, no *NaN*, $\pm\infty$ etc.)
  - *p*-adics
  - $\mathbb{Q}$
  - $\mathbb{R}$ (axiomatic, non-constructive)
  - Constructive exact reals
  - Singleton $\{0\}$ (in fact finite groups!)

Axiomatisation

☞ Axioms for $\oplus$

- $\forall z_0 z_1,\ \ |\overline{z_0 \oplus_n z_1} - (\overline{z_0} + \overline{z_1})| \leq |\overline{z_0 \oplus_u z_1} - (\overline{z_0} + \overline{z_1})|$
- $\forall z_0 z_1,\ \ |\overline{z_0 \oplus_n z_1} - (\overline{z_0} + \overline{z_1})| \leq |(\overline{z_0} + \overline{z_1}) - \overline{z_0 \oplus_d z_1}|$

can be replaced by *either* of

1. $\forall z_0 z_1,\ \ |\overline{z_0 \oplus_n z_1} - (\overline{z_0} + \overline{z_1})| \leq \frac{\overline{(z_0 \oplus_u z_1) \ominus (z_0 \oplus_d z_1)}}{2}$
2. $\forall z_0 z_1 z,\ \ |\overline{z_0 \oplus_n z_1} - (\overline{z_0} + \overline{z_1})| \leq |z - (\overline{z_0} + \overline{z_1})|$
3. alternatively we could define

$$z_0 \oplus_u z_1 := \inf\{z \in \mathbf{F} | \overline{z_0} + \overline{z_1} \leq z\}$$

and require that the infimum exists.

Axiomatisation

☞ We can add similar axiom schema for *composite* operations. (eg. fusedMultiplyAdd $xy+z$):

For $*\colon \mathbb{R} \times \cdots \times \mathbb{R} \longrightarrow \mathbb{R}$ we can add

$$\circledast_{u,d,n}\colon \mathbf{F} \times \cdots \times \mathbf{F} \longrightarrow \mathbf{F}$$

satisfying

- $\circledast_d(z_0, \ldots, z_k) \leq *(\overline{z_0}, \ldots, \overline{z_k}) \leq \circledast_u(z_0, \ldots, z_k)$
- $|\circledast_n(z_0, \ldots, z_k) - *(\overline{z_0}, \ldots, \overline{z_k})| \leq |\circledast_{u,d}(z_0, \ldots, z_k) - *(\overline{z_0}, \ldots, \overline{z_k})|$

Verifying TM

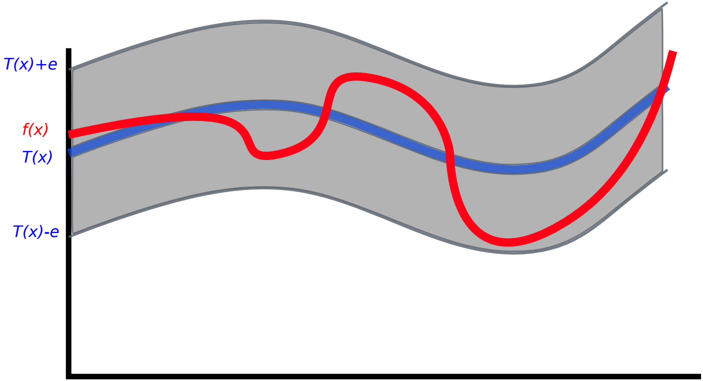
1. TM using constructive reals in Coq   Zumkeller
2. TM using rational intervals in PVS   Cháves–Daumas
3. Chebyshev models in HOL+Coq ...   Joldes, Mayero

We implement a *basic calculus* of Taylor models with coefficients from an abstract data-type **F**.

Taylor Models with Floating point Coefficients

Analysed by Revol–Makino–Berz for *COSY* system.

Tedious because of several layers of rounding and truncation.
TM: pair of polynomial T and ε *error*.



