GROMACS - Bug #574

Using autoconf produces zero forces with Altivec

09/28/2010 05:25 PM - Justin Lemkul

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Low</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Erik Lindahl</td>
</tr>
<tr>
<td>Category:</td>
<td>mdrun</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
</tr>
<tr>
<td>Affected version - extra info:</td>
<td>Difficulty:</td>
</tr>
<tr>
<td>Affected version:</td>
<td>uncategorized</td>
</tr>
</tbody>
</table>

Description

Gromacs version 4.5.1 produces zero forces when using Altivec support. From md.log:

Configuring nonbonded kernels...
Configuring standard C nonbonded kernels...
Testing Altivec/VMX support... present.
Configuring PPC/Altivec nonbonded kernels...

The result is that energy minimization fails after a single step:

Steepest Descents converged to Fmax < 1000 in 1 steps
Potential Energy  = -2.4873273e+04
Maximum force     = 0.0000000e+00 on atom 0
Norm of force      = nan

Any resulting simulation crashes immediately, indicating that the structure is insufficiently minimized.

Previous versions of Gromacs worked fine using Altivec support. The mdrun_mpi executable was installed using the following:

```
./configure --prefix=/home/rdiv1001/gromacs-4.5.1 CPPFLAGS="-I/home/rdiv1001/fftw-3.0.1-osx/include"
LDFLAGS="-L/home/rdiv1001/fftw-3.0.1-osx/lib" --disable-threads --without-x --program-suffix=_4.5.1_mpi --enable-mpi
CXXCPP="/nfs/compilers/mpich-1.2.5/bin/mpicxx -E"
make mdrun
make install-mdrun
```

In previous versions of Gromacs, I did not have to specify the CXXCPP environment variable; I don't know if this is related or not.

Related issues:

| Related to GROMACS - Bug #573: CMake doesn't detect Altivec support | Closed | 09/28/2010 |
| Related to GROMACS - Bug #682: Gromacs 4.5.3 fails to compile on ppc | Closed | 01/24/2011 |

Associated revisions

Revision 5ba7125c - 06/16/2012 10:02 PM - Erik Lindahl

New CPU detection & AVX/SSE code, removed raw assembly files.

Removed all raw assembly files and deprecated altivec support.
Removed support for NASM and other assemblers, and replaced previous SSE detection code with a new module using CPUID instead.
Added detection for SSE2, SSE4.1, AVX 128-bit with FMA, and AVX 256-bit.
Added Cmake detection of build platform based on CPUID, and output this to the log file. The executables now compare the compile-time platform and selected acceleration with the run-time platform and most suitable acceleration and warns the user if they do not match. The compiler detection code has also been reordered slightly to produce more readable warnings when OpenMP is not available, and correctly disable pragma warnings.

Added intrinsics code and math functions for SSE2, SSE4.1, AVX128/256
both in single and double precision. All math functions and permutation
code have been tested & verified. Single precision math functions are
correct apart from the least significant bit, and double precision has
roughly twice the accuracy.

This has forced me to temporarily disable the SSE & Fortran acceleration.
SSE will be added back soon based on new intrinsics-only kernels currently
in testing, and we will test if Fortran still makes sense then.

Finally, the patch includes a modification to gmx_rmsdist where
a regression issue was introduced recently by using sqrtf() for
the norm function. This caused the intel compiler to produce slightly
different results at high optimization levels, which got evident here.

Closes #926 - Raw assembly code has been removed.
Refs #923 - Old kernels removed, new will be added shortly.
Fixes #914 - Cmake now does architecture-specific optimization.
Fixes #912, #913
Fixes #857 - We detect rdtsscp support with CPUID and use it if possible.
Fixes #750
Closes #537, #574 - Altivec is now deprecated.

Change-Id: Icfca5a94076b8d82ae67b59c65b2d2ac683256d

Revision 5ba7125c - 06/16/2012 10:02 PM - Erik Lindahl

New CPU detection & AVX/SSE code, removed raw assembly files.
Removed all raw assembly files and deprecated altivec support.
Removed support for NASM and other assemblers, and replaced
previous SSE detection code with a new module using CPUID instead.
Added detection for SSE2, SSE4.1, AVX 128-bit with FMA, and AVX 256-bit.
Added Cmake detection of build platform based on CPUID, and output this
to the log file. The executables now compare the compile-time platform
and selected acceleration with the run-time platform and most suitable
acceleration and warns the user if they do not match. The compiler
detection code has also been reordered slightly to produce more readable
warnings when OpenMP is not available, and correctly disable pragma
warnings.

Added intrinsics code and math functions for SSE2, SSE4.1, AVX128/256
both in single and double precision. All math functions and permutation
code have been tested & verified. Single precision math functions are
correct apart from the least significant bit, and double precision has
roughly twice the accuracy.

This has forced me to temporarily disable the SSE & Fortran acceleration.
SSE will be added back soon based on new intrinsics-only kernels currently
in testing, and we will test if Fortran still makes sense then.

Finally, the patch includes a modification to gmx_rmsdist where
a regression issue was introduced recently by using sqrtf() for
the norm function. This caused the intel compiler to produce slightly
different results at high optimization levels, which got evident here.

Closes #926 - Raw assembly code has been removed.
Refs #923 - Old kernels removed, new will be added shortly.
Fixes #914 - Cmake now does architecture-specific optimization.
Fixes #912, #913
Fixes #857 - We detect rdtsscp support with CPUID and use it if possible.
Fixes #750
Closes #537, #574 - Altivec is now deprecated.

Change-Id: Icfca5a94076b8d82ae67b59c65b2d2ac683256d

History
#1 - 09/22/2011 04:08 PM - Rossen Apostolov
- Description updated
- Assignee deleted (Erik Lindahl)
- Target version changed from 4.5.1 to 4.5.6

#2 - 03/05/2012 11:08 AM - Rossen Apostolov
Hi,

I'll try to have a quick look at this prior to 4.6, but since PowerPC hardware is getting pretty old now the Altivec kernels are likely to be deprecated quite soon.

Erik Lindahl wrote:

Hi,

I'll try to have a quick look at this prior to 4.6, but since PowerPC hardware is getting pretty old now the Altivec kernels are likely to be deprecated quite soon.

I agree. In fact, the supercomputer here that manifests this problem will be retired in July, so even if it's fixed I will have no way to test it. I wouldn't be offended at all if this were simply closed due to the antiquated nature of the hardware. It won't be a problem for us any longer, anyway.

Then let's drop Altivec support

Closed by 5ba7125c.

- Status changed from In Progress to Closed