Formal Proofs and Numerical Computations

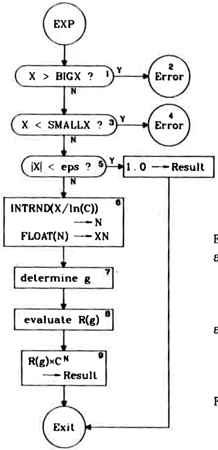
Guillaume Melquiond

Université Paris-Saclay, CNRS, ENS Paris-Saclay, Inria
Laboratoire Méthodes Formelles

June 16, 2022

Cody & Waite, 1979: Implementing Exponential

b. Flow Chart for EXP(X)



for 30 ≤ b ≤ 42

p0 = 0.24999 99999 9992
p1 = 0.00595 04254 9776
q0 = 0.5
q1 = 0.05356 75176 4522
q2 = 0.00029 72936 3682

for 43 ≤ b ≤ 56

p0 = 0.24999 99999 99999 993
p1 = 0.00694 36000 15117 929
p2 = 0.00001 65203 30026 828
q0 = 0.5
q1 = 0.05555 38666 96900 119
q2 = 0.00049 58628 84905 441

Evaluate $R(g)$ in fixed point. First form $z = g^2$. Then form $g*P(z)$ and $Q(z)$ using nested multiplication. For example, for $43 \le b \le 56$,

g*P(z) = ((p2 * z + p1) * z + p0) * g

and

Q(z) = (q2 * z + q1) * z + q0.

Finally, form

r = .5 + g*P(z)/[Q(z)-g*P(z)]

in fixed point and convert back to floating point with $R(g)$ = REFLOAT(r) (see Chapter 2).

Cody & Waite, 1979: Implementing Exponential

Approximating $\exp x$

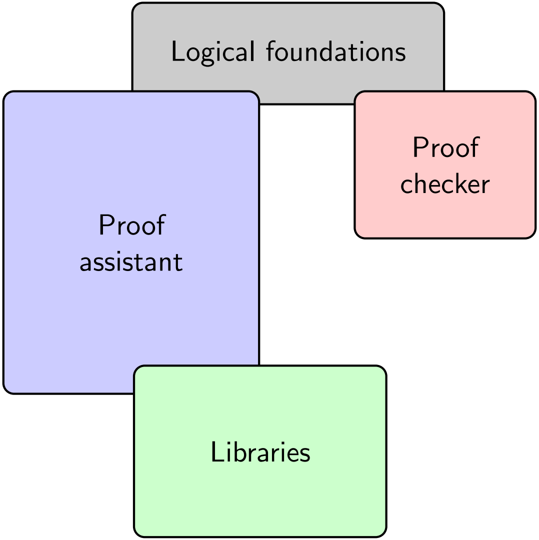
1. Argument reduction: $t = x - k \log 2$.
2. Rational approximation $f(t)$ of $\exp t$.
3. Result reconstruction: $\exp x = 2^k \exp t$.

Source of errors

- Rounding errors: $\tilde{t} \simeq x - k \log 2$ and $\tilde{f}(\tilde{t}) \simeq f(\tilde{t})$.
- Method error: $f(\tilde{t}) \simeq \exp \tilde{t}$.

Verifying a mathematical library is tedious and error-prone.

Formal Verification



Formal Statement of Correctness

Bounding the relative method error

```
Definition q1 := 8006155947364787 * pow2 (-57).
Definition q2 := 4573527866750985 * pow2 (-63).

Definition f t :=
  let t2 := t * t in
  let p := p0 + t2 * (p1 + t2 * p2) in
  let q := q0 + t2 * (q1 + t2 * q2) in
  2 * ((t * p) / (q - t * p) + 1/2).

Lemma method_error :
  forall t : R, Rabs t <= 0.35 ->
  Rabs ((f t - exp t) / exp t) <= 5e-18.
```

Using a Computer Algebra System

About SageMathCell

SageMathCell

Type some Sage code below and press Evaluate.

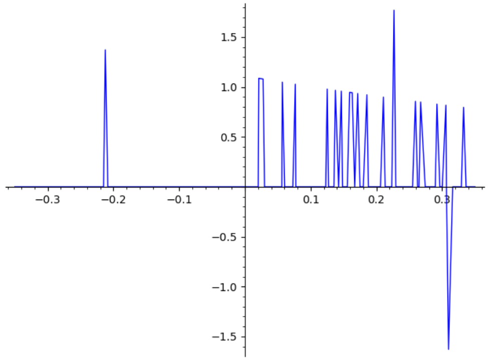
```
q2 = 4573527866750985 * 2**(-63)

t = SR.var('t')
t2 = t * t
p = p0 + t2 * (p1 + t2 * p2)
q = q0 + t2 * (q1 + t2 * q2)
f = 2 * ((t * p) / (q - t * p) + 1/2)
err(x) = (f(t = x) - exp(x)) / exp(x)
pretty_print(html(r'Relative error is bounded by %g' %
                  find_local_maximum(abs(err), -0.35, 0.35)[0]))
show(plot(err, -0.35, 0.35))
```

