An Introduction to Overset Grids

Bill Henshaw

Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
Livermore, CA

2nd Bay Area Overset Network Meeting,
Stanford California, 2011

Top 3 reasons for using overset grids.

1. Complex geometry and accuracy: You need to solve a partial differential equation (PDE) on a complex geometry and require accurate representations at boundaries (e.g. boundary layers).
2. Moving geometry : overset grids provide fast moving grid generation and high quality grids.
3. Efficiency: overset grids can take advantage of fast and memory efficient algorithms for structured (and Cartesian) grids.
   - Example: 3D, 4th-order Maxwell : Cartesian grids are 25$\times$ faster than curvilinear grids which themselves are $2-10??\times$ faster than unstructured grids.
   - Example: multigrid solvers for overset grids: can be an order of magnitude faster (e.g. 50$\times$) and more memory efficient (e.g. 10$\times$) than the best Krylov based solvers.

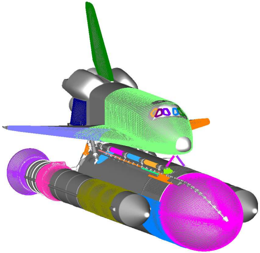
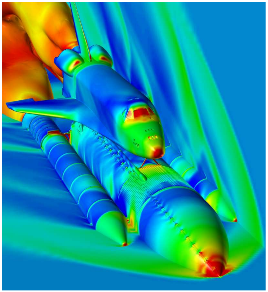
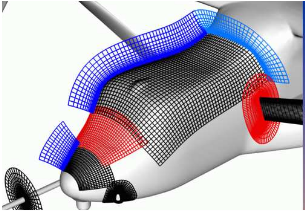
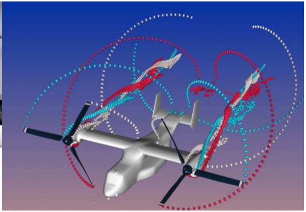
Top 3 reasons for using overset grids.

1. Complex geometry and accuracy: You need to solve a partial differential equation (PDE) on a complex geometry and require accurate representations at boundaries (e.g. boundary layers).
2. Moving geometry : overset grids provide fast moving grid generation and high quality grids.
3. Efficiency: overset grids can take advantage of fast and memory efficient algorithms for structured (and Cartesian) grids.
   - Example: 3D, 4th-order Maxwell : Cartesian grids are $25\times$ faster than curvilinear grids which themselves are $2-10??\times$ faster than unstructured grids.
   - Example: multigrid solvers for overset grids: can be an order of magnitude faster (e.g. $50\times$) and more memory efficient (e.g. $10\times$) than the best Krylov based solvers.

Top 3 reasons for using overset grids.

1. Complex geometry and accuracy: You need to solve a partial differential equation (PDE) on a complex geometry and require accurate representations at boundaries (e.g. boundary layers).
2. Moving geometry : overset grids provide fast moving grid generation and high quality grids.
3. Efficiency: overset grids can take advantage of fast and memory efficient algorithms for structured (and Cartesian) grids.
   - Example: 3D, 4th-order Maxwell : Cartesian grids are $25\times$ faster than curvilinear grids which themselves are $2-10??\times$ faster than unstructured grids.
   - Example: multigrid solvers for overset grids: can be an order of magnitude faster (e.g. $50\times$) and more memory efficient (e.g. $10\times$) than the best Krylov based solvers.

Overset grids are used to solve some of the most difficult CFD problems in aerospace.

Space shuttle figures courtesy of William Chan and Reynaldo Gomez.
V-22 Osprey figures courtesy of William Chan, Andrew Wissink and Robert Meakin.

Composite/ Chimeral/ Overset/ Overlapping Grids

A Short History

- Volkov, circa [1966] developed a *Composite Mesh* method for Laplace's equation on regions with piece-wise smooth boundaries separated by corners. Polar grids are fitted at corners to handle potential singularities.
- Starius, circa [1977] (student of H.-O. Kreiss) considered *Composite Mesh* methods for elliptic and hyperbolic problems – introduces a hyperbolic grid generator.
- Steger, circa [1980] independently conceives the idea of the overlapping grid, subsequently named the *Chimera* approach after the mythical Chimera beast having a human face, a lion's mane and legs, a goat's body, and dragon's tail. NASA groups develop grid generator PEGSUS, hyperbolic grid generation and flow solver Overflow (Steger, Benek, Suhs, Buning, Chan, Meakin, et. al.)
- B. Kreiss circa [1980] develops overlapping grid generator which subsequently leads to the CMPGRD grid generator [1983] (Chesshire, Henshaw) later leading to the Overture set of tools [1994].