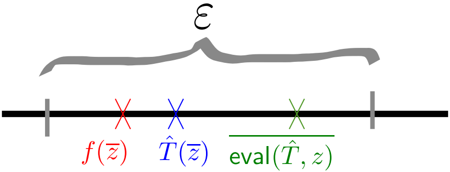
Taylor Models with Floating point Coefficients

TM: pair of polynomial T and ε the error.

For exact Taylor models ε denotes truncation error.

1. If $\hat{T}$ is obtained from T with floating point rep. for coefficients, there is rounding error $|T(x) - \hat{T}(x)|$.
2. If $z \in \mathbf{F}$, then $\hat{T}(\bar{z})$ can be calculated
   - exactly, or
   - using operations on **F** ☜ *extra rounding error*.

Ideally, If T models $f: \mathbb{R} \longrightarrow \mathbb{R}$ we should have



TM over **F** in Coq

- Let p be a sparse polynomial over **F**, eg.

$$p(x) := a_0 x^{n_0} + a_1 x^{n_1} + \cdots + a_k x^{n_k}$$

  where $n_i < n_{i+1}$ and $a_i \in \mathbf{F}$.

- $\varepsilon \in \mathbf{F}$

$\langle p, \varepsilon \rangle$ is a Taylor model.

$TM_{\mathbf{F}}$ is the type of Taylor models over **F**.

TM over **F** in Coq

$\langle p, \varepsilon \rangle \models_r f$ if:

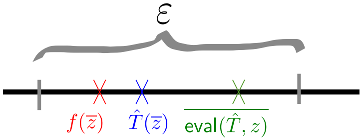
$$\forall z \in [-1, 1], |p(\bar{z}) - f(\bar{z})| \leq \varepsilon \ .$$

Let $E\colon TM_{\mathbf{F}} \times \mathbf{F} \longrightarrow \mathbf{F}$, E is an *evaluation* if

$$\forall \langle p, \varepsilon \rangle \forall \bar{z} \in [-1, 1], |\overline{E(\langle p, \varepsilon \rangle, \bar{z})} - f(\bar{z})| \leq \varepsilon \ .$$

Currently Ariadne has a concrete evaluation (eval).

$\langle p, \varepsilon \rangle \models f$ models f if for each z with $\bar{z} \in [-1, 1]$



$TM_{\mathbf{F}}$ calculus

Scalar multiplication

- If $\langle p, \varepsilon \rangle \models_r f$, $c \in \mathbf{F}$ then

$$\langle c \otimes_n p, \varepsilon' \rangle \models_r \bar{c} f$$

where

$$\varepsilon' := |c| \otimes_u \varepsilon \oplus_u \bigoplus_{i=0}^{k} c \otimes_u a_i \ominus_u c \otimes_d a_i$$

and $p(x) = a_0 x^{n_0} + a_1 x^{n_1} + \cdots + a_k x^{n_k}$

TM_{F} calculus

Addition

- If $\langle p_0, \varepsilon_0 \rangle \models_r f_0$, $\langle p_1, \varepsilon_1 \rangle \models_r f_1$ then

$$\langle p_0 \oplus_n p_1, \varepsilon' \rangle \models_r f_0 + f_1$$

where

$$\varepsilon' := \varepsilon_0 \oplus_u \varepsilon_1 \oplus_u \bigoplus_{n_i = m_j} a_i \otimes_u b_j \ominus a_i \otimes_d b_j$$

and $p_0(x) = a_0 x^{n_0} + a_1 x^{n_1} + \cdots + a_k x^{n_k}$,
$p_1(x) = b_0 x^{m_0} + b_1 x^{m_1} + \cdots + b_k x^{m_l}$

$TM_{\mathbf{F}}$ calculus

We can find ε for the following operations:

- *monomial* product; if $\langle p, \varepsilon\rangle \models_r f$ then $\langle xp, \varepsilon\rangle \models_r xf(x)$
- *multiplication*; if $\langle p_0, \varepsilon_0\rangle \models_r f_0$, $\langle p_1, \varepsilon_1\rangle \models_r f_1$ then

$$\langle p_0 \otimes_n p_1, \varepsilon'\rangle \models_r f_0 f_1$$

$TM_{\mathbf{F}}$ calculus

Suppose $\langle p, \varepsilon\rangle \models f$, $c \in \mathbf{F}$, then we can find ε' s.t.

$$\langle c \otimes_n p, \varepsilon'\rangle \models \bar{c}f$$

But we need to amend the axiomatisation.

