Interval arithmetic in GNU Octave

For more information, visit http://www.octave.org/get-involved.html
Read http://www.octave.org/bugs.html to learn how to submit bug reports.
For information about changes from previous versions, type 'news'.
>> pkg load interval
>> sum (infsupdec (magic (3)))
ans = 1×3 interval vector
 [15]_com [15]_com
>> _______

Workspace Command Window Documentation

Oliver Heimlich oheim@posteo.de SWIM 2016, Lyon, France

Outline



IEEE Standard for Interval Arithmeti	v
	c
IEEE Computer Society	
Sponsored by the EEE Microprocessor Standards Committee	
EEE Park Avenue IEEE Std T Nark Avenue Std SA	788 ^w -2015

$$[0,1] + [2,3] = [2,4]$$
$$\frac{[1,\infty]_{dac}}{[0,1]_{com}} = [1,\infty]_{trv}$$
$$exp[0,1] = hull[1,e]$$

The Octave universe Implementation of IEEE Std 1788-2015 Interval arithmetic unit tests

SWIM 2016

What is Octave?

"A free numerical environment mostly compatible with MATLAB"

"free"

- part of the GNU project
- free = libre \neq gratis
- FSF high priority project 2008–2016 to replace MATLAB

"compatible"

- if it works in MATLAB, it should work in Octave
- if it breaks, it is considered a bug
- if it works in Octave, it can break in MATLAB

history

- 1993 started by John W. Eaton
- 1995 structs and plotting
- 2001 packages / toolboxes
- 2007 sparse matrices
- 2011 profiler
- 2015 GUI, 64-bit indexing



```
oliver@Oliver-Tuxedo:~
                                                                                                  Figure 1
>> pkg load interval
                                                                                 File Edit Help
>> demo @infsup/plot3 3
                                                                                 🕑 Z+ Z- 🕂 🔤 🗟 Axes Grid Autoscale
@infsup/plot3 example 3:
 clf
  [x, y] = meshgrid (mince (infsup ("[-5, 5]"), 20), ...
                     mince (infsup ("[0.1, 5]"), 10));
                                                                                   1
 z = log (hypot (x, y));
                                                                                   0
 blue = [38 139 210]/255; base2 = [238 232 213]/255;
                                                                                   -1
 plot3 (x, y, z, base2, blue);
 view (330, 12)
                                                                                   -2
>> figure; demo @infsup/fsolve 2
                                                                                          2
@infsup/fsolve example 2:
                                                                                            1
                                                                                                               2
                                                                                                           0
                                                                                                  -6
clf
hold on
                                                                                                  Figure 2
arid on
shade = [238 232 213] / 255;
                                                                                 File Edit Help
blue = [38 139 210] / 255;
                                                                                 Axes Grid Autoscale
# This 3D ring is difficult to approximate with interval boxes
f = Q(x, y, z) hypot (hypot (x, y) - 2, z);
 [~, paving, inner] = fsolve (f, infsup ([-4; -4; -2], [4; 4; 2]), ...
                                 infsup (0, 0.5), ...
                                                                                   1
                                 optimset ('TolX', 0.2));
                                                                                  0.5
                                                                                   0
plot3 (paving(1, not (inner)), ...
                                                                                  -0.5
        paving(2, not (inner)), ...
                                                                                   -1
                                                                                   -3
        paving(3, not (inner)), shade, blue);
view (50, 60)
                                                                                        -1
                                                                                                                1
                                                                                          0
                                                                                                             0
                                                                                                          -1
>> _
                                                                                                  з
                                                                                                    -3
```

2

What Octave is made of

Community

- mailing lists, wiki, IRC chat
- conference (> 30 participants/yr)
- Summer of Code (3–8 slots/yr)
- very active, friendly, supportive

Code

- 820,000 lines of C/C++
- 730,000 lines of m-scripts
- 60,000 lines of Fortran
- 99 contributors per year

Data from Open HUB Data status: late 2015

Oliver Heimlich

Comment sont maintenues les applications Open Source



CommitStrip.com

Octave Forge

Octave-Forge - Extra packages for GNU Octave

Home · Packages · Developers · Documentation · FAQ · Bugs · Mailing Lists · Links · C

Packages

These packages are meant for current versions of Octave. See the unmaint

bim Package for solving Diffusion Advection Reaction (DAR) Partial Differential Equations details

Repository of free extra packages

- released individually
- by domain experts
- add yours!