Composite/ Chimera/ Overset/ Overlapping Grids
A Short History

- Volkov, circa [1966] developed a *Composite Mesh* method for Laplace's equation on regions with piece-wise smooth boundaries separated by corners. Polar grids are fitted at corners to handle potential singularities.
- Starius, circa [1977] (student of H.-O. Kreiss) considered *Composite Mesh* methods for elliptic and hyperbolic problems – introduces a hyperbolic grid generator.
- Steger, circa [1980] independently conceives the idea of the overlapping grid, subsequently named the *Chimera* approach after the mythical Chimera beast having a human face, a lion's mane and legs, a goat's body, and dragon's tail. NASA groups develop grid generator PEGSUS, hyperbolic grid generation and flow solver Overflow (Steger, Benek, Suhs, Buning, Chan, Meakin, et. al.)
- B. Kreiss circa [1980] develops overlapping grid generator which subsequently leads to the CMPGRD grid generator [1983] (Chesshire, Henshaw) later leading to the Overture set of tools [1994].

Composite/ Chimera/ Overset/ Overlapping Grids

A Short History

- Volkov, circa [1966] developed a *Composite Mesh* method for Laplace's equation on regions with piece-wise smooth boundaries separated by corners. Polar grids are fitted at corners to handle potential singularities.
- Starius, circa [1977] (student of H.-O. Kreiss) considered *Composite Mesh* methods for elliptic and hyperbolic problems – introduces a hyperbolic grid generator.
- Steger, circa [1980] independently conceives the idea of the overlapping grid, subsequently named the *Chimera* approach after the mythical Chimera beast having a human face, a lion's mane and legs, a goat's body, and dragon's tail. NASA groups develop grid generator PEGSUS, hyperbolic grid generation and flow solver Overflow (Steger, Benek, Suhs, Buning, Chan, Meakin, et. al.)
- B. Kreiss circa [1980] develops overlapping grid generator which subsequently leads to the CMPGRD grid generator [1983] (Chesshire, Henshaw) later leading to the Overture set of tools [1994].

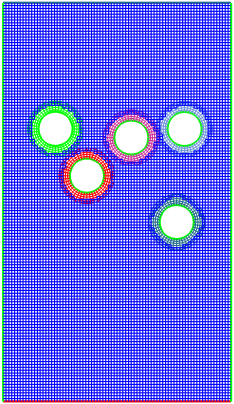
Composite/ Chimera/ Overset/ Overlapping Grids

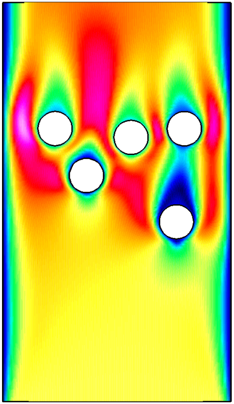
A Short History

- Volkov, circa [1966] developed a *Composite Mesh* method for Laplace's equation on regions with piece-wise smooth boundaries separated by corners. Polar grids are fitted at corners to handle potential singularities.
- Starius, circa [1977] (student of H.-O. Kreiss) considered *Composite Mesh* methods for elliptic and hyperbolic problems – introduces a hyperbolic grid generator.
- Steger, circa [1980] independently conceives the idea of the overlapping grid, subsequently named the *Chimera* approach after the mythical Chimera beast having a human face, a lion's mane and legs, a goat's body, and dragon's tail. NASA groups develop grid generator PEGSUS, hyperbolic grid generation and flow solver Overflow (Steger, Benek, Suhs, Buning, Chan, Meakin, et. al.)
- B. Kreiss circa [1980] develops overlapping grid generator which subsequently leads to the CMPGRD grid generator [1983] (Chesshire, Henshaw) later leading to the Overture set of tools [1994].

What are overlapping grids and why are they useful?

Overlapping grid: a set of structured grids that overlap.

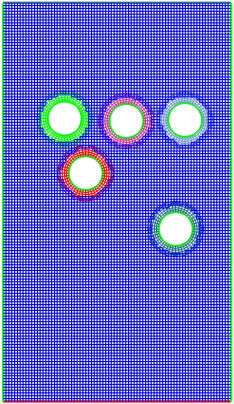


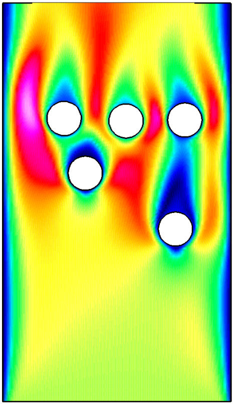


- Overlapping grids can be rapidly generated as bodies move.
- High quality grids under large displacements.
- Cartesian grids for efficiency.
- Efficient for high-order accurate methods.

What are overlapping grids and why are they useful?

Overlapping grid: a set of structured grids that overlap.

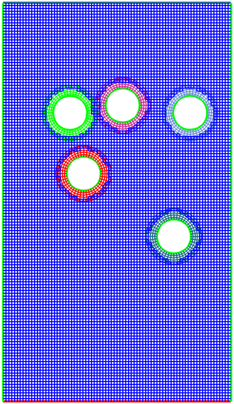


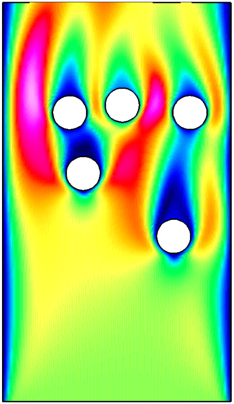


- Overlapping grids can be rapidly generated as bodies move.
- High quality grids under large displacements.
- Cartesian grids for efficiency.
- Efficient for high-order accurate methods.

