CgWave: Composite Grid Wave Equation Solver.
User Guide and Reference Manual

W. D. Henshaw[a,2,*]

[a] *Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180, USA*

Abstract

Here is the user guide and reference manual for CgWave, a composite grid wave equation solver based on Overture. CgWave is meant to be a simple example of an efficient Overture based PDE solver that also runs in parallel. CgWave is also used by CgWaveHoltz to compute solutions to the time-harmonic wave equation (i.e. Helmholtz equation) using Daniel Appelö's WaveHoltz approach.

Keywords: Wave equation; overset grids.

Contents

1. Introduction

Note: This is a work in progress. Note all features are implemented. yet.

Here is the user guide and reference manual for CgWave, a composite grid wave equation solver based on Overture. CgWave solves the wave equation in second-order form using overset grids. CgWave solves problems in two and three space dimensions.

Here is a citation [1].

Things to do:

1. Add 4th order compatibility conditions.
2. Sixth and eight-order accurate BCs.
3. Finish 3D version of CgWave.
4. Finish upwind-dissipation for implicit time-stepping

---

* Department of Mathematical Sciences, Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180, USA. *Email address:* `henshw@rpi.edu` (W. D. Henshaw)

[1] This work was performed under DOE contracts from the ASCR Applied Math Program.

[2] Research supported by the National Science Foundation under grant DMS-1519934.

[3] Research supported by a U.S. Presidential Early Career Award for Scientists and Engineers.

2. Governing Equations

CgWave solves the initial boundary-value problem for the wave equation in second-order form for $u = u(\mathbf{x}, t)$,

$$\partial_t^2 u = c^2 \Delta u + f(\mathbf{x}, t), \qquad \text{for } \mathbf{x} \in \Omega,\ t > 0, \tag{1a}$$

$$Bu = g \qquad \text{for } \mathbf{x} \in \partial\Omega,\ t > 0, \tag{1b}$$

$$u(x, 0) = u_0(\mathbf{x}), \quad \partial_t u(\mathbf{x}, 0) = u_1(\mathbf{x}), \text{for } \mathbf{x} \in \partial\Omega. \tag{1c}$$

Here $Bu = g$ denotes some boundary conditions. Boundary conditions include

- Dirichlet
- Neumann
- Mixed
- Even symmetry
- Radiation boundaries (far field) – to-do.

3. Numerical Scheme

The equations 1 are advanced using a modified equation approach [2]. Second and fourth-order accurate schemes are currently available.

At fourth-order accuracy the scheme takes the form

$$D_{+t} D_{-t} U_{\mathbf{i}}^n = c^2 \Delta_{4h} U_{\mathbf{i}}^n + \frac{\Delta t^2}{12} (c^2 \Delta_{2h})^2 U_{\mathbf{i}}^n + f(\mathbf{x}_{\mathbf{i}}, t^n) + \frac{\Delta t^2}{12} \Big(c^2 \Delta_{2h} f(\mathbf{x}_{\mathbf{i}}, t^n) + \partial_t^2 f(\mathbf{x}_{\mathbf{i}}, t^n) \Big)$$

where Δ_{ph} is a p-th order accurate approximation to Δ.

4. Upwind dissipation

Upwind dissipation for the wave equation in second-order form was first discussed in [3] and later extended to Maxwell's equations in [4]. The simplified version presented here is developed in a paper not yet completed [5].

At the continuous level, the upwind dissipation adds a term proportional to a high spatial derivative of $\partial_t u$, and roughly takes the form in one-dimension as

$$\partial_t^2 u = c^2 \Delta u - \nu_p \, \frac{c}{\Delta x} (-\Delta x^2 \partial_x^2)^q \, \partial_t u, \tag{2}$$

for some coefficient ν_q, and where $q = p/2 + 1$ is defined in terms of the order of accuracy of the scheme p. To avoid a time-step restriction, upwind dissipation is added using a predictor-corrector scheme (UWPC). Let us describe the approach in one space dimension. The predictor consist of the usual modified equation update to determine the predicted value $u_j^p \approx u_j^{n+1}$,

