

# When a logarithm is just a misspelled algorithm

*Mioara Joldes*

Supervised by Nicolas Brisebarre and Jean-Michel Muller

Arénaire Team, Laboratoire de l'Informatique du Parallélisme  
École Normale Supérieure de Lyon

Neuvième forum des jeunes mathématiciennes, November 6-7, 2009

# Dark Discussions at Cafe Infinity<sup>1</sup>

---

<sup>1</sup>Using Computers as Proof - Math is fun Forum  
<http://www.mathisfunforum.com/viewtopic.php?id=11990>

# Dark Discussions at Cafe Infinity<sup>1</sup>

*“So on my last discrete math II class we were given the following question:”* for all  $n \in \mathbb{N}$  and  $x \in \mathbb{R}$ , with  $n < x < n + 1$ , prove that

$$\log(x^2) > \log(n^2 + n)$$

---

<sup>1</sup>Using Computers as Proof - Math is fun Forum  
<http://www.mathisfunforum.com/viewtopic.php?id=11990>

# Dark Discussions at Cafe Infinity<sup>1</sup>

*“So on my last discrete math II class we were given the following question:”* for all  $n \in \mathbb{N}$  and  $x \in \mathbb{R}$ , with  $n < x < n + 1$ , prove that

$$\log(x^2) > \log(n^2 + n)$$

*“I tackled the problem for 20 minutes without success, and then suddenly realized, it was definitely wrong. I grabbed my calculator and chose  $n = 5$ , and  $x = 5.1$ , and computed:”*

---

<sup>1</sup>Using Computers as Proof - Math is fun Forum  
<http://www.mathisfunforum.com/viewtopic.php?id=11990>

# Dark Discussions at Cafe Infinity<sup>1</sup>

*“So on my last discrete math II class we were given the following question:”* for all  $n \in \mathbb{N}$  and  $x \in \mathbb{R}$ , with  $n < x < n + 1$ , prove that

$$\log(x^2) > \log(n^2 + n)$$

*“I tackled the problem for 20 minutes without success, and then suddenly realized, it was definitely wrong. I grabbed my calculator and chose  $n = 5$ , and  $x = 5.1$ , and computed:”*

$$\log(x^2) = \log(5.1^2) = 3.258\dots \quad \log(n^2 + n) = \log(30) = 3.401\dots$$

---

<sup>1</sup>Using Computers as Proof - Math is fun Forum  
<http://www.mathisfunforum.com/viewtopic.php?id=11990>

# Dark Discussions at Cafe Infinity<sup>1</sup>

*“So on my last discrete math II class we were given the following question:”* for all  $n \in \mathbb{N}$  and  $x \in \mathbb{R}$ , with  $n < x < n + 1$ , prove that

$$\log(x^2) > \log(n^2 + n)$$

*“I tackled the problem for 20 minutes without success, and then suddenly realized, it was definitely wrong. I grabbed my calculator and chose  $n = 5$ , and  $x = 5.1$ , and computed:”*

$$\log(x^2) = \log(5.1^2) = 3.258\dots \quad \log(n^2 + n) = \log(30) = 3.401\dots$$

*“I then wrote «The claim is false» on the exam sheet, and gave the above calculations to support it, to receive credit for the problem.”*

---

<sup>1</sup>Using Computers as Proof - Math is fun Forum  
<http://www.mathisfunforum.com/viewtopic.php?id=11990>

# Dark Discussions at Cafe Infinity<sup>1</sup>

*"After getting the exam back, my teacher laughed at me and said my «disproof by calculator» was garbage, and reprimanded me for trusting the accuracy of its calculations. He then gave us his own «proof» of the claim.*

*But I trusted my calculator more than that... So this time i said:*

$$4 < 6 < 9 \Rightarrow 2 < \sqrt{6} < 3$$

so take  $n = 2$  and  $x = \sqrt{6}$ , then

$$\log(x^2) > \log(n^2 + n) \Leftrightarrow \log(6) > \log(6)$$

*which is obviously false, no calculators needed.*

---

<sup>1</sup>Using Computers as Proof - Math is fun Forum  
<http://www.mathisfunforum.com/viewtopic.php?id=11990>

# Dark Discussions at Cafe Infinity<sup>1</sup>

In conclusion:

*The teacher couldn't ignore this... Then I discovered and pointed out the error in his proof, which he acknowledged.*

*But I do find it surprising that he distrusted the accuracy of a calculator so much. In fact, my calculator gives me 9 digits after the decimal place. Why would it bother to give me 9 if the first 2 were not reliable?*

*He, however, greatly doubted calculators were this sophisticated, and insisted they can never be trusted.*

*Now I understand that a calculator only gives an approximation, but I at least trust the first  $n - 1$  digits if  $n$  are given.*

*Is this unreasonable? What do YOU think?*

---

<sup>1</sup>Using Computers as Proof - Math is fun Forum  
<http://www.mathisfunforum.com/viewtopic.php?id=11990>

- Introduction
  - Floating point arithmetic
  - Mathematical Libraries
  - Correctly rounded elementary functions
  - Supremum norm of error functions
- Rigorous computing tools
  1. Interval arithmetic
  2. Taylor models
- Conclusion

## Floating point (FP)

*"Look engineers. All we're asking for is an infinite number of transistors on a finite-sized chip. You can't even do that?"*

A real number  $x$  is approximated in machine by a rational:

$$x = (-1)^s \times m \times \beta^e$$

- $\beta$  is the radix (usually  $\beta = 2$ )
- $s$  is a sign bit
- $m$  is the mantissa, a rational number of  $n_m$  digits in radix  $\beta$ :

$$m = d_0, d_1 d_2 \dots d_{n_m-1}$$

- $e$  is the exponent, a signed integer on  $n_e$  bits

## Floating point - Some common misconceptions <sup>2</sup>

- ✗ *A floating-point number somehow represents an interval of values around the “real value”.*
  - ! An FP number only represents itself (a rational).
  - ! If there is an epsilon or an uncertainty somewhere in your data, it is your job (as a programmer) to model and handle it.
- ✗ *I need 3 significant digits in the end, a double holds 15 decimal digits, therefore I shouldn't worry about precision.*
  - ! You can destroy 14 significant digits in one subtraction, but it is relatively easy to avoid if you expect it

---

<sup>2</sup>Florent de Dinechin,  
<http://lipforge.ens-lyon.fr/www/crlibm/documents/cern.pdf>

## Floating point - Some common misconceptions <sup>2</sup>

- ✗ *All floating-point operations involve a (somehow fuzzy) rounding error.*
- ✓ Since 1985 there is a IEEE standard (IEEE-754) for FP arithmetic.
- ✓ Correct Rounding: An operation whose entries are FP numbers must return what we would get by infinitely precise operation followed by rounding. The standard defines 4 rounding modes.