What are overlapping grids and why are they useful?

Overlapping grid: a set of structured grids that overlap.

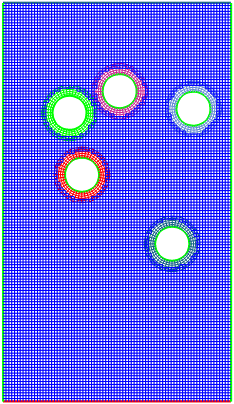


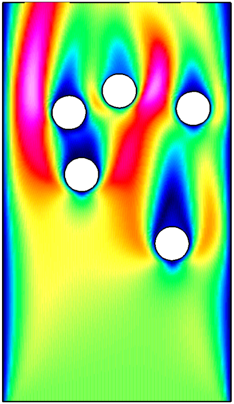


- Overlapping grids can be rapidly generated as bodies move.
- High quality grids under large displacements.
- Cartesian grids for efficiency.
- Efficient for high-order accurate methods.

What are overlapping grids and why are they useful?

Overlapping grid: a set of structured grids that overlap.

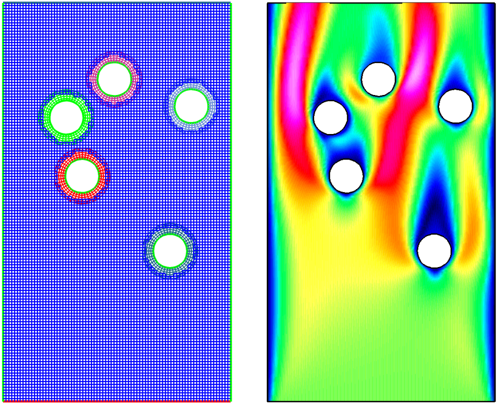




- Overlapping grids can be rapidly generated as bodies move.
- High quality grids under large displacements.
- Cartesian grids for efficiency.
- Efficient for high-order accurate methods.

What are overlapping grids and why are they useful?

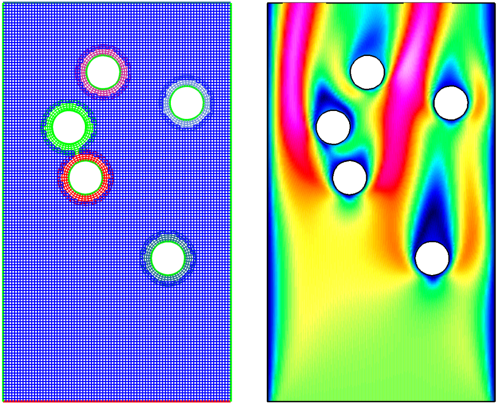
Overlapping grid: a set of structured grids that overlap.



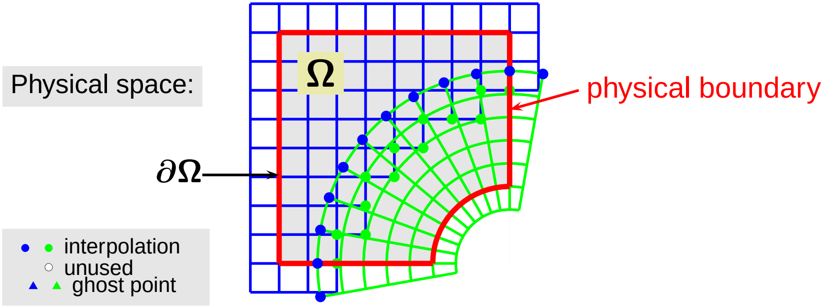
- Overlapping grids can be rapidly generated as bodies move.
- High quality grids under large displacements.
- Cartesian grids for efficiency.
- Efficient for high-order accurate methods.

What are overlapping grids and why are they useful?

Overlapping grid: a set of structured grids that overlap.

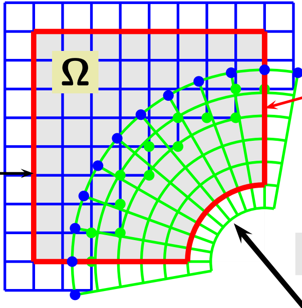


- Overlapping grids can be rapidly generated as bodies move.
- High quality grids under large displacements.
- Cartesian grids for efficiency.
- Efficient for high-order accurate methods.


Physical space:
Ω
∂Ω
physical boundary
interpolation
unused
ghost point

Components of an Overlapping Grid

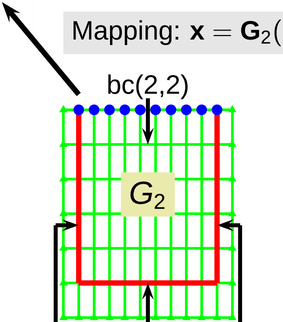
Physical space:



interpolation
unused
ghost point

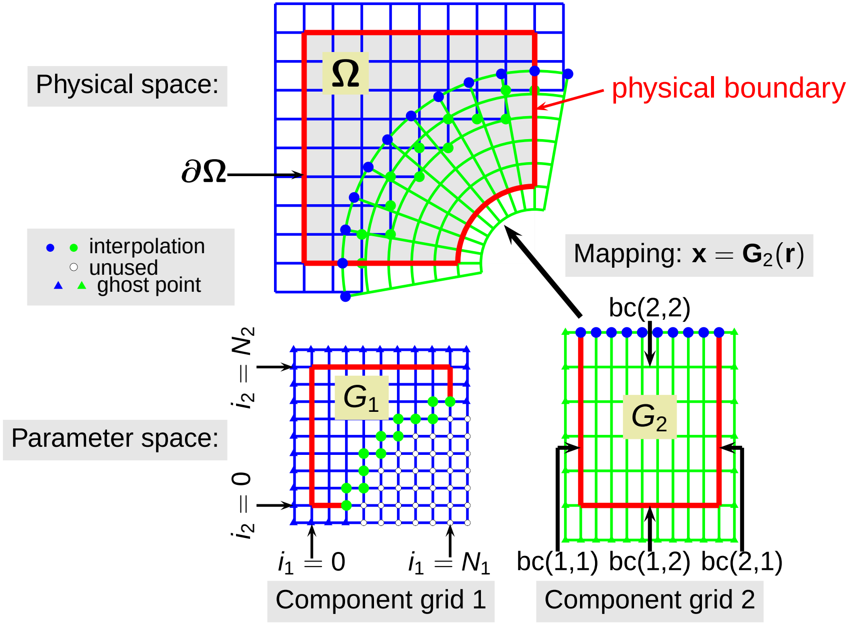
Mapping: $\mathbf{x} = \mathbf{G}_2(\mathbf{r})$

Parameter space:

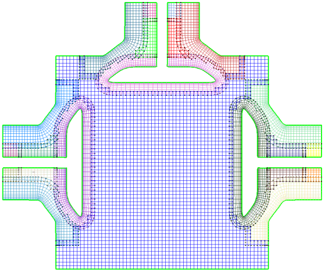


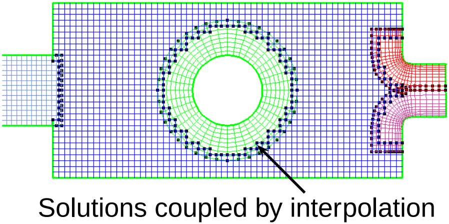
Component grid 2

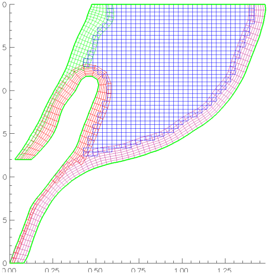
Components of an Overlapping Grid



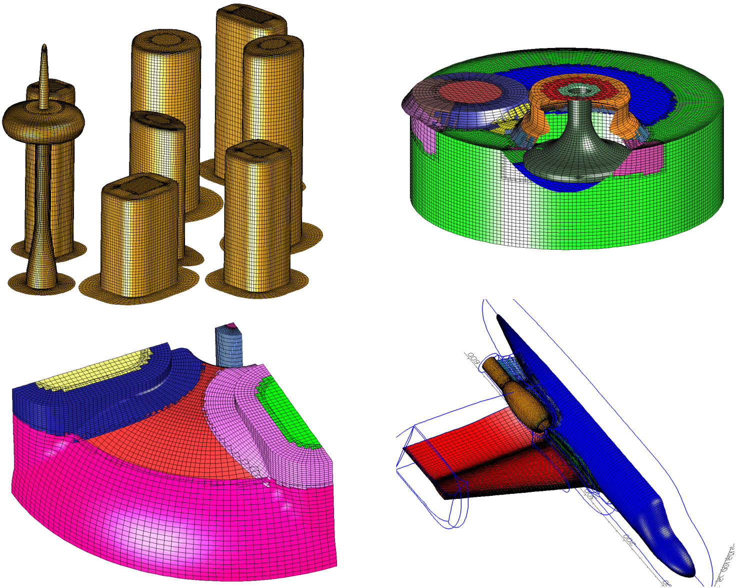
Sample 2D overlapping grids (built with Ogen)







Sample 3D overlapping grids (Ogen)



A one-dimensional overlapping grid example

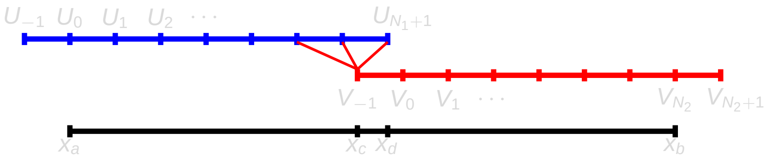
To solve the advection-diffusion equation

$$
\begin{aligned}
u_t + a u_x &= \nu u_{xx}, && x \in (0,1) \\
u(0,t) = g_0(t), \quad u_x(1,t) &= g_1(t), && \text{(boundary conditions)} \\
u(x,0) &= u_0(x), && \text{(initial conditions)}
\end{aligned}
$$

introduce grid points on the two overlapping component grids,

$$
\begin{aligned}
x_i^{(1)} &= x_a + i\Delta x_1, & i &= -1,0,1,\ldots,N_1+1, & \Delta x_1 &= (x_d - x_a)/N_1 \\
x_j^{(2)} &= x_c + (j+1)\Delta x_2, & j &= -1,0,1,\ldots,N_2+1, & \Delta x_2 &= (x_b - x_c)/N_2
\end{aligned}
$$

and approximations $U_i^n \approx u(x_i^{(1)}, n\Delta t)$, $V_i^n \approx u(x_i^{(2)}, n\Delta t)$.