OCTAVE/MATLAB routines for working with the biospeckle laser technique

🕨 download details





Common Gatway Interface for Octave download

download



communications

Digital Communications, Error Correcting Codes (Channel Code), Source Code functions, Modulation and Galois Fields

details download

Algorithms for smoothing noisy data

download

data-smoothing



control

Computer-Aided Control System Design (CACSD) Tools for GNU Octave, based on the proven SLICOT Library





database

Interface to SQL databases, currently only postgresql using libpq

SWIM 2016



dataframe

details

Data manipulation toolbox similar to R data

Oliver Heimlich

Octave Forge

Octave-Forge - Extra packages for GNU Octave

Home · Packages · Developers · Documentation · FAQ · Bugs · Mailing Lists · Links

interval

Package Version:	1.4.1
Last Release Date:	2016-02-13
Package Author:	Oliver Heimlich <oheim@posteo.de></oheim@posteo.de>
Package	Oliver Heimlich
Maintainer:	<oheim@posteo.de></oheim@posteo.de>
License:	<u>GPL-3.0+</u>

Download Package (older versions) Function Reference Package Documentation NEWS

Description

The interval package for real-valued interval arithmetic allows one to evaluate functions over subsets of their domain. All results are verified, because interval computations automatically keep track of any errors. These concepts can be used to handle uncertainties, estimate arithmetic errors and produce reliable results. Also it can be applied to computer-assisted proofs, constraint programming, and verified computing. The implementation is based on interval boundaries represented by binary64 numbers and is conforming to IEEE Std 1788-2015, IEEE standard for interval arithmetic.

Details

Dependencies: Oliver Heimlich <u>Octave</u> (>= 3.8.0)

SWIM 2016

... with online documentation

For 3D plotting the <u>@infsup/meshqtid</u> function, as usual, becomes handy. The following example shows how two different ranges for X and Y coordinates are used to construct a grid, where the function <u>@infsup/atan2</u> is evaluated. In this particular case the interval grid has gaps, because X and Y coordinates have been constructed such that intervals do not intersect.

red = [220 50 47] ./ 255; shade = [238 232 213] ./ 255;

x = midrad (1 : 6, 0.25); y = midrad (-3 : 3, 0.25); [x, y] = meshgrid (x, y); z = atan2 (y, x); plot3 (x, y, z, shade, red)

view ([-35, 30])
box off
set (gca, "xgrid", "on", "ygrid", "on", "zgrid", "on")



Octave Online

The OCTAVE ONLINE

```
octave:1> pkg load interval
octave:2> demo @infsup/fsolve 1
@infsup/fsolve example 1:
clf
hold on
arid on
 axis equal
 shade = [238 232 213] / 255;
blue = [38 139 210] / 255;
 cyan = [42 161 152] / 255;
 red = [220 50 47] / 255:
# 2D ring
f = @(x, y) hypot (x, y);
 [outer, paving, inner] = fsolve (f, infsup ([-3; -3], [3; 3]), ...
                                  infsup (0.5, 2), ...
                                  optimset ('TolX', 0.1));
# Plot the outer interval enclosure
plot (outer(1), outer(2), shade)
# Plot the guaranteed inner interval enclosures of the preimage
plot (paving(1, inner), paving(2, inner), blue, cyan);
# Plot the boundary of the preimage
 plot (paving(1, not (inner)), paving(2, not (inner)), red);
```



Run Octave in your browser

- interactive
- sign in to upload script files

Jupyter Notebook

Webapp: Create and share documents with live code

Popular among young scientists (Python generation)

Octave support: github.com/Calysto/octave_kernel

Oliver Heimlich



SWIM 2016

Okay.