---

<sup>2</sup>Florent de Dinechin,  
<http://lipforge.ens-lyon.fr/www/crlibm/documents/cern.pdf>

# Floating point (FP)

- ✓ IEEE-754 requests correct rounding for :  $+$ ,  $-$ ,  $\times$ ,  $\div$ ,  $\sqrt{\cdot}$ .  
Advantages:
- ✓ FP programs with only these operations are deterministic
- ✓ Accuracy and portability are improved

# Floating point (FP)

- ✓ IEEE-754 requests correct rounding for :  $+$ ,  $-$ ,  $\times$ ,  $\div$ ,  $\sqrt{\cdot}$ .  
Advantages:
- ✓ FP programs with only these operations are deterministic
- ✓ Accuracy and portability are improved

What about elementary functions (sin, cos, log, etc.)?

# Mathematical Libraries

- Software systems: scientific computing, financial, embedded systems.
- Need to compute  $\sin$ ,  $\cos$ ,  $\exp$  in finite precision, floating-point environment.

# Mathematical Libraries

- Software systems: scientific computing, financial, embedded systems.
- Need to compute  $\sin$ ,  $\cos$ ,  $\exp$  in finite precision, floating-point environment.
- Most Mathematical Libraries do not provide correctly rounded functions.
- IEEE-754-2008 recommends correct rounding.

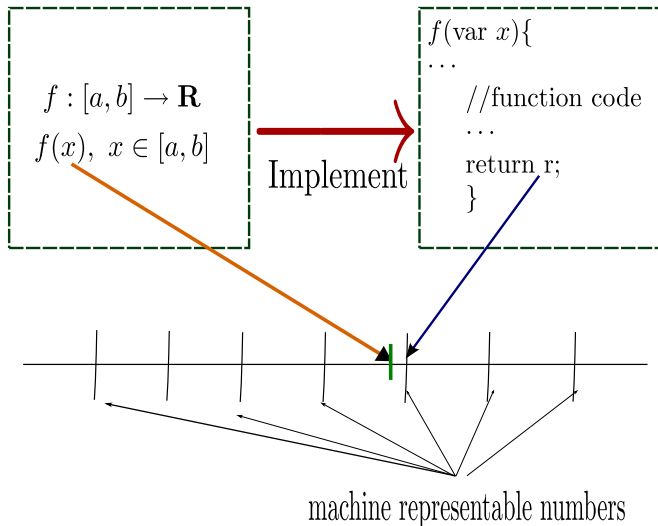
# Mathematical Libraries

- Software systems: scientific computing, financial, embedded systems.
- Need to compute  $\sin$ ,  $\cos$ ,  $\exp$  in finite precision, floating-point environment.
- Most Mathematical Libraries do not provide correctly rounded functions.
- IEEE-754-2008 recommends correct rounding.
- Ainaire team develops the Correctly Rounded Libm (CRLibm<sup>†</sup>).