A one-dimensional overlapping grid example

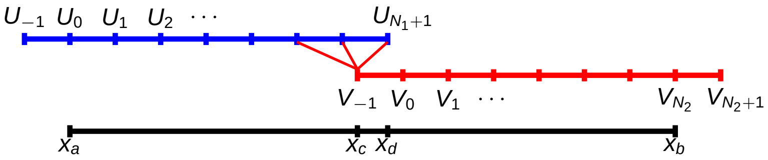
To solve the advection-diffusion equation

$$\begin{aligned} u_t + a u_x &= \nu u_{xx}, && x \in (0,1) \\ u(0,t) = g_0(t), \quad u_x(1,t) &= g_1(t), && \text{(boundary conditions)} \\ u(x,0) &= u_0(x), && \text{(initial conditions)} \end{aligned}$$

introduce grid points on the two overlapping component grids,

$$\begin{aligned} x_i^{(1)} &= x_a + i\Delta x_1, & i &= -1,0,1,\ldots,N_1+1, & \Delta x_1 &= (x_d - x_a)/N_1 \\ x_j^{(2)} &= x_c + (j+1)\Delta x_2, & j &= -1,0,1,\ldots,N_2+1, & \Delta x_2 &= (x_b - x_c)/N_2 \end{aligned}$$

and approximations $U_i^n \approx u(x_i^{(1)}, n\Delta t)$, $V_i^n \approx u(x_i^{(2)}, n\Delta t)$.



Given the solution at time t^n, compute the solution at time t^{n+1}:

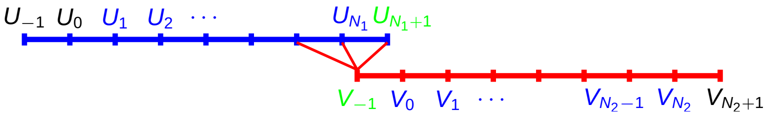
$$(U_i^{n+1} - U_i^n)/\Delta t = -a\frac{U_{i+1}^n - U_{i-1}^n}{2\Delta x} + \nu\frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{\Delta x^2}, \qquad i = 1, 2, \dots, N_1$$

$$(V_j^{n+1} - V_j^n)/\Delta t = -a\frac{V_{i+1}^n - V_{i-1}^n}{2\Delta x} + \nu\frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{\Delta x^2}, \qquad j = 0, 2, \dots, N_2$$

$$U_0^{n+1} = g(t^n), \quad D_0 V_{N_2}^{n+1} = g_1(t^{n+1}), \qquad \text{(boundary conditions)}$$

$$U_{N_1+1}^{n+1} = (1-\alpha)(1-\frac{\alpha}{2})\, V_{-1}^{n+1} + \alpha(2-\alpha)\, V_0^{n+1} + \frac{\alpha}{2}(\alpha-1)\, V_1^{n+1}, \quad \text{(interpolation)}$$

$$V_{-1}^{n+1} = (1-\beta)(1-\frac{\beta}{2})\, U_{N_1-1}^{n+1} + \beta(2-\beta)\, U_{N_1}^{n+1} + \frac{\beta}{2}(\beta-1)\, U_{N_1+1}^{n+1}, \quad \text{(interpolation)}$$



How to advance the solution on an overlapping grid.

(1) interior equations, (2) boundary conditions, (3) interpolation points.

Given the solution at time t^n, compute the solution at time t^{n+1}:

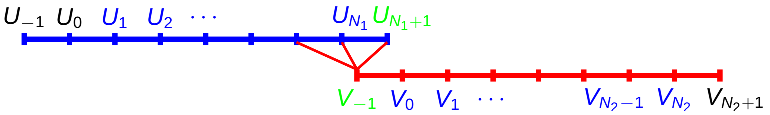
$$(U_i^{n+1} - U_i^n)/\Delta t = -a\frac{U_{i+1}^n - U_{i-1}^n}{2\Delta x} + \nu\frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{\Delta x^2}, \qquad i = 1, 2, \dots, N_1$$

$$(V_j^{n+1} - V_j^n)/\Delta t = -a\frac{V_{i+1}^n - V_{i-1}^n}{2\Delta x} + \nu\frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{\Delta x^2}, \qquad j = 0, 2, \dots, N_2$$

$$U_0^{n+1} = g(t^n), \quad D_0 V_{N_2}^{n+1} = g_1(t^{n+1}), \qquad \text{(boundary conditions)}$$

$$U_{N_1+1}^{n+1} = (1-\alpha)(1-\frac{\alpha}{2})\,V_{-1}^{n+1} + \alpha(2-\alpha)\,V_0^{n+1} + \frac{\alpha}{2}(\alpha-1)\,V_1^{n+1}, \quad \text{(interpolation)}$$

$$V_{-1}^{n+1} = (1-\beta)(1-\frac{\beta}{2})\,U_{N_1-1}^{n+1} + \beta(2-\beta)\,U_{N_1}^{n+1} + \frac{\beta}{2}(\beta-1)\,U_{N_1+1}^{n+1}, \quad \text{(interpolation)}$$



How to advance the solution on an overlapping grid.

