Effect of non-IEEE-compliant optimizations on WRF numerical results

Gerardo Cisneros*, Scott R. Dembek†, Jimy Dudhia‡
and John S. Kain§

Abstract

An Intel® Fortran compiler flag that relaxes IEEE arithmetic rules for divisions and square roots has been a part of the `arch/configure.defaults` file for the Intel® Itanium®-based SGI Altix since WRF version 2.0.2. At the time of its introduction, use of the flag resulted in substantial performance gains at apparently little loss in precision. Since then, several dozen upgrades and four major revisions of the compiler have been released. Recent tests with WRF version 2.2 and the latest upgrades of the three most recent major revisions show that continued use of the compiler flag can result in substantial numerical differences, especially with the later compilers, but its removal incurs only a modest loss of performance. An account of the testing and the results is given.

1 Introduction

The Intel® Fortran compiler has been the compiler of choice for SGI Altix systems that use the Intel® Itanium® processor since their introduction. Starting with WRF V2.0.2 and Intel® Fortran 8.1, a set of undocumented options that allowed the compiler to apply non-IEEE-compliant optimizations to divisions, reciprocals, square roots, and reciprocals of square roots was introduced into those sections of WRF's `arch/configure.defaults` file used to configure WRF for the SGI Altix. Starting with WRF V2.1.1, the set of four options was replaced by a documented option, `-IPF_fp_relaxed`, that provided the same non-IEEE-compliant optimizations, and the option has remained in place[1] through WRF version V2.2.1. In the following the set of four undocumented options and their documented replacement, `-IPF_fp_relaxed`, are referred to as the "relaxed arithmetic option".

At the time of its introduction, the relaxed arithmetic option resulted in noticeable gains in performance[2] with little loss of precision, the latter judged

*Silicon Graphics, S.A. de C.V., Av. Vasco de Quiroga 3000 PB, 01210 México, DF, México; e-mail: `gerardo@sgi.com`

†e-mail: `scott.a.dembek@nasa.gov`

‡NCAR; e-mail: `dudhia@ucar.edu`

§NOAA NSSL; e-mail: `Jack.Kain@noaa.gov`

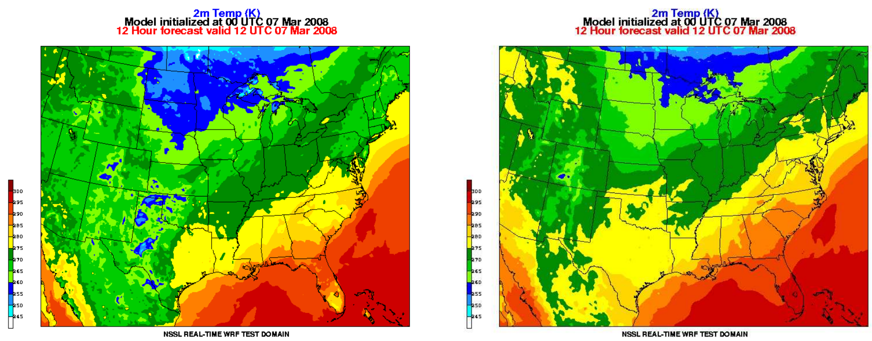

Figure 1: 2m Temperature for 2003_03_07-12:00 using Intel® Fortran 10.1.013 (left) and 10.0.026 (right), in both cases with the relaxed arithmetic option. The difference between the ends of a given color zone is 5K.

by validation criteria set forth by various procurement-related benchmarks. Recent testing with the latest Intel® Fortran compiler, however, indicates that the use of the relaxed arithmetic option is not needed for performance and can lead to serious numerical deviations. We conducted tests of WRF V2.2 with three different versions of the Intel® Fortran compilers and three different options affecting compliance with IEEE arithmetic rules, and report on numerical deviations from the results obtained using the strictest option, as well as on performance variations observed among the various versions and options.