Octave is great!

It is 2014.

The draft standard P1788 is ready.

Let me implement it.



- **1** Simplicity but reasonable efficiency
- 2 Easy to use and learn
- 3 Easy to develop new parts
- 4 All famous methods at one place

Jaroslav Horácek Introducing LIME – general interval toolbox for Matlab/Intlab SWIM 2014, Uppsala, Sweden

Oliver Heimlich

DIEEE 1788 CONFORMANCE

1 Simplicity but reasonable efficiency

- 2 Easy to use and learn
- 3 Easy to develop new parts
- 4 All famous methods at one place

Design principles

only one flavor: set-based interval arithmetic

inf-sup data type boundaries in **binary64**

arithmetic carried out by MPFR routines (correctly rounded, no rounding mode switches)



Selected features

implicit constructors

implicit promotion of undecorated intervals

```
>> x = infsup (0, 1)
x = [0, 1]
>> x + 1
ans = [1, 2]
>> x + "[0.5, 0.75]"
ans = [0.5, 1.75]
>> y = infsupdec (0, 1)
y = [0, 1] \_ com
>> x / y
warning: Implicitly
decorated bare interval;
resulting decoration may be
wrong
ans = [0, Inf] trv
```

Selected features

correctly rounded reduction operations on intervals

```
>> A = [1; 2^{-1074}; -1]
\mathbf{A} =
    1.0000e+00
   4.9407e - 324
   -1.0000e+00
>> sum (A)
ans = 0
>> sum (infsup (A))
ans \subset [4.9406e-324, 4.9407e-324]
>> issingleton (ans)
ans = 1
```

Selected features

(by accident) compatible with Intlab to some extend

Introduction to INTERVAL ANALYSIS

Ramon E. Moore Worthington, Ohio

R. Baker Kearfott University of Louisiana at Lafayette Lafayette, Louisiana

Michael J. Cloud Lawrence Technological University Southfield, Michigan

Society for Industrial and Applied Mathematics Philadelphia

Difficulties during implementation

Interval operations:

- 1. reverse pow
- 2. reverse sin, cos, tan, atan2
- 3. periodic: sin, cos, tan

Unknown ULP accuracy of native floating-point operations (systemdependent). $\mathsf{pow1}_1^-(\mathbf{y}, \mathbf{z})$:

				y ∞ [0, 0]
z @ [0, 1]	beforeP	equalP	finishedBy	containsP
overlaps/starts	$[\overline{z}^{1/\underline{y}}, +\infty[$	Ø	$[\overline{z}^{1/\underline{y}}, +\infty[$	$]0,\overline{z}^{1/\overline{y}}] \cup [\overline{z}^{1/\underline{y}},+\infty[$
containedByP	$[\overline{z}^{1/\underline{y}}, \underline{z}^{1/\overline{y}}]$	Ø	$[\overline{z}^{1/\underline{y}}, +\infty[$	$]0,\overline{z}^{1/\overline{y}}] \cup [\overline{z}^{1/\underline{y}},+\infty[$
finishes	$[1, \underline{z}^{1/\overline{y}}]$]0, +∞[]0,+∞[]0,+∞[
equalP/finishedBy	$[1, +\infty[$]0, +∞[]0, +∞[]0, +∞[
containsP/startedBy	$[\overline{z}^{1/\overline{y}}, +\infty[$]0, +∞[]0,+∞[]0,+∞[
overlappedBy	$[\overline{z}^{1/\overline{y}}, \underline{z}^{1/\overline{y}}]$]0, +∞[]0,+∞[]0,+∞[
metBy	$[\overline{z}^{1/\overline{y}}, 1]$]0, +∞[]0, +∞[]0,+∞[
afterP	$[\overline{z}^{1/\overline{y}}, \underline{z}^{1/\underline{y}}]$	Ø]0, <u>z^{1/y}</u>]	$]0, \underline{z}^{1/\underline{y}}] \cup [\underline{z}^{1/\overline{y}}, +\infty[$