Add constants $1_{\mathbf{F}}, \varepsilon_m \in \mathbf{F}$, let $2_{\mathbf{F}} := 1_{\mathbf{F}} \oplus_u 1_{\mathbf{F}}$, and add axioms

- $\overline{1_{\mathbf{F}}} = 1$
- $0 < \overline{\varepsilon_m}$
- $|\overline{z_0 \otimes_n z_1} - \overline{z_0} \times \overline{z_1}| \leq \overline{|z_0 \otimes_n z_1| \otimes_n 2_{\mathbf{F}} \otimes_n \varepsilon_m}$

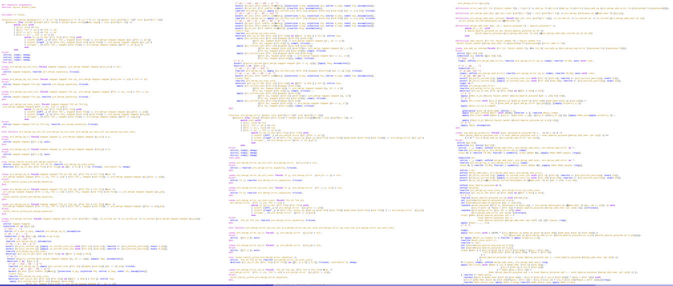
Intended meaning: ε_m is some value $> 2ulp$.

☞ For addition we need axioms for $\oplus$.

Formal Proofs

```
TaylorModel operator+(TaylorModel x, TaylorModel y) {
  TaylorModel r(x.argument size());
  Term xterm=x.begin(); Term yterm=y.begin();
  while (xterm!=x.end() && yterm!=y.end()) {
    if (xterm.key == yterm.key) {
      Float u = add up(xterm.value,yterm.value);
      Float l = add down(xterm.value,yterm.value);
      r.error = add up(r.error,sub up(u,l)/2);
      r.new term( xterm.key,
          add near(xterm.value, yterm.value) );
      ++xterm; ++yterm;
    } else if(xterm.key<yterm.key) {
      r.new term( xterm ); ++xterm;
    } else if(yterm.key<xterm.key) {
      r.new term( yterm ); ++yterm;
    }
  }
  r->error = add up(r.error, x.error, y.error);
  return r;
}
```

20 lines of C++



370 lines of Coq

230 spec (60%)
140 proof (40%)

Extending the $TM_{\mathbf{F}}$ Calculus?

division to be added to the axiomatisation for **F**.

- *NaN* and $\pm\infty$ will be added.
- Operations and axioms to be updated on $\mathbf{F} + \{NaN, \pm\infty\}$.
- *anti-differentiation* of Taylor models is possible using integer division.

Further Parametrisation?

To 'future-proof' the framework ideally we have to develop

- Polynomials: a minimal interface covering
  - representation: sparse, incremental sparse, . . .
  - evaluation: ordinary, Horner, . . .
  - calibration: sweeping the truncation error
- Function Models: a type for approx. of continuous functions.

Further Parametrisation?

To ‘future-proof’ the framework ideally we have to develop

- Polynomials: a minimal interface covering
- Function Models: a type for approx. of continuous functions.

  The minimal interface should cover

  - evaluation on floats, intervals and real numbers
  - composition of approximations
  - composition of a computable function and an approximation.

☞ Good for transcendental functions.