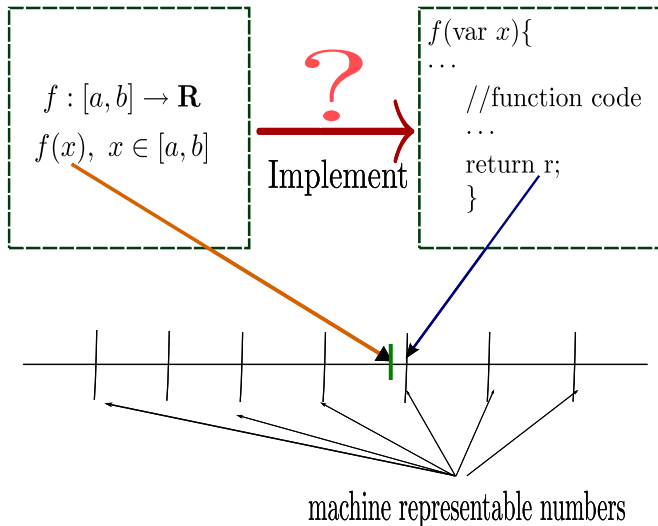
---

<sup>†</sup><http://lipforge.ens-lyon.fr/www/crlibm/>

# Correctly rounded functions



# Correctly rounded functions



# Correctly rounded functions

- Table Maker's Dilemma

Deg. 4			
m	Size	Lagrange (2) <sup>2</sup> Error	Lagrange (2) <sup>2</sup> Error
0	69719	3463744	34669391
1	70049	3465932	3466131
2	70379	3468120	3465323
3	70707	3470308	3464515
4	71037	3472496	3463707
5	71367	3474684	3462899
6	71697	3476872	3462091
7	72027	3479060	3461283
8	72357	3481248	3460475
9	72687	3483436	3459667
10	73017	3485624	3458859
11	73347	3487812	3458051
12	73677	3490000	3457243
13	74007	3492188	3456435
14	74337	3494376	3455627
15	74667	3496564	3454819
16	74997	3498752	3454011
17	75327	3500940	3453203
18	75657	3503128	3452395
19	75987	3505316	3451587
20	76317	3507504	3450779
21	76647	3509692	3449971
22	76977	3511880	3449163
23	77307	3514068	3448355
24	77637	3516256	3447547
25	77967	3518444	3446739
26	78297	3520632	3445931
27	78627	3522820	3445123
28	78957	3525008	3444315
29	79287	3527196	3443507
30	79617	3529384	3442699
31	79947	3531572	3441891
32	80277	3533760	3441083
33	80607	3535948	3440275
34	80937	3538136	3439467
35	81267	3540324	3438659
36	81597	3542512	3437851
37	81927	3544700	3437043
38	82257	3546888	3436235
39	82587	3549076	3435427
40	82917	3551264	3434619
41	83247	3553452	3433811
42	83577	3555640	3433003
43	83907	3557828	3432195
44	84237	3560016	3431387
45	84567	3562204	3430579
46	84897	3564392	3429771
47	85227	3566580	3428963
48	85557	3568768	3428155
49	85887	3570956	3427347
50	86217	3573144	3426539
51	86547	3575332	3425731
52	86877	3577520	3424923
53	87207	3579708	3424115
54	87537	3581896	3423307
55	87867	3584084	3422499
56	88197	3586272	3421691
57	88527	3588460	3420883
58	88857	3590648	3420075
59	89187	3592836	3419267
60	89517	3595024	3418459
61	89847	3597212	3417651
62	90177	3599400	3416843
63	90507	3601588	3416035
64	90837	3603776	3415227
65	91167	3605964	3414419
66	91497	3608152	3413611
67	91827	3610340	3412803
68	92157	3612528	3411995
69	92487	3614716	3411187
70	92817	3616904	3410379
71	93147	3619092	3409571
72	93477	3621280	3408763
73	93807	3623468	3407955
74	94137	3625656	3407147
75	94467	3627844	3406339
76	94797	3630032	3405531
77	95127	3632220	3404723
78	95457	3634408	3403915
79	95787	3636596	3403107
80	96117	3638784	3402299
81	96447	3640972	3401491
82	96777	3643160	3400683
83	97107	3645348	3399875
84	97437	3647536	3399067
85	97767	3649724	3398259
86	98097	3651912	3397451
87	98427	3654100	3396643
88	98757	3656288	3395835
89	99087	3658476	3395027
90	99417	3660664	3394219
91	99747	3662852	3393411
92	100077	3665040	3392603
93	100407	3667228	3391795
94	100737	3669416	3390987
95	101067	3671604	3390179
96	101397	3673792	3389371
97	101727	3675980	3388563
98	102057	3678168	3387755
99	102387	3680356	3386947
100	102717	3682544	3386139

Deg. 85

Deg. 4			
m	Size	Lagrange (2) <sup>2</sup> Error	Lagrange (2) <sup>2</sup> Error
10	74719	3541779	3444080
11	75049	3543967	3443272
12	75379	3546155	3442464
13	75707	3548343	3441656
14	76037	3550531	3440848
15	76367	3552719	3440040
16	76697	3554907	3439232
17	77027	3557095	3438424
18	77357	3559283	3437616
19	77687	3561471	3436808
20	78017	3563659	3436000
21	78347	3565847	3435192
22	78677	3568035	3434384
23	79007	3570223	3433576
24	79337	3572411	3432768
25	79667	3574599	3431960
26	79997	3576787	3431152
27	80327	3578975	3430344
28	80657	3581163	3429536
29	80987	3583351	3428728
30	81317	3585539	3427920
31	81647	3587727	3427112
32	81977	3589915	3426304
33	82307	3592103	3425496
34	82637	3594291	3424688
35	82967	3596479	3423880
36	83297	3598667	3423072
37	83627	3600855	3422264
38	83957	3603043	3421456
39	84287	3605231	3420648
40	84617	3607419	3419840
41	84947	3609607	3419032
42	85277	3611795	3418224
43	85607	3613983	3417416
44	85937	3616171	3416608
45	86267	3618359	3415800
46	86597	3620547	3414992
47	86927	3622735	3414184
48	87257	3624923	3413376
49	87587	3627111	3412568
50	87917	3629299	3411760
51	88247	3631487	3410952
52	88577	3633675	3410144
53	88907	3635863	3409336
54	89237	3638051	3408528
55	89567	3640239	3407720
56	89897	3642427	3406912
57	90227	3644615	3406104
58	90557	3646803	3405296
59	90887	3648991	3404488
60	91217	3651179	3403680
61	91547	3653367	3402872
62	91877	3655555	3402064
63	92207	3657743	3401256
64	92537	3659931	3400448
65	92867	3662119	3399640
66	93197	3664307	3398832
67	93527	3666495	3398024
68	93857	3668683	3397216
69	94187	3670871	3396408
70	94517	3673059	3395600
71	94847	3675247	3394792
72	95177	3677435	3393984
73	95507	3679623	3393176
74	95837	3681811	3392368
75	96167	3684000	3391560
76	96497	3686188	3390752
77	96827	3688376	3389944
78	97157	3690564	3389136
79	97487	3692752	3388328
80	97817	3694940	3387520
81	98147	3697128	3386712
82	98477	3699316	3385904
83	98807	3701504	3385096
84	99137	3703692	3384288
85	99467	3705880	3383480
86	99797	3708068	3382672
87	100127	3710256	3381864
88	100457	3712444	3381056
89	100787	3714632	3380248
90	101117	3716820	3379440
91	101447	3719008	3378632
92	101777	3721196	3377824
93	102107	3723384	3377016
94	102437	3725572	3376208
95	102767	3727760	3375400
96	103097	3729948	3374592
97	103427	3732136	3373784
98	103757	3734324	3372976
99	104087	3736512	3372168
100	104417	3738700	3371360