Heimlich, Nehmeier, Wolff von Gudenberg – Computing Reverse Interval Power Functions, SCAN 2012

Current state

IEEE 1788 conformance since first release (Jan 2015), much more has already been added since then:

Plotting (2D, 3D)

User manual with examples

Set inversion (SIVIA) and contractor programming

Matrix inversion, matrix exponential, matrix norm, and many more supplementary functions

Oliver Heimlich

SWIM 2016

Distribution

GNU/Linux: your favorite system distribution

Windows: part of Octave Installer

MacPorts, FreshPorts, ...

Octave's package manager: pkg install -forge interval

	real-valued interval arithmetic for Octave			
general	testing migrations			
source: octave- interval (extra, misc)	excuses: • Too young, only 0 of 5 days old			
version: 1.5.0-1	 Updating octave-interval fixes old bugs: #825895 Not considered 			
maintainer: Debian Octave Group (archive)	• Not considered			
uploaders: Oliver	news			
Heimlich	[2016-06-05] Accepted octave-interval 1.5.0-1 (source amd64) into unstable (Oliver Heimlich) (signed by: Sébastien Villemot)			
atchi any	[2016-02-19] octave-interval 1.4.1-1 MIGRATED to testing (Debian testing watch	n)		
VCS: Git (Browse)	[2016-02-13] Accepted octave-interval 1.4.1-1 (source amd64) into unstable (Oliver Heimlich) (signed by: Sébastien Villemot)			
versions 🖪 🖪 🖿	[2016-02-12] Accepted octave-interval 1.4.0-1 (source amd64) into unstable (Oliver Heimlich) (signed by: Sébastien Villemot)			
testing: 1.4.1-1	[2016-02-05] Accepted octave-interval 1.2.0-1 (source amd64) into unstable,			
unstable: 1.5.0-1	unstable (Oliver Heimlich) (signed by: Sébastien Villemot)			

octave-interval



 \bigcirc

Interval unit tests

Test framework development

laTestGen

A unit test generator for implementations of the upcoming IEEE interval arithmetic standard written in Java

M.Jedich, M.Nehmeier, A.Dallmann, J. Wolff von Gudenberg

> Institute of Computer Science University of Würzburg Germany

ITF1788: An Interval Testframework for IEEE 1788

SWIM 2013

Maximilian Kiesner, Marco Nehmeier, and Jürgen Wolff von Gudenberg

Institute of Computer Science, University of Würzburg Am Hubland, D 97074 Würzburg, Germany maximilian.kiesner@stud-mail.uni-wuerzburg.de nehmeier@informatik.uni-wuerzburg.de wolff@informatik.uni-wuerzburg.de

