Cgmp: A Multi-Physics Multi-Domain Solver for FSI and CHT
User Guide and Reference Manual

William D. Henshaw,
Department of Mathematical Sciences,
Rensselaer Polytechnic Institute,
Troy, NY, USA, 12180.

June 26, 2021

Abstract:
This document describes **Cgmp**, a solver written using the **Overture** framework to solve multi-physics multi-domain problems. The solver can be used, for example, to solve conjugate heat transfer (CHT) problems where fluid flow in one domain is coupled to heat transfer in another *solid* domain. CgMp can also be used to solve fluid-structure interaction (FSI) problems such as deforming solids in an incompressible fluid.

Contents

List of Figures

1 Introduction

This document is currently under development.

Cgmp solves problems on overset (overlapping/composite/Chimera) grids and is built upon the **Overture** framework [1],[3],[2]. A block diagram of the main components of Overture is given in Figure 1. Overture provides support for solving PDEs on overset grids. CgMp is primarily a driver program that calls different CG (Composite Grid) solvers: (see Figure 2),

CgAd: (Advection-Diffusion) solves parabolic equations for heat transfer in solids.

CgCns: solves compressible flow (reacting, multi-fluid, multi-phase).

Cgins: solves for incompressible flow and heat transfer.

CgSm: solves the equations of linear and nonlinear elasticity for the deformation solids.

CgMp also coordinates the transfer of information between different domain solvers (e.g. the temperature and fluxes at the interface between two heat conducting materials.) CgMp can join any number of domain solvers. There can be multiple instances of CgAd/CgCns/CgIns/CgSm solvers, for example, for different types of fluid and solids.

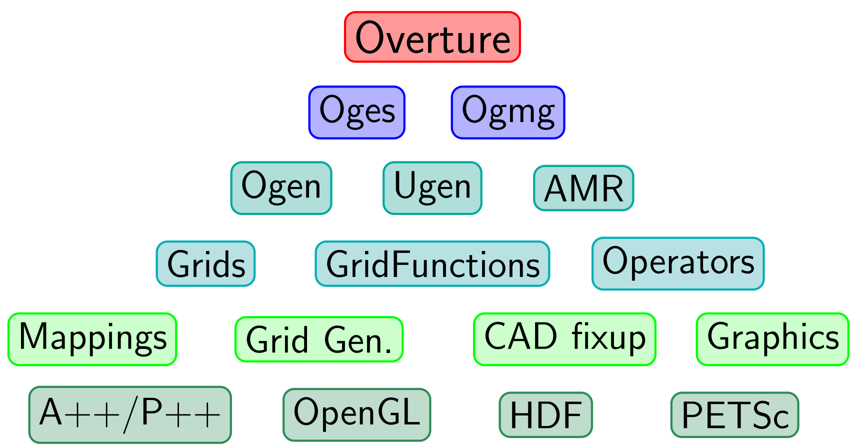

Figure 1: Overture block diagram. Overture supports the solution of PDEs on overset grids. It has classes to hold grids and grid functions. It has support for component grid generation and overset grid generation. Oges is the interface to sparse solvers such as those in PETSc. A++/P++ is the serial/parallel array class for C++.

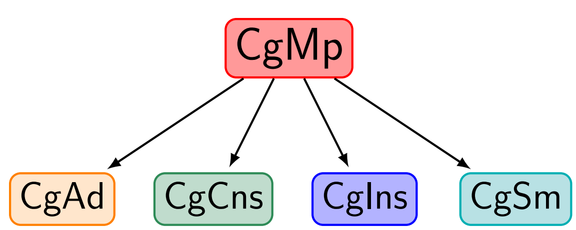

Figure 2: Cgmp drives different CG solvers to perform CHT and FSI simulations. CgAd solves for heat transfer in solids. CgCns solves compressible flow. Cgins solves incompressible flow. CgSm solves for deformations of solids, linear and nonlinear elasticity.

2 The Equations

3 Time-stepping

4 Results

5 Convergence results

This section details the results of various convergence tests. Convergence results are run using the **twilight-zone** option, also known less formally as the **method of analytic solutions**. In this case the equations are forced so the the solution will be a known analytic function.

The tables show the maximum errors in the solution components. The rate shown is estimated convergence rate, σ, assuming error $\propto h^\sigma$. The rate is estimated by a least squares fit to the data.

The 2D trigonometric solution used as a twilight zone function is

$$
\begin{aligned}
u &= \frac{1}{2}\cos(\pi\omega_0 x)\cos(\pi\omega_1 y)\cos(\omega_3\pi t) + \frac{1}{2} \\
v &= \frac{1}{2}\sin(\pi\omega_0 x)\sin(\pi\omega_1 y)\cos(\omega_3\pi t) + \frac{1}{2} \\
p &= \cos(\pi\omega_0 x)\cos(\pi\omega_1 y)\cos(\omega_3\pi t) + \frac{1}{2}
\end{aligned}
$$

The 3D trigonometric solution is

$$
\begin{aligned}
u &= \cos(\pi\omega_0 x)\cos(\pi\omega_1 y)\cos(\pi\omega_2 z)\cos(\omega_3\pi t) \\
v &= \frac{1}{2}\sin(\pi\omega_0 x)\sin(\pi\omega_1 y)\cos(\pi\omega_2 z)\cos(\omega_3\pi t) \\
w &= \frac{1}{2}\sin(\pi\omega_0 x)\sin(\pi\omega_1 y)\sin(\pi\omega_2 z)\cos(\omega_3\pi t) \\
p &= \frac{1}{2}\sin(\pi\omega_0 x)\cos(\pi\omega_1 y)\cos(\pi\omega_2 z)\sin(\omega_3\pi t)
\end{aligned}
$$