2 Testing

As part of the testing of a newly-installed release of the latest Intel® Fortran compiler (version 10.1.013) on the SGI Altix at the NOAA National Severe Storms Laboratory (NSSL), a 12-hour forecast was run using WRF V2.2 over the NSSL real-time test domain at 16 km resolution (horizontal grid size 244×186, 35 vertical levels, 80s integration timestep, hourly history outputs). The 2m temperatures were found to be unusually low with respect to results from the same version of WRF compiled with an older compiler, version 10.0.026, which were deemed to be very close to reality. Figure 1 shows the 2m temperature plots for the two test runs. The color coding makes it evident that there are regions that differ by 10K or more between compiler versions.

The next two experiments consisted in replacing the relaxed arithmetic option with `-fp-model precise` (the "precise arithmetic" option) in builds using either compiler; Figure 2 shows that both versions gave substantially the same results, and both are closer to the results for the 10.0.026 compiler with the relaxed arithmetic option, although there are still some small regions with differences approaching 5K.

Since precise arithmetic is known to affect performance adversely, two more

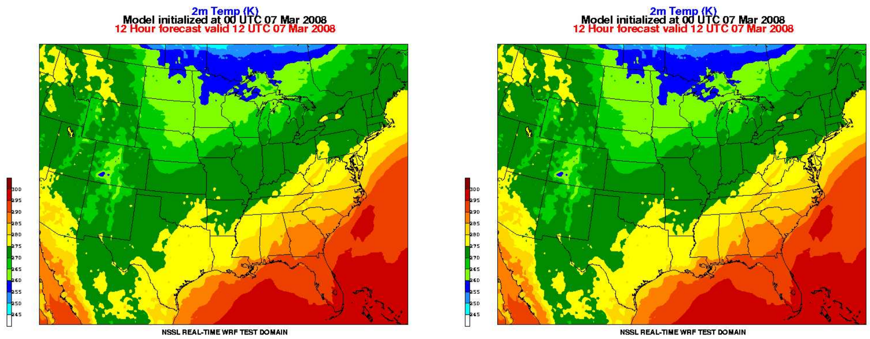

Figure 2: 2m Temperature for 2003_03_07-12:00 using Intel® Fortran 10.1.013 (left) and 10.0.026 (right), in both cases with the precise arithmetic option.

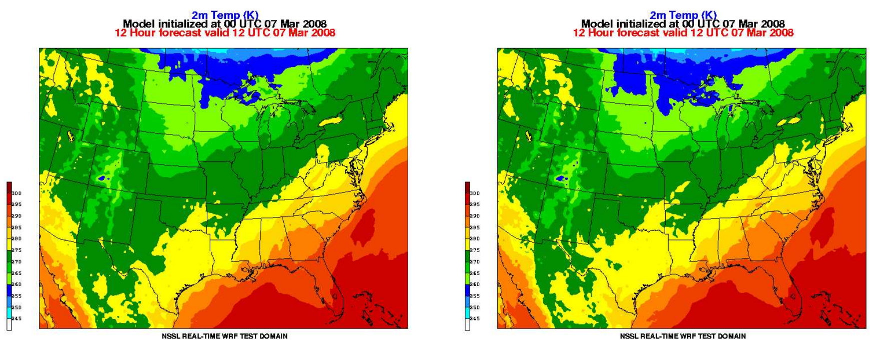

Figure 3: 2m Temperature for 2003_03_07-12:00 using Intel® Fortran 10.1.013 (left) and 10.0.026 (right), in both cases with the default arithmetic option.

builds were done with either compiler, omitting the arithmetic-related options (in what follows, the "default arithmetic" option). The results are shown in Figure 3; some differences are apparent, but nothing as severe as with the relaxed arithmetic results. Also, the default arithmetic results using the 10.1.013 compiler appear to be closer to the results using precise arithmetic than the results from the default arithmetic executable built with the 10.0.026 compiler.

To further analyze differences caused by the various arithmetic options, three executables of WRF 2.2 were built with them using the Intel® Fortran 9.1.052 compiler (the last update for version 9.1). The results from running them on the same input data are shown in Figure 4. For this version of the compiler, all three plots are almost visually indistinguishable, and `diffwrf` confirms that this should be the case; for the T2 field, the pointwise maximum between any two of the 12-h output files is smaller than 0.005K.