Evaluate

Language: Sage

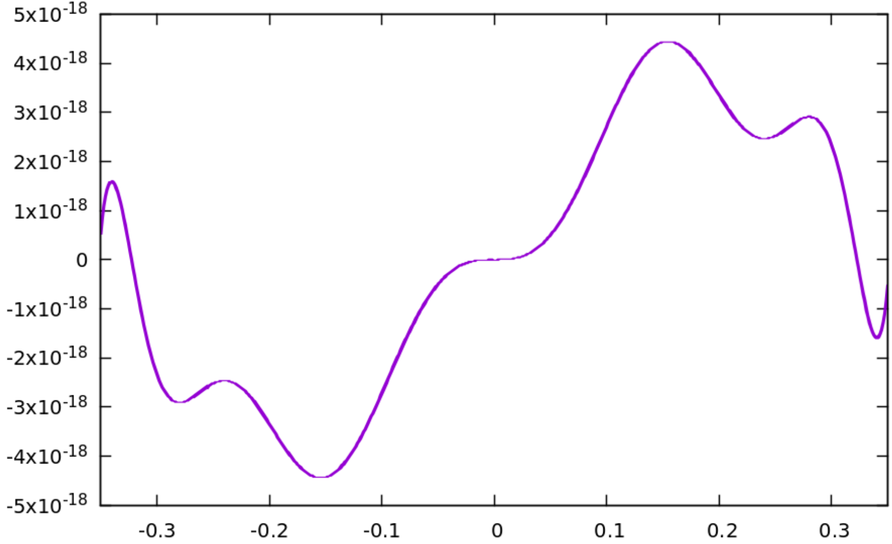
Share

Relative error is bounded by 1.57372e-16



Help | Powered by SageMath

Using the Coq Proof Assistant



Bounding the relative method error

```
Lemma method_error :
  forall t : R, Rabs t <= 0.35 ->
  Rabs ((f t - exp t) / exp t) <= 5e-18.
Proof.
  intros t Ht.
  interval with (i_bisect t, i_taylor t, i_prec 80).
  (* Finished transaction in 2.768 secs *)
Qed.
```

Formal Verification by Computational Reflection

To prove $\forall x \in X,\ \forall \vec{y} \in \vec{Y},\ e(x, \vec{y}) \in Z$

1. Compute a polynomial p and an interval Δ such that $\forall x \in X,\ \forall \vec{y} \in \vec{Y},\ e(x, \vec{y}) - p(x) \in \Delta$.
2. Check that $p(X) + \Delta \subseteq Z$.
3. If not, split X and $\vec{Y}$ into smaller intervals and start again.

- By structural induction on e.
- Using interval arithmetic and Taylor models.
- Fully reflective, i.e., same as $\text{check}(\ulcorner e \urcorner, X, \vec{Y}, Z) = \text{true}$.
- Floating-point numbers are emulated inside the logic of Coq.
- No axioms other than those defining the set of real numbers.

Proper and Improper Integrals

Helfgott's proof of the ternary Goldbach conjecture

Every odd number greater than 5 is the sum of three primes.

$$\int_{-\infty}^{\infty} \frac{(0.5 \cdot \ln(\tau^2 + 2.25) + 4.1396 + \ln \pi)^2}{0.25 + \tau^2} \, d\tau \leq 226.844.$$

```
Goal RInt (fun tau =>
       (0.5 * ln(tau^2 + 2.25) + 4.1396 + ln PI)^2
       / (0.25 + tau^2))
     (-100000) 100000
  = 226.8435 ± 2e-4.
Proof. integral. Qed. (* Finished in 1.29 secs *)

Goal RInt_gen (fun tau =>
       ... * (powerRZ tau (-2) * (ln tau)^2))
     (at_point 100000) (Rbar_locally p_infty)
  = 0.00317742 ± 1e-5.
Proof. integral. Qed. (* Finished in 0.228 secs *)
```

Some More Examples of CoqInterval

Plotting the method error of Cody & Waite's exponential

```
Plot ltac:(plot (fun t => (f t - exp t) / exp t)
    (-0.35) 0.35 with (i_prec 80)).
```

Finding roots

```
Definition foo x (H: x + cos x = 2) := ltac:(root H).
About foo.
(* foo : forall x : R, x + cos x = 2 ->
6728983409886093/2^51 <= x <= 6728983409886103/2^51 *)
```

https://coqinterval.gitlabpages.inria.fr/

What About Rounding Errors? Flocq & Gappa

Accuracy of Cody & Waite's exponential

```
Definition cw_exp (x : R) :=
  let k := nearbyint (mul x InvLog2) in
  let t := sub (sub x (mul k Log2h)) (mul k Log2l) in
  ...

Theorem exp_correct : forall x : R,
  generic_format radix2 (FLT_exp (-1074) 53) x ->
  -746 <= x <= 710 ->
  Rabs ((cw_exp x - exp x) / exp x) <= pow2 (-51).
Proof.
... generalize (method_error t Bt).
... gappa.
Qed.
```

https://flocq.gitlabpages.inria.fr/
https://gappa.gitlabpages.inria.fr/