When $\omega_0 = \omega_1 = \omega_2$ it follows that $\nabla\cdot\mathbf{u} = 0$. There are also algebraic polynomial solutions of different orders.

grid	N	p	u	v	T	$\nabla\cdot\mathbf{u}$	T_s
innerOuter1	20	1.32e-02	3.69e-03	2.74e-03	8.37e-04	2.68e-02	1.75e-03
innerOuter2	40	2.17e-03	4.03e-04	3.26e-04	1.85e-04	5.01e-03	4.33e-04
innerOuter4	80	4.20e-04	5.31e-05	3.89e-05	4.55e-05	1.71e-03	1.09e-04

Table 1: Inner-outer. Maximum errors, polynomial, $t = 1.$, $\nu = .1$, $k_T = \nu/P_r$, $k_s = .01$ (?).

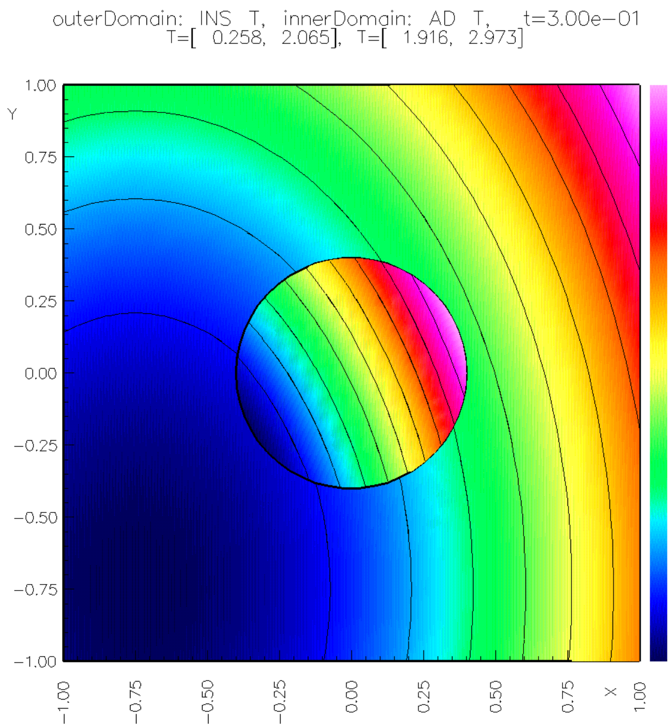

Figure 3: INS outside, AD inside, TZ.

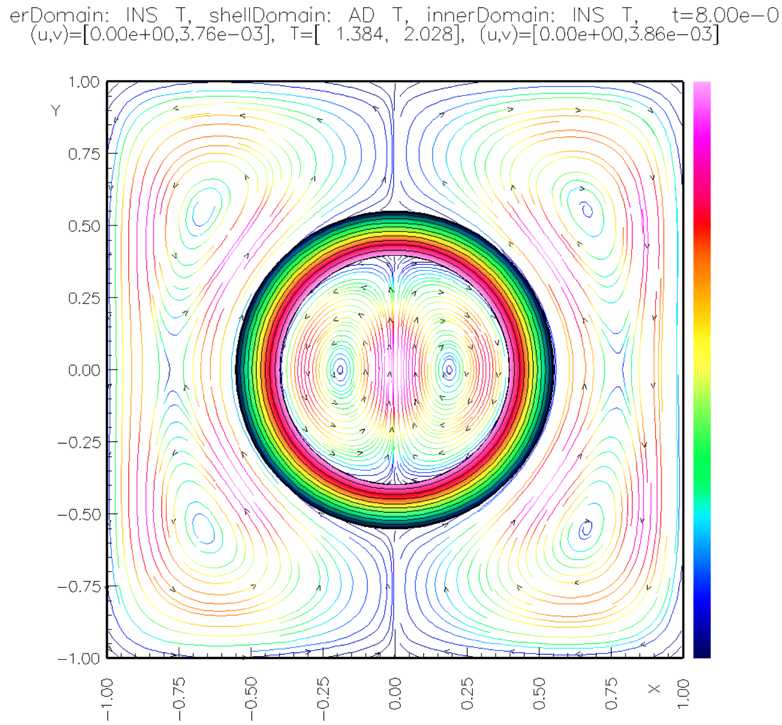

Figure 4: INS outside, AD in shell, INS inside.

6 Some interesting examples

Here is a collection of interesting examples computed with Cgmp.

6.1 A Hot cylindrical shell separating two incompressible fluids

Figure 4 shows a solid "hot" shell separating two incompressible (Boussinesq) fluids

References

[1] D. L. Brown, G. S. Chesshire, W. D. Henshaw, and D. J. Quinlan, *Overture: An object oriented software system for solving partial differential equations in serial and parallel environments*, in Proceedings of the Eighth SIAM Conference on Parallel Processing for Scientific Computing, 1997, p. .

[2] D. L. Brown, W. D. Henshaw, and D. J. Quinlan, *Overture: An object oriented framework for solving partial differential equations*, in Scientific Computing in Object-Oriented Parallel Environments, Springer Lecture Notes in Computer Science, 1343, 1997, pp. 177–194.

[3] W. D. Henshaw, *Overture: An object-oriented system for solving PDEs in moving geometries on overlapping grids*, in First AFOSR Conference on Dynamic Motion CFD, June 1996, L. Sakell and D. D. Knight, eds., 1996, pp. 281–290.

Index