

USING *openGR* FOR NUMERICAL RELATIVITY

P. WALTER*, J. ALLEN, R. A. MATZNER

*Center for Relativity, Department of Physics, University of Texas at Austin,
Austin, TX 78712, USA*

**E-mail: pwalter@physics.utexas.edu*

M. ANDERSON

*Math Department, Brigham Young University,
Provo, UT 84602, USA*

A. NEROZZI

*Centro Multidisciplinar de Astrofísica - CENTRA, Dept. de Física, Instituto Superior Técnico,
Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal*

U. SPERHAKE

*California Institute of Technology,
Pasadena, CA 91109, USA*

This article focuses on the scaling performance of the evolution portion of *openGR*, a code for Numerical Relativity. As expected, *openGR* scales considerably better for unigrid domains than it does for domains using Fixed Mesh Refinement (FMR).

Keywords: Numerical Relativity; Proceedings; World Scientific Publishing.

1. Introduction

We discuss how *openGR*, a *SAMRAI*-based finite difference code for General Relativity, scales by investigating both strong and weak scaling.¹⁻⁴ A relation between the wall clock time t of a simulation and the number of processors n can be written as $t = an^b$, where b represents the slope on a log-log graph.

Strong scaling consists of measuring performance by running the same simulation on a varying number of processors. The ideal case corresponds to a slope b of -1 on a log-log plot, giving $t \sim \frac{1}{n}$. Eventually, a limit will be reached where the communication overhead per processor (ghostzone cells) will become comparable to the size of the job per processor (non-ghostzone cells), corresponding to a leveling off of the wall clock time.

Weak scaling consists of varying the overall size of the simulation such that each processor is running the same size of job. Deviations away from the ideal case of zero slope place a limit on the size of the job that can be run.

Since the majority of wall clock time of a simulation is spent evolving a given spacetime, we present scaling results of the wall clock time spent in PVODE (the evolution portion of *openGR*). All of these examples were run on the TACC computer *Ranger*.

2. Single Puncture Scaling on Unigrid Domain

Figure 1a shows the strong scaling of simulations of a single puncture (a single black hole) at rest in a unigrid domain. Each simulation was carried out to a time $t = 4.5M$, where M is the mass of the puncture. Details are shown in Table 1.

Figure 1b shows the weak scaling of the simulations of a single puncture at rest. The average slope for all the curves combined is .23. For simulations of 1,536 processors or less, the slope is .09. We conclude that unigrid simulations run with good scaling efficiency at least up to 4,096 processors.

Table 1. Unigrid simulations of a single puncture at rest evolved to $t = 4.5M$, where M is the mass of the puncture. The physical domain is $-100M$ to $100M$ in x, y, z .

Job	A	B	C	D	E
Resolution	$5M$	$2.5M$	$1.25M$	$\frac{5M}{8} = .625M$	$\frac{5M}{16} = .3125M$
# of Points	40^3	80^3	160^3	320^3	640^3
<i>Strong Scaling:</i>	Proc. Range		16 – 256	32 – 1024	512 – 4096
	Slope		-.95	-.89	-.85
<i>Weak Scaling:</i>	Average	Slope	.23		