Deg. 85

# Correctly rounded functions

- Table Maker's Dilemma

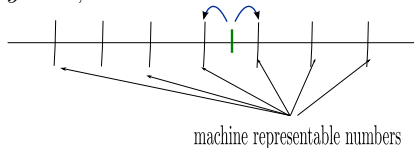
Deg. 4 +---			
m	Si	Logarithm	Si
0	69751	1463744	4660901
1	70047	1463744	4660901
2	70343	1463744	4660901
3	70639	1463744	4660901
4	70935	1463744	4660901
5	71231	1463744	4660901
6	71527	1463744	4660901
7	71823	1463744	4660901
8	72119	1463744	4660901
9	72415	1463744	4660901
10	72711	1463744	4660901
11	73007	1463744	4660901
12	73303	1463744	4660901
13	73599	1463744	4660901
14	73895	1463744	4660901
15	74191	1463744	4660901
16	74487	1463744	4660901
17	74783	1463744	4660901
18	75079	1463744	4660901
19	75375	1463744	4660901
20	75671	1463744	4660901
21	75967	1463744	4660901
22	76263	1463744	4660901
23	76559	1463744	4660901
24	76855	1463744	4660901
25	77151	1463744	4660901
26	77447	1463744	4660901
27	77743	1463744	4660901
28	78039	1463744	4660901
29	78335	1463744	4660901
30	78631	1463744	4660901
31	78927	1463744	4660901
32	79223	1463744	4660901
33	79519	1463744	4660901
34	79815	1463744	4660901
35	80111	1463744	4660901
36	80407	1463744	4660901
37	80703	1463744	4660901
38	81000	1463744	4660901
39	81296	1463744	4660901
40	81592	1463744	4660901
41	81888	1463744	4660901
42	82184	1463744	4660901
43	82480	1463744	4660901
44	82776	1463744	4660901
45	83072	1463744	4660901
46	83368	1463744	4660901
47	83664	1463744	4660901
48	83960	1463744	4660901
49	84256	1463744	4660901
50	84552	1463744	4660901
51	84848	1463744	4660901
52	85144	1463744	4660901
53	85440	1463744	4660901
54	85736	1463744	4660901
55	86032	1463744	4660901
56	86328	1463744	4660901
57	86624	1463744	4660901
58	86920	1463744	4660901
59	87216	1463744	4660901
60	87512	1463744	4660901
61	87808	1463744	4660901
62	88104	1463744	4660901
63	88400	1463744	4660901
64	88696	1463744	4660901
65	88992	1463744	4660901
66	89288	1463744	4660901
67	89584	1463744	4660901
68	89880	1463744	4660901
69	90176	1463744	4660901
70	90472	1463744	4660901
71	90768	1463744	4660901
72	91064	1463744	4660901
73	91360	1463744	4660901
74	91656	1463744	4660901
75	91952	1463744	4660901
76	92248	1463744	4660901
77	92544	1463744	4660901
78	92840	1463744	4660901
79	93136	1463744	4660901
80	93432	1463744	4660901
81	93728	1463744	4660901
82	94024	1463744	4660901
83	94320	1463744	4660901
84	94616	1463744	4660901
85	94912	1463744	4660901
86	95208	1463744	4660901
87	95504	1463744	4660901
88	95800	1463744	4660901
89	96096	1463744	4660901
90	96392	1463744	4660901
91	96688	1463744	4660901
92	96984	1463744	4660901
93	97280	1463744	4660901
94	97576	1463744	4660901
95	97872	1463744	4660901
96	98168	1463744	4660901
97	98464	1463744	4660901
98	98760	1463744	4660901
99	99056	1463744	4660901

Deg. 85

Deg. 4 +---			
m	Si	Logarithm	Si
10	75419	1463744	4660901
11	75715	1463744	4660901
12	76011	1463744	4660901
13	76307	1463744	4660901
14	76603	1463744	4660901
15	76899	1463744	4660901
16	77195	1463744	4660901
17	77491	1463744	4660901
18	77787	1463744	4660901
19	78083	1463744	4660901
20	78379	1463744	4660901
21	78675	1463744	4660901
22	78971	1463744	4660901
23	79267	1463744	4660901
24	79563	1463744	4660901
25	79859	1463744	4660901
26	80155	1463744	4660901
27	80451	1463744	4660901
28	80747	1463744	4660901
29	81043	1463744	4660901
30	81339	1463744	4660901
31	81635	1463744	4660901
32	81931	1463744	4660901
33	82227	1463744	4660901
34	82523	1463744	4660901
35	82819	1463744	4660901
36	83115	1463744	4660901
37	83411	1463744	4660901
38	83707	1463744	4660901
39	84003	1463744	4660901
40	84299	1463744	4660901
41	84595	1463744	4660901
42	84891	1463744	4660901
43	85187	1463744	4660901
44	85483	1463744	4660901
45	85779	1463744	4660901
46	86075	1463744	4660901
47	86371	1463744	4660901
48	86667	1463744	4660901
49	86963	1463744	4660901
50	87259	1463744	4660901
51	87555	1463744	4660901
52	87851	1463744	4660901
53	88147	1463744	4660901
54	88443	1463744	4660901
55	88739	1463744	4660901
56	89035	1463744	4660901
57	89331	1463744	4660901
58	89627	1463744	4660901
59	89923	1463744	4660901
60	90219	1463744	4660901
61	90515	1463744	4660901
62	90811	1463744	4660901
63	91107	1463744	4660901
64	91403	1463744	4660901
65	91699	1463744	4660901
66	91995	1463744	4660901
67	92291	1463744	4660901
68	92587	1463744	4660901
69	92883	1463744	4660901
70	93179	1463744	4660901
71	93475	1463744	4660901
72	93771	1463744	4660901
73	94067	1463744	4660901
74	94363	1463744	4660901
75	94659	1463744	4660901
76	94955	1463744	4660901
77	95251	1463744	4660901
78	95547	1463744	4660901
79	95843	1463744	4660901
80	96139	1463744	4660901
81	96435	1463744	4660901
82	96731	1463744	4660901
83	97027	1463744	4660901
84	97323	1463744	4660901
85	97619	1463744	4660901
86	97915	1463744	4660901
87	98211	1463744	4660901
88	98507	1463744	4660901
89	98803	1463744	4660901
90	99099	1463744	4660901
91	99395	1463744	4660901
92	99691	1463744	4660901
93	99987	1463744	4660901
94	100283	1463744	4660901
95	100579	1463744	4660901
96	100875	1463744	4660901
97	101171	1463744	4660901
98	101467	1463744	4660901
99	101763	1463744	4660901

P = Deg. 85

- I want 10 significant digits
- I have an approximation scheme that gives 12
- Usually that's enough to round  $y = x,xxxxxxxxx17 \pm 10^{-12}$
- $y = x,xxxxxxxxx83 \pm 10^{-12}$
- Dilemma when  $y = x,xxxxxxxxx50 \pm 10^{-12}$