(1) interior equations, (2) boundary conditions, (3) interpolation points.

Given the solution at time t^n, compute the solution at time t^{n+1}:

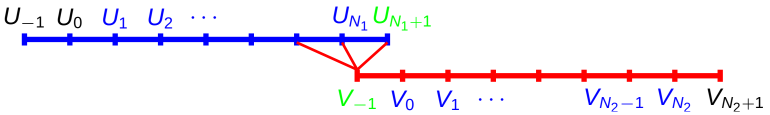
$$(U_i^{n+1} - U_i^n)/\Delta t = -a\frac{U_{i+1}^n - U_{i-1}^n}{2\Delta x} + \nu\frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{\Delta x^2}, \qquad i = 1, 2, \ldots, N_1$$

$$(V_j^{n+1} - V_j^n)/\Delta t = -a\frac{V_{i+1}^n - V_{i-1}^n}{2\Delta x} + \nu\frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{\Delta x^2}, \qquad j = 0, 2, \ldots, N_2$$

$$U_0^{n+1} = g(t^n), \quad D_0 V_{N_2}^{n+1} = g_1(t^{n+1}), \qquad \text{(boundary conditions)}$$

$$U_{N_1+1}^{n+1} = (1-\alpha)(1-\frac{\alpha}{2})\,V_{-1}^{n+1} + \alpha(2-\alpha)\,V_0^{n+1} + \frac{\alpha}{2}(\alpha-1)\,V_1^{n+1}, \quad \text{(interpolation)}$$

$$V_{-1}^{n+1} = (1-\beta)(1-\frac{\beta}{2})\,U_{N_1-1}^{n+1} + \beta(2-\beta)\,U_{N_1}^{n+1} + \frac{\beta}{2}(\beta-1)\,U_{N_1+1}^{n+1}, \quad \text{(interpolation)}$$



Theory for finite difference schemes

There is extensive numerical analysis theory underpinning this work.

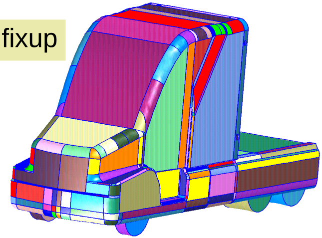
- classic von Neumann stability analysis (periodic domains).
- energy estimates (L_2-norm estimates).
- normal mode analysis, GKS theory (initial boundary value problems).

Some references:

• Gustafsson, Kreiss, Oliger, *Time Dependent Problems and Difference Methods*, (book).
• Strikwerda, *Finite Difference Schemes and Partial Differential Equations*, (book).
• Gustafsson, Kreiss, Sundström, *Stability Theory of Difference Approx. for Mixed Initial Boundary Value Problems, I. and II.*, Math. Comp.
• Starius, *On Composite Mesh Difference Methods for Hyperbolic Differential Equations*, Numer. Math.