$$u_j^p = 2u_j^2 - u_j^{n-1} + \Delta t^2 \Big(c^2 D_{h,xx} u_j^n + \ldots), \tag{3}$$

$$\text{applyBoundaryConditions}(u^p). \tag{4}$$

The dissipation is added in a corrector step where $\partial_t u^n$ is approximated with D_{0t},

$$u^{n+1} = u^p - \nu_p \, \lambda \, (-\Delta_+ \Delta_-)^q \Big(\frac{u^p - u^{n-1}}{2} \Big), \tag{5}$$

$$\text{applyBoundaryConditions}(u^{n+1}), \tag{6}$$

where ν_q is the coefficient of the upwind dissipation and λ is the CFL parameter,

$$\lambda \stackrel{\text{def}}{=} \frac{c\Delta t}{\Delta x}. \tag{7}$$

The stability condition turns out to be [5] **check me**

$$\lambda < 1, \tag{8}$$
$$\zeta < 2, \tag{9}$$
$$\zeta \stackrel{\text{def}}{=} \nu_p\,\lambda\,(4\sin^2(\xi/2))^q \tag{10}$$

which implies we need the usual CFL condition, $\lambda < 1$ as well as the restriction on ν_p

$$\nu_p < \frac{2}{\lambda\,4^q} = \frac{1}{\lambda\,2^{p+1}} \tag{11}$$

In two-dimensions **check me**

$$\zeta \stackrel{\text{def}}{=} \nu_p\,\Big((\lambda_x\,(4\sin^2(\xi_x/2))^q + \lambda_y(4\sin^2(\xi_y/2))^q\Big) \tag{12}$$

and we need $\zeta < 2$ or

$$\nu_p\,\Big((\lambda_x\,(4\sin^2(\xi_x/2))^q + \lambda_y(4\sin^2(\xi_y/2))^q\Big) < 2, \tag{13}$$

or

$$\nu_p < \frac{2}{\lambda_x 4^q + \lambda_y 4^q} = \frac{1}{2^{p+1}}\,\frac{1}{\lambda_x + \lambda_y} \tag{14}$$

In two-dimensions the CFL condition is

$$\lambda_x^2 + \lambda_y^2 < 1 \tag{15}$$

which implies $\lambda_x + \lambda_y < \sqrt{2}$. Whence

$$\nu_p < \frac{1}{\sqrt{2}}\,\frac{1}{2^{p+1}} \tag{16}$$

In three-dimensions

$$\nu_p < \frac{1}{2^{p+1}}\,\frac{1}{\lambda_x + \lambda_y + \lambda_z} \tag{17}$$

where

$$\lambda_x^2 + \lambda_y^2 + \lambda_z^2 < 1 \tag{18}$$

which implies $\lambda_x + \lambda_y + \lambda + z < \sqrt{3}$ and

$$\nu_p < \frac{1}{\sqrt{3}}\,\frac{1}{2^{p+1}} \tag{19}$$

Summary: In d-dimensions we require

$$\nu_p < \frac{1}{\sqrt{d}}\,\frac{1}{2^{p+1}}. \tag{20}$$

Note. The value of $\nu_2 = 1/8$ suggested in [5] seems to be too big in 2D or 3D, instead we seem to need

$$\nu_2 = \frac{1}{\sqrt{d}}\,\frac{1}{8} \tag{21}$$

in d-dimensions which is smaller than $1/8$ for $d = 2, 3$. This conclusion agrees with computations.

For fourth-order, $p = 2$, we require

$$\nu_4 < \frac{1}{\sqrt{d}} \frac{1}{32} \tag{22}$$

The suggested value for $\nu_4 = 5/288 = 1/(57.6)$ Now $32\sqrt{3} \approx 55.456$ and thus this suggested value of $\nu_4 = 5/288$ should work in 2D or 3D. This conclusion also agrees with computations.

Note: We could choose ν_p from the condition $\zeta < 2$ - choose a value slightly less than the largest value allowed by stability.