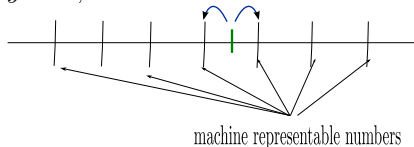


# Correctly rounded functions

- Table Maker's Dilemma

Deg. 4 +---				Deg. 4 +-			
m	Size	Logarithm	Error	m	Size	Logarithm	Error
0	69751	1.663744	1.6609701	1.621	697744.00		
1	70047	1.67193	1.67133	1.670	697744.19		
2	70317	1.67843	1.67722	1.670	697744.38		
3	70567	1.68444	1.68341	1.670	697744.57		
4	70797	1.68999	1.68891	1.670	697744.76		
5	71007	1.69511	1.69377	1.670	697744.95		
6	71197	1.69984	1.69810	1.670	697745.14		
7	71367	1.70421	1.70201	1.670	697745.33		
8	71507	1.70824	1.70551	1.670	697745.52		
9	71627	1.71194	1.70841	1.670	697745.71		
10	71727	1.71531	1.71091	1.670	697745.90		
11	71807	1.71837	1.71301	1.670	697746.09		
12	71867	1.72114	1.71471	1.670	697746.28		
13	71907	1.72364	1.71611	1.670	697746.47		
14	71937	1.72589	1.71721	1.670	697746.66		
15	71957	1.72791	1.71811	1.670	697746.85		
16	71967	1.72971	1.71881	1.670	697747.04		
17	71967	1.73131	1.71931	1.670	697747.23		
18	71957	1.73271	1.71961	1.670	697747.42		
19	71937	1.73391	1.71971	1.670	697747.61		
20	71907	1.73491	1.71971	1.670	697747.80		
21	71867	1.73571	1.71961	1.670	697747.99		
22	71817	1.73631	1.71941	1.670	697748.18		
23	71757	1.73671	1.71911	1.670	697748.37		
24	71687	1.73691	1.71871	1.670	697748.56		
25	71607	1.73691	1.71821	1.670	697748.75		
26	71517	1.73671	1.71761	1.670	697748.94		
27	71417	1.73631	1.71691	1.670	697749.13		
28	71307	1.73571	1.71611	1.670	697749.32		
29	71187	1.73491	1.71521	1.670	697749.51		
30	71057	1.73391	1.71421	1.670	697749.70		
31	70917	1.73271	1.71311	1.670	697749.89		
32	70767	1.73131	1.71191	1.670	697750.08		
33	70607	1.72971	1.71061	1.670	697750.27		
34	70437	1.72791	1.70921	1.670	697750.46		
35	70257	1.72589	1.70771	1.670	697750.65		
36	70067	1.72364	1.70611	1.670	697750.84		
37	69867	1.72114	1.70441	1.670	697751.03		
38	69647	1.71837	1.70261	1.670	697751.22		
39	69407	1.71531	1.70071	1.670	697751.41		
40	69147	1.71194	1.69871	1.670	697751.60		
41	68867	1.70824	1.69661	1.670	697751.79		
42	68567	1.70421	1.69441	1.670	697751.98		
43	68247	1.70000	1.69211	1.670	697752.17		
44	67907	1.69564	1.68971	1.670	697752.36		
45	67547	1.69114	1.68721	1.670	697752.55		
46	67167	1.68651	1.68461	1.670	697752.74		
47	66767	1.68174	1.68191	1.670	697752.93		
48	66347	1.67684	1.67911	1.670	697753.12		
49	65907	1.67181	1.67621	1.670	697753.31		
50	65447	1.66664	1.67321	1.670	697753.50		
51	64967	1.66134	1.67011	1.670	697753.69		
52	64467	1.65591	1.66691	1.670	697753.88		
53	63947	1.65034	1.66361	1.670	697754.07		
54	63407	1.64464	1.66021	1.670	697754.26		
55	62847	1.63881	1.65671	1.670	697754.45		
56	62267	1.63284	1.65311	1.670	697754.64		
57	61667	1.62674	1.64941	1.670	697754.83		
58	61047	1.62051	1.64561	1.670	697755.02		
59	60407	1.61414	1.64171	1.670	697755.21		
60	59747	1.60764	1.63771	1.670	697755.40		
61	59067	1.60101	1.63361	1.670	697755.59		
62	58367	1.59424	1.62941	1.670	697755.78		
63	57647	1.58734	1.62511	1.670	697755.97		
64	56907	1.58031	1.62071	1.670	697756.16		
65	56147	1.57314	1.61621	1.670	697756.35		
66	55367	1.56584	1.61161	1.670	697756.54		
67	54567	1.55841	1.60691	1.670	697756.73		
68	53747	1.55084	1.60211	1.670	697756.92		
69	52907	1.54314	1.59721	1.670	697757.11		
70	52047	1.53531	1.59221	1.670	697757.30		
71	51167	1.52734	1.58711	1.670	697757.49		
72	50267	1.51924	1.58191	1.670	697757.68		
73	49347	1.51101	1.57661	1.670	697757.87		
74	48407	1.50264	1.57121	1.670	697758.06		
75	47447	1.49414	1.56571	1.670	697758.25		
76	46467	1.48551	1.56011	1.670	697758.44		
77	45467	1.47674	1.55441	1.670	697758.63		
78	44447	1.46784	1.54861	1.670	697758.82		
79	43407	1.45881	1.54271	1.670	697759.01		
80	42347	1.44964	1.53671	1.670	697759.20		
81	41267	1.44034	1.53061	1.670	697759.39		
82	40167	1.43091	1.52441	1.670	697759.58		
83	39047	1.42134	1.51811	1.670	697759.77		
84	37907	1.41164	1.51171	1.670	697759.96		
85	36747	1.40181	1.50521	1.670	697760.15		
86	35567	1.39184	1.49861	1.670	697760.34		
87	34367	1.38174	1.49191	1.670	697760.53		
88	33147	1.37151	1.48511	1.670	697760.72		
89	31907	1.36114	1.47821	1.670	697760.91		
90	30647	1.35064	1.47121	1.670	697761.10		
91	29367	1.34001	1.46411	1.670	697761.29		
92	28067	1.32924	1.45691	1.670	697761.48		
93	26747	1.31834	1.44961	1.670	697761.67		
94	25407	1.30731	1.44221	1.670	697761.86		
95	24047	1.29614	1.43471	1.670	697762.05		
96	22667	1.28484	1.42711	1.670	697762.24		
97	21267	1.27341	1.41941	1.670	697762.43		
98	19847	1.26184	1.41161	1.670	697762.62		
99	18407	1.25014	1.40371	1.670	697762.81		
100	16947	1.23831	1.39571	1.670	697763.00		