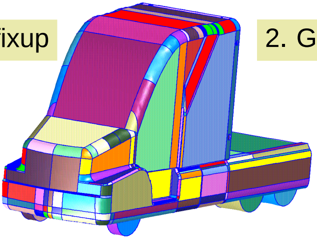
From CAD to Mesh to Solution

1. Cad fixup

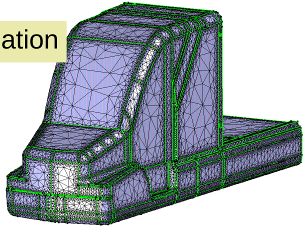


From CAD to Mesh to Solution

1. Cad fixup

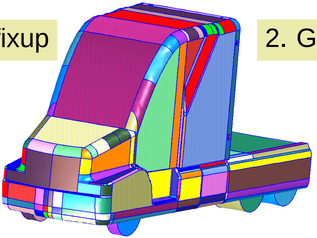


2. Global triangulation

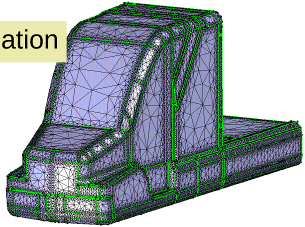


From CAD to Mesh to Solution

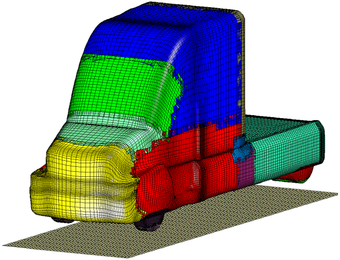
1. Cad fixup



2. Global triangulation

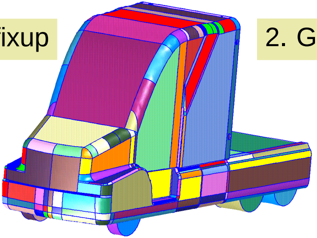


3. Overlapping grid

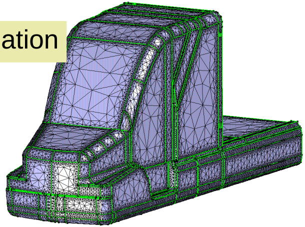


From CAD to Mesh to Solution

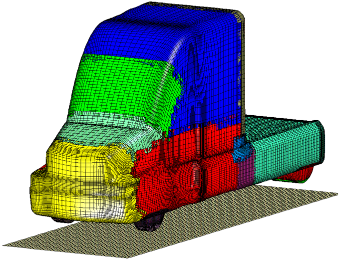
1. Cad fixup



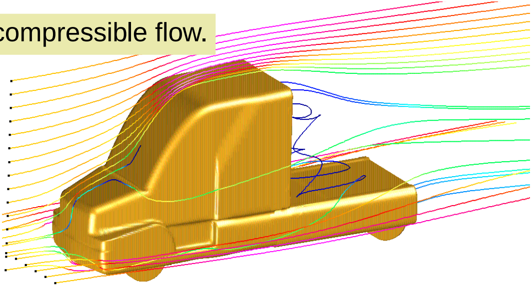
2. Global triangulation



3. Overlapping grid



4. Incompressible flow.



Software for overlapping grids.

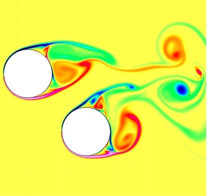
Here is a partial list of software for both grid generation and solving PDEs on overlapping grids (availability varies, e.g. some is export controlled).

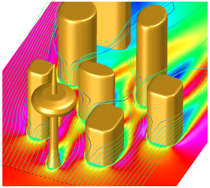
- Chimera Grid Tools : utilities, libraries and scripts for grid manipulation, component grid generation, and solution analysis.
- PEGSUS: (versions 4 and 5) : grid connectivity.
- SUGGAR, DIRTLIB : grid connectivity and interpolation utilities.
- Compressible Navier-Stokes solvers: OVERFLOW, BEGGAR, HELIOS, ADPDIS3D, SAFARI, ...
- Incompressible Navier-Stokes solvers: INS2D, INS3D, EllipSys3d, CFDShip-Iowa, ...
- Overture: grid generation and PDE solvers for fluid flows (Cgcns, Cgins), electromagnetics (Cgmx), elastic wave-equation (Cgsm), conjugate heat transfer and fluid structure interactions (Cgmp).

Applications and Movies

(taken from Overture based solvers)

Cgins: incompressible Navier-Stokes solver.



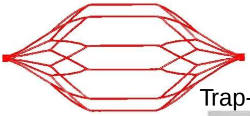


- 2nd-order and 4th-order accurate (DNS).
- support for moving rigid-bodies (not parallel yet).
- heat transfer (Boussinesq approximation).
- semi-implicit (time accurate), pseudo steady-state (efficient line solver), full implicit.

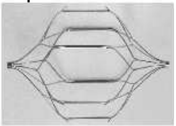
• WDH., *A Fourth-Order Accurate Method for the Incompressible Navier-Stokes Equations on Overlapping Grids*, J. Comput. Phys, **113**, no. 1, (1994) 13–25.

Flow past a blood-clot filter using cgins

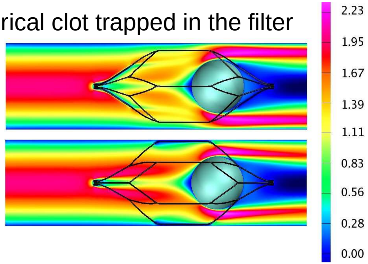
Overlapping grid for the filter



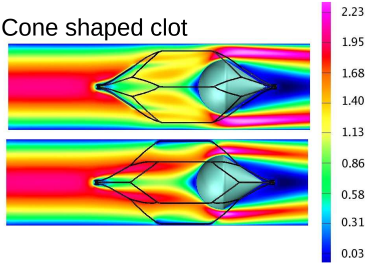
Trap-ease wire filter



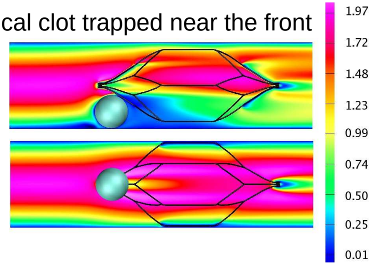
Spherical clot trapped in the filter



Cone shaped clot

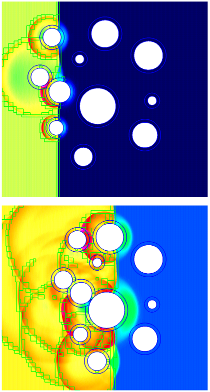


Spherical clot trapped near the front



M.A. Singer, WDH, S.L. Wang, *Computational Modeling of Blood Flow in the Trapease Inferior Vena Cava Filter*, Journal of Vascular and Interventional Radiology, **20**, 2009.

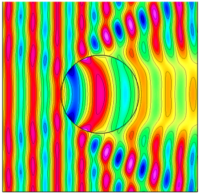
Cgcns: compressible N-S and reactive-Euler.