An alternative scheme which allows a bigger value for ν_p is to evaluate $D_{0t} U^n$ in a Gauss-Seidel fashion. The predictor sets a preliminary value for u_j^{n+1},

$$u_j^{n+1} = 2u_j^2 - u_j^{n-1} + \Delta t^2 \Big(c^2 D_{h,xx} u_j^n + \ldots), \tag{23}$$

$$\text{applyBoundaryConditions}(u^{n+1}). \tag{24}$$

The corrector adds the upwind dissipation to u_j^{n+1}, always using the latest value in the right-hand-side,

$$u_j^{n+1} \leftarrow u_j^{n+1} - \nu_p \, \lambda \, (-\Delta_+ \Delta_-)^q \Big(\frac{u_j^{n+1} - u_j^{n-1}}{2} \Big), \tag{25}$$

$$\text{applyBoundaryConditions}(u^{n+1}), \tag{26}$$

This version is stable for $p = 2$ with $\nu_2 = 1/8$ (found in practice, need to do the analysis). This version has the advantage of not needed storage to hold $(u^p - u^{n-1})/2$.

Note: The actual upwind scheme implemented in CgWave (and CgMx) uses a slightly modified algorithm that avoids one application of the boundary conditions (applying the BCs and interface conditions in CgMx can sometimes be expensive) . Instead of adding the dissipation at the end of the step to u^{n+1}, we add it at the start of the step to u^n,

$$u_j^n \leftarrow u_j^n - \nu_p \, \lambda \, (-\Delta_+ \Delta_-)^q \Big(\frac{u_j^n - u_j^{n-2}}{2} \Big), \tag{27}$$

and do not apply the boundary conditions (this works since formally one application of the dissipation adds a small $\mathcal{O}(h^{p+2})$ correction to u_j^n so the BCs will still be satisfied to the expected order of accuracy). This is followed by the usual update

$$u_j^{n+1} = 2u_j^2 - u_j^{n-1} + \Delta t^2 \Big(c^2 D_{h,xx} u_j^n + \ldots), \tag{28}$$

$$\text{applyBoundaryConditions}(u^{n+1}). \tag{29}$$

5. Numerical results

Here are some numerical results.

5.1. Plane Wave

Here are errors in computing an exact plane wave solution

$$u = \sin(2\pi(k_x y + k_y y + k_z z) - \omega t)$$

with $\omega/k = c$ and $k = |\mathbf{k}|$.

Square - Plane Wave Order 2

grid	N	u	r
square8	1	4.6e-2	
square16	2	1.3e-2	3.66
square32	4	3.0e-3	4.25
square64	8	7.4e-4	4.01
square128	16	1.9e-4	3.94
rate		2.00	

Table 1: CgWave, planeWave, max norm, order=2, $t = .5$, cfl=0.9, diss=0, kx=1, ky=0, kz=0, Sun Mar 1 11:12:19 2020

Square -Plane Wave Order 4

grid	N	u	r
square8	1	1.3e-2	
square16	2	5.1e-4	26.26
square32	4	2.9e-5	17.91
square64	8	1.4e-6	20.35
square128	16	7.3e-8	19.24
square256	32	4.0e-9	18.42
square512	64	2.3e-10	17.30
square1024	128	1.4e-11	16.55
rate		4.25	

Table 2: CgWave, planeWave, max norm, order=4, $t = .5$, cfl=0.9, diss=0, kx=1, ky=0, kz=0, Sun Mar 1 11:15:18 2020

CIC - circle in a channel - Plane Wave Order 2

grid	N	u	r
cic2	1	1.1e-2	
cic4	2	2.6e-3	4.18
cic8	4	6.2e-4	4.20
cic16	8	1.5e-4	4.11
cic32	16	3.7e-5	4.05
rate		2.05	

Table 3: CgWave, planeWave, max norm, order=2, $t = .5$, cfl=0.9, diss=0, kx=1, ky=0, kz=0, Sun Mar 1 13:35:54 2020