- I want 10 significant digits
- I have an approximation scheme that gives 12
- Usually that's enough to round  $y = x,xxxxxxxxx17 \pm 10^{-12}$
- $y = x,xxxxxxxxx83 \pm 10^{-12}$
- Dilemma when  $y = x,xxxxxxxxxxx50 \pm 10^{-12}$



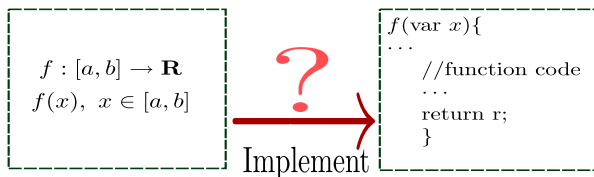
- Increase working precision
- Worst cases search
- V. Lefevre and J-M. Muller

# Correctly rounded functions

- Table Maker's Dilemma - Worst cases search (V. Lefevre and J-M. Muller)
- Example of such a worst case:

$$\log_2(1.0110000101010101010111110111010110001000010110110100*2^{512}) =$$
$$1000000000.011101110000001011010000001111010011011100 \underbrace{1\dots1}_{56} 01\dots$$

# Correctly rounded functions



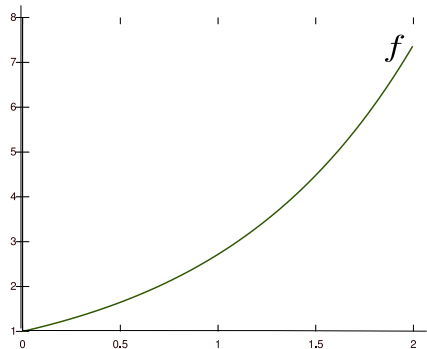
A complex chain of tools:

- Argument reduction
- Compute an approximation polynomial  $p^*$
- Find from  $p^*$  a polynomial  $p$  with fp coeffs
- Bound<sup>3</sup> approximation error:  $(p - f)/f$
- Implement  $p$  in fp arithmetic
- Bound round-off errors

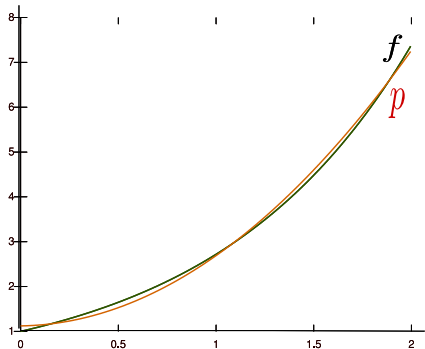
---

<sup>3</sup>S. Chevillard, M. Joldes and C. Lauter, "Certified and fast computation of supremum norms of approximation errors", in 19th IEEE Symposium on Computer Arithmetic, June 2009

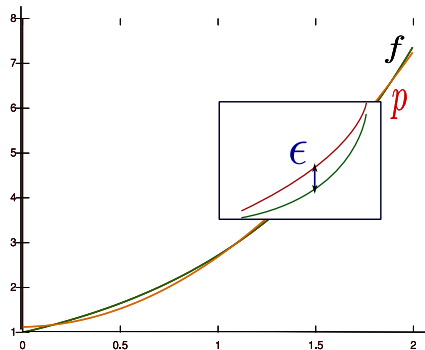
# Supremum Norms of Error Functions



# Supremum Norms of Error Functions



# Supremum Norms of Error Functions

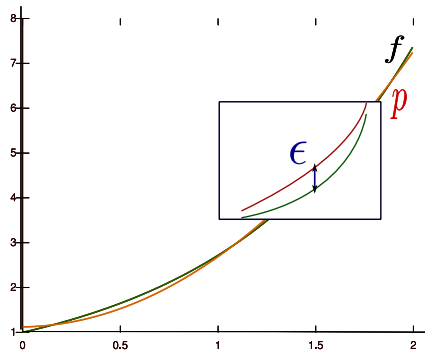


$$\varepsilon(x) = f(x) - p(x), \quad x \in [a, b] \text{ or}$$

$$\varepsilon(x) = \frac{p(x)}{f(x)} - 1, \quad x \in [a, b]$$

$$\text{Define } \|\varepsilon\|_{\infty} = \sup_{x \in [a, b]} \{|\varepsilon(x)|\}$$

# Supremum Norms of Error Functions



$$\varepsilon(x) = f(x) - p(x), \quad x \in [a, b] \text{ or}$$

$$\varepsilon(x) = \frac{p(x)}{f(x)} - 1, \quad x \in [a, b]$$

$$\text{Define } \|\varepsilon\|_{\infty} = \sup_{x \in [a, b]} \{|\varepsilon(x)|\}$$

- Compute a **certified** bound: small interval  $\mathbf{r}$  s.t.  $\|\varepsilon\|_{\infty} \in \mathbf{r}$
- “Quick and dirty” supremum norms - another class of algorithms

# Rigorous computing tools

- Why?
  - Get the correct answer, not an "almost" correct one
  - Bridge the gap between scientific computing and pure mathematics - speed and reliability

# Rigorous computing tools

- Why?
  - Get the correct answer, not an "almost" correct one
  - Bridge the gap between scientific computing and pure mathematics - speed and reliability
- How?
  - Use FP as support for computations (fast)
  - Bound roundoff errors
  - Bound discretization or truncation errors in numerical algorithms
  - Compute **enclosures** instead of **approximations**

# Rigorous computing tools

- Why?
  - Get the correct answer, not an "almost" correct one
  - Bridge the gap between scientific computing and pure mathematics - speed and reliability
- How?
  - Use FP as support for computations (fast)
  - Bound roundoff errors
  - Bound discretization or truncation errors in numerical algorithms
  - Compute **enclosures** instead of **approximations**
- What?
  1. Interval arithmetic (IA)
  2. Taylor models (TM)