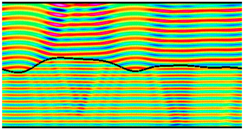


- reactive and non-reactive Euler equations, Don Schwendeman (RPI).
- compressible Navier-Stokes.
- multi-fluid formulation, Jeff Banks (LLNL).
- adaptive mesh refinement and moving grids.

- WDH., D. W. Schwendeman, *Parallel Computation of Three-Dimensional Flows using Overlapping Grids with Adaptive Mesh Refinement*, J. Comp. Phys. **227** (2008).
- WDH., DWS, *Moving Overlapping Grids with Adaptive Mesh Refinement for High-Speed Reactive and Nonreactive Flow*, J. Comp. Phys. **216** (2005).
- WDH., DWS, An adaptive numerical scheme for high-speed reactive flow on overlapping grids, J. Comp. Phys. **191** (2003).

Cgmx: electromagnetics solver.





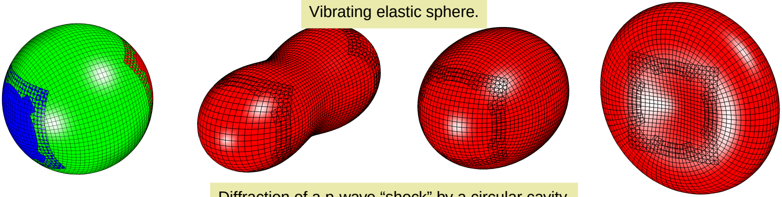
- fourth-order accurate, 2D, 3D.
- Efficient time-stepping with the modified-equation approach
- High-order accurate symmetric difference approximations.
- High-order-accurate *centered* boundary and interface conditions.

• WDH., *A High-Order Accurate Parallel Solver for Maxwell's Equations on Overlapping Grids*, SIAM J. Scientific Computing, **28**, no. 5, (2006).

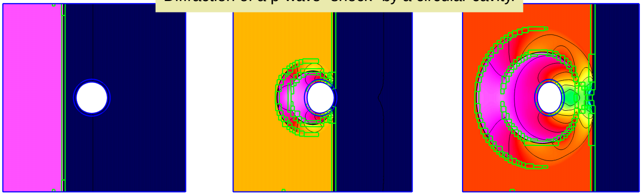
Cgsm: solve the elastic wave equation.

- linear elasticity on overlapping grids, with adaptive mesh refinement,
- conservative finite difference scheme for the second-order system,
- upwind Godunov scheme for the first-order-system.

Vibrating elastic sphere.



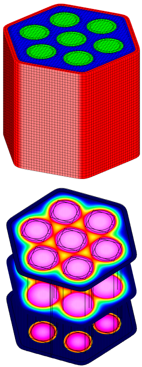
Diffraction of a p-wave “shock” by a circular cavity.



- D. Appelö, J.W. Banks , WDH, D.W. Schwendeman, *Numerical Methods for Solid Mechanics on Overlapping Grids: Linear Elasticity*, LLNL-JRNL-422223, submitted.

Cgmp: a multi-domain multi-physics solver.

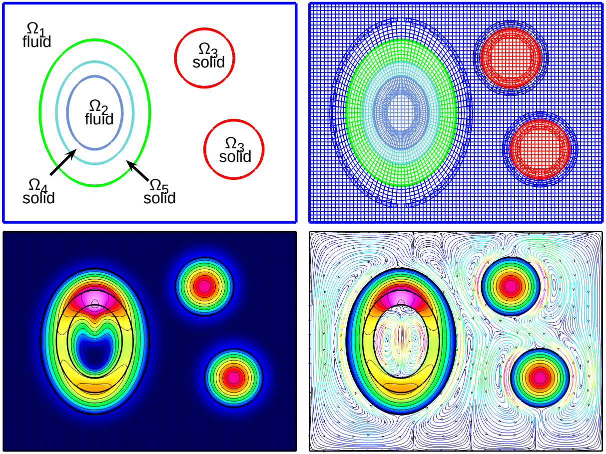
Conjugate heat transfer: coupling incompressible flow to heat conduction in solids.



- overlapping grids for each fluid or solid domain,
- a partitioned solution algorithm (separate physics solvers in each sub-domain),
- (cgins) incompressible Navier-Stokes equations (with Boussinesq approximation) for fluid domains,
- (cgad) heat equation for solid domains,
- a key issue is interface coupling.

• WDH., K. K. Chand, *A Composite Grid Solver for Conjugate Heat Transfer in Fluid-Structure Systems*, J. Comput. Phys, 2009.

The multi-domain composite grid approach



The fluid and solid sub-domains, overlapping grids and solution (temperature and streamlines) to a CHT problem. Solvers: cgins (fluid sub-domains), cgad (solid sub-domains), cgmp (coupled problem).

Summary: Overset Grids.

- Overset grids can be used to efficiently and accurately solve partial differential equations in complex geometry.
- Overset grids can be an order of magnitude faster and more memory efficient that unstructured grid algorithms.
- Overset grids are especially useful for problems with moving geometry.
- Automating the grid generation process is an important area of research.