CIC - circle in a channel - Plane Wave Order 4

grid	N	u	r
cic2	1	2.7e-4	
cic4	2	1.3e-5	20.61
cic8	4	5.9e-7	22.32
cic16	8	3.0e-8	19.70
cic32	16	1.7e-9	17.33
rate		4.33	

Table 4: CgWave, planeWave, max norm, order=4, t = .5, cfl=0.9, diss=0, kx=1, ky=0, kz=0, Sun Mar 1 13:36:19 2020

Box - Plane Wave Order 2

grid	N	u	r
box1	1	5.8e-2	
box2	2	1.4e-2	4.26
box4	4	2.8e-3	4.82
box8	8	6.3e-4	4.49
rate		2.18	

Table 5: CgWave, planeWave, max norm, order=2, ts=explicit, orderInTime=-1, dtMax0=10000000000, t = .5, cfl=0.9, diss=0, -known=planeWave, kx=1, ky=1, kz=1, Sat Jul 17 06:23:09 2021

Box - Plane Wave Order 4

grid	N	u	r
box1	1	1.3e-2	
box2	2	4.6e-4	28.38
box4	4	1.6e-5	29.55
box8	8	6.3e-7	24.45
box16	16	2.8e-8	22.64
rate		4.71	

Table 6: CgWave, planeWave, max norm, order=4, ts=explicit, orderInTime=-1, dtMax0=10000000000, t = .5, cfl=0.9, diss=0, -known=planeWave, kx=1, ky=1, kz=1, Sat Jul 17 06:09:54 2021

5.2. Time-periodic solution in a box (showing forcing)

An exact solution to the forced wave equation in a rectangular box $[0,1]^d$ is given by

$$u_e(\mathbf{x},t) \stackrel{\text{def}}{=} \sin(k_x x)\,\sin(k_y y)\,[\sin(k_z z)]\,\cos(\omega t),$$

where the forcing function is

$$f(\mathbf{x},t) = \Big(-\omega^2 + c^2(k_x^2+k_y^2)\Big)u_e(\mathbf{x},t)$$

Square - Order 2

grid	N	u	r
square8	1	5.6e-2	
square16	2	2.1e-3	27.37
square32	4	1.9e-4	10.79
square64	8	1.0e-5	18.45
square128	16	5.6e-7	18.26
square256	32	4.0e-8	14.17
square512	64	2.7e-9	14.96
rate		4.02	

Table 7: CgWave, helmholtz, max norm, order=4, t = .5, cfl=0.9, diss=0, kx=2, ky=2, kz=0, Sun Mar 1 13:18:51 2020

Square - Order 4

grid	N	u	r
square8	1	5.6e-2	
square16	2	2.1e-3	27.37
square32	4	1.9e-4	10.79
square64	8	1.0e-5	18.45
square128	16	5.6e-7	18.26
square256	32	4.0e-8	14.17
square512	64	2.7e-9	14.96
rate		4.02	

Table 8: CgWave, helmholtz, max norm, order=4, t = .5, cfl=0.9, diss=0, kx=2, ky=2, kz=0, Sun Mar 1 13:18:51 2020

CIC - circle in a channel - Order 2

grid	N	u	r
cic2	1	1.7e-2	
cic4	2	2.4e-3	6.86
cic8	4	5.8e-4	4.17
cic16	8	1.4e-4	4.03
cic32	16	3.6e-5	4.01
rate		2.18	

Table 9: CgWave, helmholtz, max norm, order=2, t = .5, cfl=0.9, diss=0, kx=1, ky=1, kz=0, Sun Mar 1 13:42:02 2020

CIC - circle in a channel - Order 4

grid	N	u	r
cic2	1	3.3e-4	
cic4	2	1.9e-5	17.74
cic8	4	7.1e-7	26.27
cic16	8	2.3e-8	30.26
cic32	16	5.9e-10	39.80
rate		4.78	

Table 10: CgWave, helmholtz, max norm, order=4, t = .5, cfl=0.9, diss=0, kx=1, ky=1, kz=0, Sun Mar 1 13:41:12 2020