# Interval Arithmetic - Thou shalt not lie!

- Each interval = pair of floating-point numbers

# Interval Arithmetic - Thou shalt not lie!

- Each interval = pair of floating-point numbers
- $\pi \in [3.1415, 3.1416]$

# Interval Arithmetic - Thou shalt not lie!

- Each interval = pair of floating-point numbers
- $\pi \in [3.1415, 3.1416]$
- Interval Arithmetic Operations  
Eg.  $[1, 2] + [-3, 2] = [-2, 4]$

# Interval Arithmetic - Thou shalt not lie!

- Each interval = pair of floating-point numbers

- $\pi \in [3.1415, 3.1416]$

- Interval Arithmetic Operations

Eg.  $[1, 2] + [-3, 2] = [-2, 4]$

- Range bounding for functions

Eg.  $x \in [-1, 2], f(x) = x^2 + x + 1$

$$F(X) = X^2 + X + 1$$

$$F([-1, 2]) = [-1, 2]^2 + [-1, 2] + [1, 1]$$

$$F([-1, 2]) = [0, 4] + [-1, 2] + [1, 1]$$

$$F([-1, 2]) = [0, 7]$$

# Interval Arithmetic - Thou shalt not lie!

- Each interval = pair of floating-point numbers

- $\pi \in [3.1415, 3.1416]$

- Interval Arithmetic Operations

Eg.  $[1, 2] + [-3, 2] = [-2, 4]$

- Range bounding for functions

Eg.  $x \in [-1, 2], f(x) = x^2 + x + 1$

$$F(X) = X^2 + X + 1$$

$$F([-1, 2]) = [-1, 2]^2 + [-1, 2] + [1, 1]$$

$$F([-1, 2]) = [0, 4] + [-1, 2] + [1, 1]$$

$$F([-1, 2]) = [0, 7]$$

$$x \in [-1, 2], f(x) \in [0, 7], \text{ but } \text{Im}(f(x)) = [3/4, 7]$$

# Interval Arithmetic - Overestimation

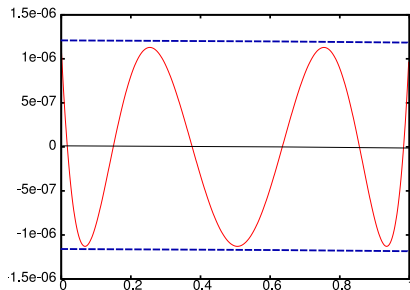
Example:

$$f(x) = e^x, \quad x \in [0, 1]$$

$$p(x) = \sum_{i=0}^5 c_i x^i$$

s.t.  $\|f - p\|_{\infty}$  is as small as possible (Remez algorithm)

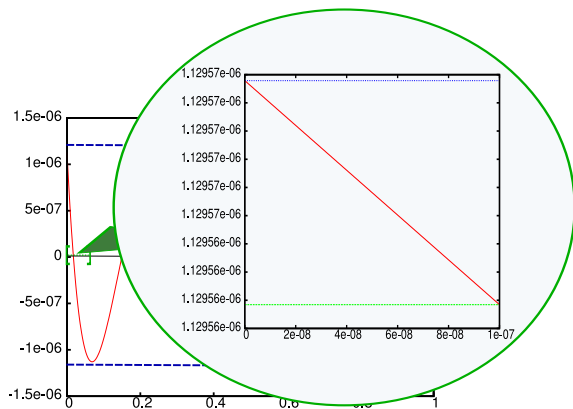
$$\varepsilon(x) = f(x) - p(x)$$



Using IA,  $\varepsilon(x) \in [-0.4, 0.4]$ , but  $\|\varepsilon(x)\|_{\infty} \simeq 1.1295e - 6$ :

# Interval Arithmetic - Overestimation

Overestimation can be reduced by using intervals of smaller width.



In this case, over  $[0, 1]$  we need  $10^7$  intervals!

# Taylor Models - Why do we need them?

## 1. Interval Arithmetic

- ✓ Each variable replaced with an interval (two FP numbers)
- ✓ FP arithmetic is fast, enclosures are reliable
- ✗ Can not identify different occurrences of the same variable

# Taylor Models - Why do we need them?

## 1. Interval Arithmetic

- ✓ Each variable replaced with an interval (two FP numbers)
- ✓ FP arithmetic is fast, enclosures are reliable
- ✗ Can not identify different occurrences of the same variable

## 2. Taylor Models

- Each function replaced with (polynomial, interval remainder)
- Because operations  $(+, -, \times)$  with polynomials are easier

# Taylor Models - How do we obtain them?

Let  $n \in \mathbb{N}$ ,  $n$  times differentiable function  $f$  over  $[a, b]$  around  $x_0$ .

$$\bullet f(x) = \underbrace{\sum_{i=0}^{n-1} \frac{f^{(i)}(x_0)(x-x_0)^i}{i!}}_{T(x)} + \underbrace{\Delta_n(x, \xi)}_{\text{remainder}}$$

# Taylor Models - How do we obtain them?

Let  $n \in \mathbb{N}$ ,  $n$  times differentiable function  $f$  over  $[a, b]$  around  $x_0$ .

$$\bullet f(x) = \underbrace{\sum_{i=0}^{n-1} \frac{f^{(i)}(x_0)(x-x_0)^i}{i!}}_{T(x)} + \underbrace{\Delta_n(x, \xi)}_{\text{remainder}}$$

$$\bullet \Delta_n(x, \xi) = \frac{f^{(n)}(\xi)(x-x_0)^n}{n!}, \quad x \in [a, b], \quad \xi \text{ lies strictly between } x \text{ and } x_0$$

# Taylor Models - How do we obtain them?

Let  $n \in \mathbb{N}$ ,  $n$  times differentiable function  $f$  over  $[a, b]$  around  $x_0$ .

- $f(x) = \underbrace{\sum_{i=0}^{n-1} \frac{f^{(i)}(x_0)(x - x_0)^i}{i!}}_{T(x)} + \underbrace{\Delta_n(x, \xi)}_{\text{remainder}}$
- $\Delta_n(x, \xi) = \frac{f^{(n)}(\xi)(x - x_0)^n}{n!}$ ,  $x \in [a, b]$ ,  $\xi$  lies strictly between  $x$  and  $x_0$
- Compute an interval enclosure  $\Delta$  for  $\Delta_n(x, \xi)$

# Taylor Models - How do we obtain them?

Let  $n \in \mathbb{N}$ ,  $n$  times differentiable function  $f$  over  $[a, b]$  around  $x_0$ .