The `diffwrf` utility that is built alongside WRF provides statistics that quantify the qualitative differences observed in the figures above; selected statis-

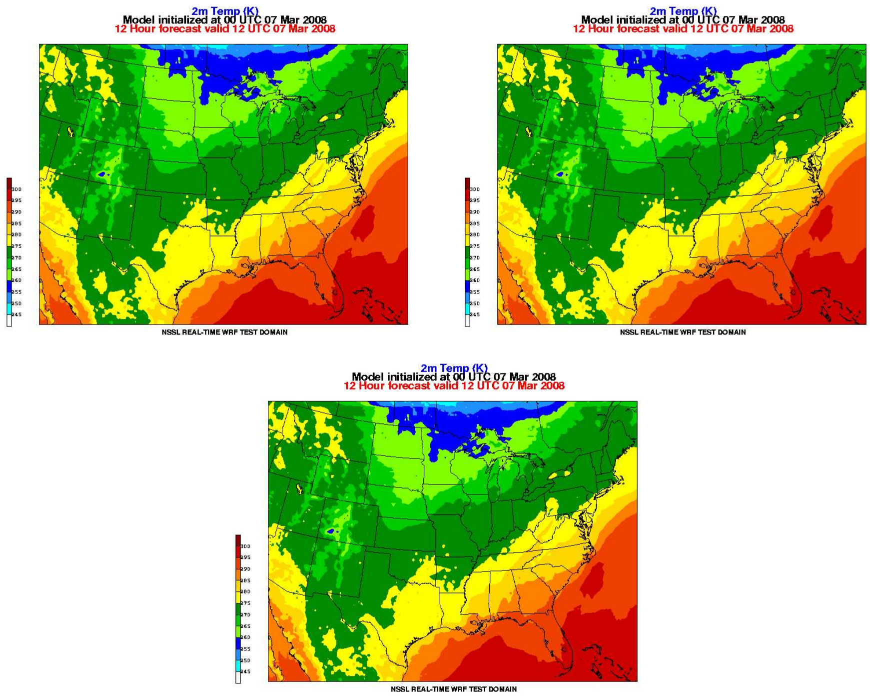

Figure 4: 2m Temperature for 2003_03_07-12:00 using Intel® Fortran 9.1.052 and the precise arithmetic option (top left), the relaxed arithmetic option (top right) and the default arithmetic option (bottom).

tics for the 2m temperature field are shown in Table 1.

To investigate the extent to which the temperature anomaly caused by use of the relaxed arithmetic option persists in the vertical direction, sounding images were compared. Figure 5 shows two sounding images valid at Amarillo, TX, at 1200 UTC 7 March. The first is a 12h model forecast from the NSSL realtime run using Intel® Fortran 10.1.0.13 with the relaxed arithmetic option at the full 4km resolution, and the second is the observed sounding. The model sounding shows that the temperature anomaly appears to be limited to the lowest one or two model layers (the lowest model level is about 35m above the ground). Also, although not evident from this single sounding location, it appears that the problem only occurs under clear sky; in areas where the sounding is near saturation at any level, there does not seem to be a problem.

With this information we isolated the source of the anomaly by examining the surface radiation via the GLW field; this led to the observation that certain indices into look-up tables used by subroutine `RTRN` of `phys/module_ra_rrtm.F` were being computed from elements of a floating-point array containing IEEE NaNs, which turned out to be due a compiler defect in Intel Fortran version