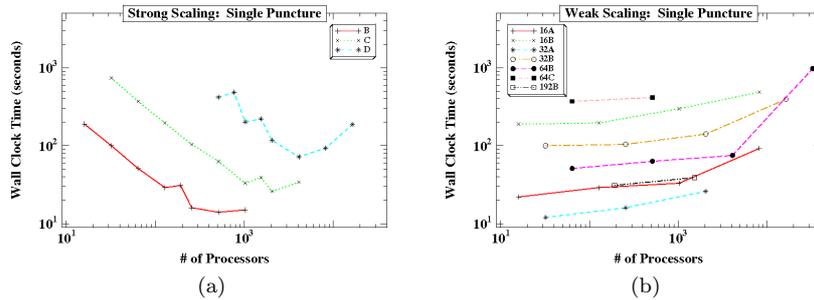


Fig. 1. Strong and weak scaling runs for single puncture evolutions in a unigrid domain (see Table 1). (a) Strong scaling for runs of jobs B, C, and D, where each curve represents the same job being carried out on varying numbers of processors. (b) Weak scaling simulations where each subsequent data point on curve represents a job containing eight times the number of points carried out on eight times the number of processors. The labels for the curves given in the key define the number of processors and the job being run for the leftmost data point on the curve. For instance, on the curve 64B, the leftmost data point is 64 processors running job B. The next point on the curve is 512 processors running job C, then 4,096 processors running job D and 32,768 processors running job E.

3. Scaling of Two Punctures with FMR

Figure 2a shows the strong scaling of simulations of two punctures, initially at rest, which then infall. All simulations were carried out to $t = .2M$. Here, M is the sum of the masses of both punctures. Details are given in Table 2. Results show consistent

behavior up to $\sim 1.5k$ processors, reflecting the lack of ideal but acceptable scaling. For those jobs, runs greater than $4k$ processors clearly have no scaling advantage. Figure 2b shows weak scaling where the average slope of all three curves combined is .62. We are investigating the reason for the poorer scaling in the FMR case.

Table 2. Simulations of two equal-mass punctures initially at rest with nine levels of mesh refinement, the two finest of which track the punctures. M is the total mass of the two punctures. Each subsequent job has twice the resolution throughout the domain. Each level is refined by a factor of two. Simulations evolved to $t = .2M$.

Physical Domain	[$-100M, 100M$] in x,y,z		
Puncture Locations	($-\frac{3M}{2}, 0, 0$) and ($\frac{3M}{2}, 0, 0$)		
Job	A	B	C
Resolution of Coarsest Grid	$4M$	$2M$	M
Resolution of Finest Grid	$\frac{M}{64}$	$\frac{M}{128}$	$\frac{M}{256}$
<i>Strong Scaling</i> : Proc. Range	16 – 192	192 – 1536	4096 – 16384
Slope	-.63	-.5	-.03
<i>Weak Scaling</i> : Avg. Slope	.62		

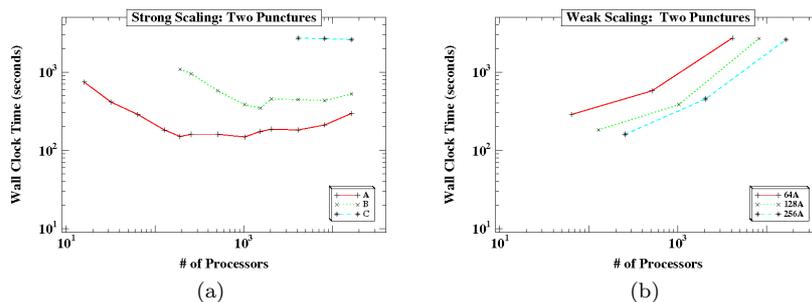


Fig. 2. Strong and weak scaling runs for evolutions of two punctures with 9 levels of fixed mesh refinement. The two finest levels are moving boxes that track and stay centered on each puncture (see Table 2). (a) Strong scaling for runs A, B and C, where points on a given curve represent the same job for a varying number of processors. (b) Weak scaling simulations where each subsequent data point on curve represents a job containing roughly eight times the number of points carried out on eight times the number of processors.

References

1. A. M. Wissink, R. D. Hornung, S. R. Kohn, S. S. Smith and N. S. Elliott, Large Scale Structured AMR Calculations Using the SAMRAI Framework, in *SC01 Proceedings*, (Denver, CO, 2001). Also available as LLNL technical report UCRL-JC-144755.
2. M. Anderson, Constrained Evolution in Numerical Relativity, PhD thesis, University of Texas at Austin, (TX, USA, 2004).
3. M. Anderson and R. A. Matzner, *Found. of Phys.* **35**, 1477 (2005).
4. P. Walter, Using openGR for Numerical Relativity, PhD thesis, University of Texas at Austin, (TX, USA, 2009).