

Ruby trunk - Feature #12647

Make tanh faster when it lacks of HAVE_TANH

08/02/2016 09:02 AM - qitar888 (Chia-sheng Chen)

Status:	Closed
Priority:	Normal
Assignee:	
Target version:	
Description	
<p>I found $\tanh(x)$ in <code>math.c</code> return $\sinh(x) / \cosh(x)$, which function call is a waist of time.</p> <p>So I changed to $(\exp(2*x) - 1) / (\exp(2*x) + 1)$, the extract form of <code>tanh()</code>.</p> <p>But if <code>HAVE_SINH</code> and <code>HAVE_COSH</code> are defined, only <code>HAVE_TANH</code> undefined (a weird library but possible) Then the origin formulae are faster.</p> <p>Eventually the solution is to determined whether <code>HAVE_SINH/COSH</code> exists, and set the formulae accordingly.</p>	
Testing Code:	
<pre>require 'benchmark/ips' Benchmark.ips do x x.config(:times => 1000, :warmup => 5) x.report(RUBY_DESCRIPTION) { Math.tanh(12.345) } end</pre>	
Outcome:	
<pre>##### Origin with HAVE_SINH, HAVE_COSH, HAVE_TANH all undef Warming up ----- ruby 2.4.0dev (2016-08-02 trunk 55797) [x86_64-linux] 122.226k i/100ms Calculating ----- ruby 2.4.0dev (2016-08-02 trunk 55797) [x86_64-linux] 1.856M (± 4.9%) i/s - 9.289M in 5.016718s ##### All tag undifined, modified tanh() Warming up ----- ruby 2.4.0dev (2016-08-02 trunk 55797) [x86_64-linux] 165.404k i/100ms Calculating ----- ruby 2.4.0dev (2016-08-02 trunk 55797) [x86_64-linux] 2.857M (± 2.5%) i/s - 14.390M in 5.039981s ##### Only HAVE_TANH undifined Warming up ----- ruby 2.4.0dev (2016-08-02 trunk 55797) [x86_64-linux] 205.575k i/100ms Calculating ----- ruby 2.4.0dev (2016-08-02 trunk 55797) [x86_64-linux] 3.732M (± 2.2%) i/s - 18.707M in 5.015756s</pre>	
<p>About 50% improvement when all three tags are undefined. Doesn't drop when only <code>HAVE_TANH</code> is undefined.</p>	
<p>BTW, I want to ask how to send a patch. Adding a feature here and attaching my patch is enough?</p>	

Associated revisions

Revision 4483d59f - 08/02/2016 12:42 PM - nobu (Nobuyoshi Nakada)

math.c: faster tanh

- math.c (tanh): make faster by the extract form if three hyperbolic functions are unavailable. [Feature #12647]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55799 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55799 - 08/02/2016 12:42 PM - nobu (Nobuyoshi Nakada)

math.c: faster tanh

- math.c (tanh): make faster by the extract form if three hyperbolic functions are unavailable. [Feature #12647]

Revision 55799 - 08/02/2016 12:42 PM - nobu (Nobuyoshi Nakada)

math.c: faster tanh

- math.c (tanh): make faster by the extract form if three hyperbolic functions are unavailable. [Feature #12647]

Revision 55799 - 08/02/2016 12:42 PM - nobu (Nobuyoshi Nakada)

math.c: faster tanh

- math.c (tanh): make faster by the extract form if three hyperbolic functions are unavailable. [Feature #12647]

Revision 55799 - 08/02/2016 12:42 PM - nobu (Nobuyoshi Nakada)

math.c: faster tanh

- math.c (tanh): make faster by the extract form if three hyperbolic functions are unavailable. [Feature #12647]

History

#1 - 08/02/2016 09:03 AM - qitar888 (Chia-sheng Chen)

- Description updated

#2 - 08/02/2016 09:04 AM - qitar888 (Chia-sheng Chen)

- Description updated

#3 - 08/02/2016 12:42 PM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Closed

Applied in changeset r55799.

math.c: faster tanh

- math.c (tanh): make faster by the extract form if three hyperbolic functions are unavailable. [Feature [#12647](#)]

Files

Update-tanh-function-to-be-faster.patch	598 Bytes	08/02/2016	qitar888 (Chia-sheng Chen)
---	-----------	------------	----------------------------