- $f(x) = \underbrace{\sum_{i=0}^{n-1} \frac{f^{(i)}(x_0)(x - x_0)^i}{i!}}_{T(x)} + \underbrace{\Delta_n(x, \xi)}_{\text{remainder}}$
- $\Delta_n(x, \xi) = \frac{f^{(n)}(\xi)(x - x_0)^n}{n!}$ ,  $x \in [a, b]$ ,  $\xi$  lies strictly between  $x$  and  $x_0$
- Compute an interval enclosure  $\Delta$  for  $\Delta_n(x, \xi)$
- We have:  $f(x) - T(x) \in \Delta, \forall x \in [a, b]$

# Taylor Models - How do we obtain them?

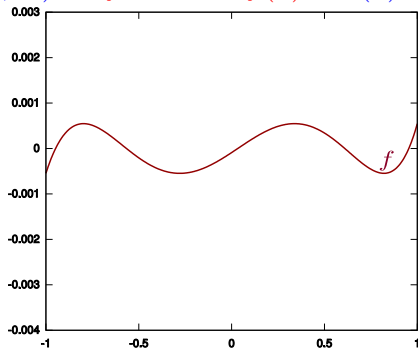
Let  $n \in \mathbb{N}$ ,  $n$  times differentiable function  $f$  over  $[a, b]$  around  $x_0$ .

- $f(x) = \underbrace{\sum_{i=0}^{n-1} \frac{f^{(i)}(x_0)(x - x_0)^i}{i!}}_{T(x)} + \underbrace{\Delta_n(x, \xi)}_{\text{remainder}}$
- $\Delta_n(x, \xi) = \frac{f^{(n)}(\xi)(x - x_0)^n}{n!}$ ,  $x \in [a, b]$ ,  $\xi$  lies strictly between  $x$  and  $x_0$
- Compute an interval enclosure  $\Delta$  for  $\Delta_n(x, \xi)$
- We have:  $f(x) - T(x) \in \Delta, \forall x \in [a, b]$
- The couple  $(T, \Delta)$  is a Taylor model for  $f$  of order  $n$ , over  $[a, b]$

# Taylor Models - How do they look like?

*“A tube around the function”*

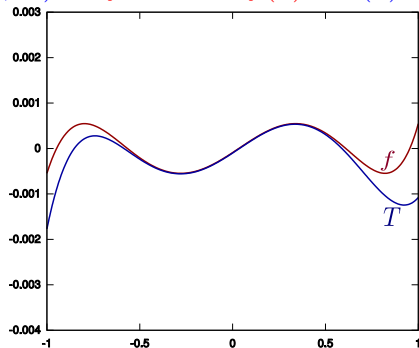
Given a tm  $(T, \Delta)$  for  $f$ , we have  $f(x) - T(x) \in \Delta, \forall x \in [a, b]$



# Taylor Models - How do they look like?

*“A tube around the function”*

Given a tm  $(T, \Delta)$  for  $f$ , we have  $f(x) - T(x) \in \Delta, \forall x \in [a, b]$

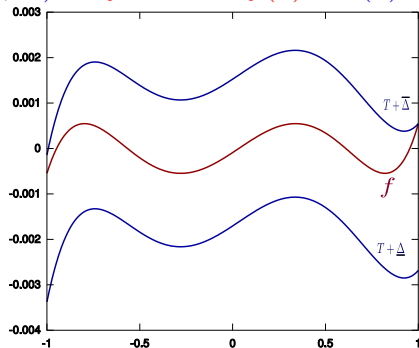


Degree of  $T$ : 5

# Taylor Models - How do they look like?

*“A tube around the function”*

Given a tm  $(T, \Delta)$  for  $f$ , we have  $f(x) - T(x) \in \Delta, \forall x \in [a, b]$

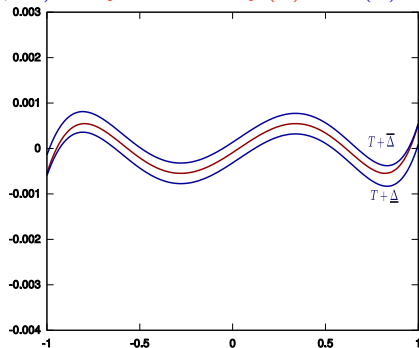


Degree of  $T$ : 5

# Taylor Models - How do they look like?

*“A tube around the function”*

Given a tm  $(T, \Delta)$  for  $f$ , we have  $f(x) - T(x) \in \Delta, \forall x \in [a, b]$

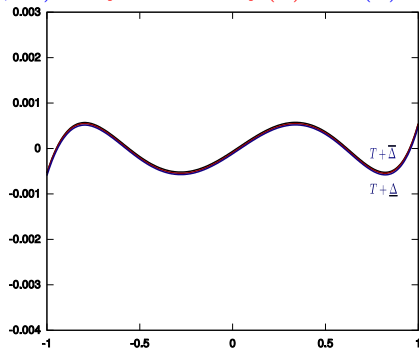


Degree of  $T$ : 6

# Taylor Models - How do they look like?

*“A tube around the function”*

Given a tm  $(T, \Delta)$  for  $f$ , we have  $f(x) - T(x) \in \Delta, \forall x \in [a, b]$



Degree of  $T$ : 7

# Taylor Models

- ✓ Arithmetic operations on TMs are defined
- ✓ Apply to multivariate functions also
- ✓ Reduce overestimation
- ✓ Rigorous computing tool successfully used in global optimization, ODE solving

## Conclusion - What to take home?

- Since 1985 there is an international standard (IEEE-754) for floating-point arithmetic
  - (Almost) All vendors comply with this norm.
  - On all modern computers it is possible to obtain a predictable and portable behaviour for a FP program up to the last digit
  - But, some work is necessary to achieve this
- Implementing elementary functions in FP is a complex process
- Use a library that guarantees correct rounding: CRLibm
- Use rigorous computing tools: Interval Arithmetic, Taylor Models

Questions?

*Do you trust your computer now?*