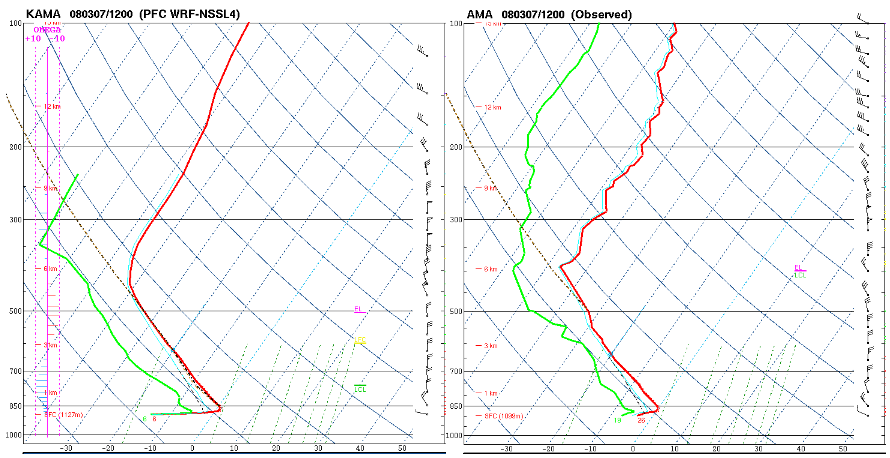

Figure 5: Soundings valid at Amarillo, TX, at 1200 UTC 7 March, from model forecast using Intel® Fortran 10.1.013 and the relaxed arithmetic option (left), and observed (right).

10.1.013 (which persists in version 10.1.015 as well) that generates incorrect code in the loop in subroutine `SETCOEF` of the same module when the relaxed arithmetic option is used in combination with `-O3`.

3 Performance

The initial observations (using the 10.0.026 and 10.1.013 compilers) were made from history output files produced by runs done an SGI Altix at NOAA NSSL, whose resources were being shared with other users at the time the jobs were run. The runs using the 9.1.052 compiler were carried out on 24 processors of an otherwise idle SGI Altix 3700 BX2 with 1.6GHz/9MBL3 Itanium2® ("Madison 9M") processors, and the runs using 10.0.026 and 10.1.013 compilers were repeated on the same system for performance measurement purposes. Table 2 shows the elapsed wallclock times for all nine 12-h runs. Differences of less than 1% in the elapsed times are negligible.

The timings show that there is very little degradation in performance from simply removing the `-IPF_fp_relaxed` option, but there is noticeable degradation when using `-fp-model precise`.

4 Conclusion

Tests with the older 9.1.052 compiler indicate that the relaxed arithmetic option is fairly safe to use with that version. However, in version 10.0.026 numerical differences are noticeable, and with the latest compiler (10.1.013) the differences are even larger, making use of the option totally unacceptable for WRF. Since

Compiler version	Option	RMS value of T2 field	T2 RMS error vs. 9.1.052 precise
9.1.052	precise	276.7326	0
	default	276.7324	0.01839
	relaxed	276.7326	0.02254
10.0.026	precise	276.7239	0.05840
	default	276.8886	0.6579
	relaxed	276.8550	0.3010
10.1.013	precise	276.7241	0.05387
	default	276.8228	0.1472
	relaxed	273.8442	4.438

Table 1: Selected T2 statistics from `diffwrf`.

Compiler version	Option	Wallclock s
9.1.052	precise	442
	default	393
	relaxed	388
10.0.026	precise	453
	default	368
	relaxed	356
10.1.013	precise	442
	default	357
	relaxed	358

Table 2: Wallclock times in seconds for 12-h runs with hourly history outputs.

the anomalies appeared to be limited to the lowest model layers under clear-sky conditions, the surface radiation was examined. This led to the root cause of the anomaly to be a compiler defect that generates incorrect code in one subroutine's loop when `-IPF-fp-relaxed` and `-O3` are used together.

Acknowledgments

The real-time forecasting effort is an activity of the NOAA Hazardous Weather Testbed. The authors would like to express their thanks to Brett Morrow of INDUS Corporation and the National Severe Storms Laboratory. Gerardo Cisneros would like to thank George R. Carr, Jr., for comments that improved the readability of the paper. Scott R. Dembek would like to acknowledge the support of the Cooperative Institute for Mesoscale Meteorological Studies.

References

[1] R. Dubtsov, A. Semenov, D. Shkurko, “WRF performance on Intel® Processors”, Eighth WRF Users’ Workshop, Boulder, CO, June 2007.

[2] A. Semenov, T. Kashevarova, P. Mankevich, D. Shkurko, K. Arturov and N. Panov, “WRF performance tuning for platforms based upon the Intel® Itanium® Processor”, Joint WRF/MM5 Users’ Workshop, Boulder, CO, June 2005.