Box - Order 2

grid	N	u	r
box1	1	2.8e-3	
box2	2	6.0e-4	4.76
box4	4	1.4e-4	4.29
box8	8	3.4e-5	4.06
rate		2.12	

Table 11: CgWave, helmholtz, max norm, order=2, ts=explicit, orderInTime=-1, dtMax0=10000000000, t = .7, cfl=0.9, diss=0, , kx=1, ky=1, kz=1, Sat Jul 17 06:49:00 2021

Box - Order 4

grid	N	u	r
box1	1	1.5e-3	
box2	2	1.2e-4	12.26
box4	4	6.4e-6	19.44
box8	8	3.3e-7	19.45
box16	16	1.4e-8	23.10
rate		4.20	

Table 12: CgWave, helmholtz, max norm, order=4, ts=explicit, orderInTime=-1, dtMax0=10000000000, t = .7, cfl=0.9, diss=0, , kx=1, ky=1, kz=1, Sat Jul 17 06:37:57 2021

Non-Box - Order 2

grid	N	u	r
nonBox1	1	6.0e-4	
nonBox2	2	1.4e-4	4.29
nonBox4	4	3.4e-5	4.06
nonBox8	8	8.5e-6	4.02
rate		2.04	

Table 13: CgWave, helmholtz, max norm, order=2, ts=explicit, orderInTime=-1, dtMax0=10000000000, t = .7, cfl=0.9, diss=0, , kx=1, ky=1, kz=1, Sat Jul 17 06:49:38 2021

Non-Box - Order 4

grid	N	u	r
nonBox1	1	1.2e-4	
nonBox2	2	6.4e-6	19.44
nonBox4	4	3.3e-7	19.45
nonBox8	8	1.4e-8	23.10
rate		4.36	

Table 14: CgWave, helmholtz, max norm, order=4, ts=explicit, orderInTime=-1, dtMax0=10000000000, $t = .7$, cfl=0.9, diss=0, , kx=1, ky=1, kz=1, Sat Jul 17 08:19:54 2021

5.3. Gaussian Plane Wave

Figure 1 shows a modulated Gaussian plane wave hitting some shapes, scheme FD44s.

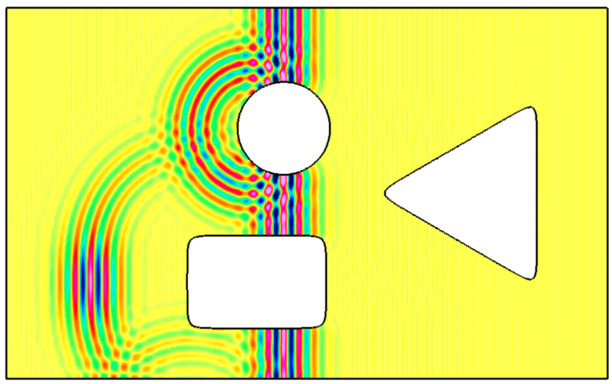

Figure 1: Modulated Gaussian plane wave hitting some shapes, FD44s.

References

[1] W. D. Henshaw, D. W. Schwendeman, Parallel computation of three-dimensional flows using overlapping grids with adaptive mesh refinement, J. Comput. Phys. 227 (16) (2008) 7469–7502.

[2] W. D. Henshaw, A high-order accurate parallel solver for Maxwell's equations on overlapping grids, SIAM J. Sci. Comput. 28 (5) (2006) 1730–1765.

[3] J. W. Banks, W. D. Henshaw, Upwind schemes for the wave equation in second-order form, J. Comput. Phys. 231 (17) (2012) 5854–5889.

[4] J. Angel, J. W. Banks, W. D. Henshaw, High-order upwind schemes for the wave equation on overlapping grids: Maxwell's equations in second-order form, J. Comput. Phys. 352 (2018) 534–567.

[5] J. Angel, J. W. Banks, W. D. Henshaw, Efficient high-order upwind difference schemes for the second-order wave equation on overlapping grids, Tech. rep., in preparation (2018).