June 17, 2015

Latest activity: github.com/oheim/ITF1788

Oliver Heimlich

Test compiler



Interval test network



Oliver Heimlich

Practical value of additional test cases

📮 benEnsta / pylbex	• Watch → 4 ★ Star • Star • Star
<> Code 🛈 Issues 3 D Pull requests 0 💷 Wiki 4 Pulse 🛄 Graphe	3
Several arithmetic errors #11 ⁽²⁾ Open oheim opened this issue on 22 Nov 2015 - 2 comments oheim commented on 22 Nov 2015	Found: 13 arithmetic errors in 5 functions
 The following errors have been found using pylbex 1.1.3 and are probably derive 1. Interval.ALL_REALS.contains(ω) should be false, because an interval is a no real number (same is true for -∞). 2. pylbex.atan2([0, 0], [entire]) should contain [0, π], but is [empty]. 3. pylbex.atan2([-2, 0.1], [entire]) should contain [-π, π], but is [empty]. 4. pylbex.atan2([-2, 0], [entire]) should contain [-π, π], but is [empty]. 5. pylbex.atan2([-2, 0], [-2, 1]) should contain [-π, π], but is [empty]. 6. pylbex.atan2([0, 1], [entire]) should contain [0, π], but is [empty]. 7. pylbex.atan2([0, 1, 1], [entire]) should contain [0, π], but is [empty]. 8. pylbex.atan2([0, 1, 1], [entire]) should contain [0, π], but is [empty]. 8. pylbex.atan4([1, ∞]) and pylbex.atanh([1, 1]) return [+∞], which is not an in 9. pylbex.sin(Interval(float.fromhex('-0x1.921FB54442D18P+1'), 0.0 possible result. 11. pylbex.bwd_abs([-1.9, 0.2], [-0.2, 0.2]) increases the size of x, which shoul backward arithmetics. 12. pylbex.bwd_pow([-1, 5], 0, [-51, 12]) reduces the size of x, which is wrong. 13. pylbex.bwd_pow([1, 1], 0, [entire]) should produce x = [entire], but is x = [- 	ved from IBEX. None yet subset of the reals and ∞ is Milestone No milestone No milestone Assignees No one assigned Notifications Notifications because you were mentioned. you were happen for 2 participants n, 1]. None yet

Practical value of additional test cases

The following cases of poor accuracy could be improved:

1. pow([empty], 0) returns [1, 1], should be [empty]. 2. pylbex.sqrt([entire]), pylbex.sqrt([0, 1]) and other values for x produce a negative lower boundary, should be zero. 3. pylbex.tan(Interval(0.0, float.fromhex('0X1.921FB54442D18P+0'))) returns [entire], should be [0. 1.63312e+16]. pvIbex.bwd abs(Interval(float.fromhex('-0x1p-1022'), float.fromhex('-0x1p-1022')). Interval.ALL REALS) should produce x = [empty]. Also this function call modifies the content of Interval.ALL REALS, which is bad. 5. pylbex.bwd abs([-1, 0], [entire]) should produce x = [0, 0], but is [-1, 1]. pylbex.bwd abs([-∞, 0], [entire]) should produce x = [0, 0], but is [entire]. 7. pylbex.bwd abs([-∞, -1], [entire]) should produce x = [empty], but is [entire]. 8. pylbex.bwd abs([-∞, 1], [entire]) should produce x = [-1, 1], but is [entire]. 9. Interval(float.fromhex('-0X0.00000000002P-1022'), float.fromhex('0X0.00000000001P-1022')).mid() should use IEEE 754 rounding mode "to nearest, ties to even" and produce 0. 10. pylbex.bwd cosh([empty], $[0, \infty]$) should produce x = [empty], but is x = $[0, \infty]$. 11. pvlbex.bwd cosh([empty], [entire]) should produce x = [empty], but is x = [entire]. 12. pylbex.bwd pow([-1, 0], 0, [-1, 1]) should produce x = [empty], but is x = [-1, 1]. 13. pylbex.bwd pow([0, 0], -1, [-5, 1, 55, 5]) should produce x = [empty], but is x = [0, 0]. 14. pylbex.bwd pow($[-\infty, 0], -3, [5.1, 55.5]$) should produce x = [empty], but is x = [5.1, 55.5]. 15. pylbex.bwd pow([0, ∞], 3, [entire]) should not produce a negative lower boundary. 16. pylbex.bwd pow([-∞, 0], 3, [entire]) should not produce a positive upper boundary. 17. pylbex.bwd pow([-10, 0], -2, [entire]) should produce x = [empty], but is x = [entire]. 18. pylbex.bwd pow([- ∞ , 0], -7, [entire]) should produce x = [- ∞ , 0], but is [entire]

... and 18 additional problems in 7 functions

write tutorials

use Octave for education

share unit tests

Please contribute if you find this useful!

report problems

code

prototype new algorithms in Octave

Conclusion

Standards conforming interval arithmetic readily usable almost everywhere

Easy to use and extend for education and research

Unit tests help to build further implementations or make existing ones standards conforming

Thank you!