

Nonlinear Fluid-Structure Interaction: a Partitioned Approach and its Application through Component Technology

Christophe Kassiotis

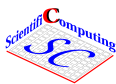
Advisors: A. Ibrahimbegović, Hermann G. Matthies and D. Duhamel

December 1, 2010 | EDF R&D, Chatou

Nonlinear Fluid-Structure Interaction: a Partitioned Approach and its Application through Component Technology

Christophe Kassiotis

November 20, 2009



Fluid Structure Interaction

- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: extreme winds or tsunami impacts on coasts

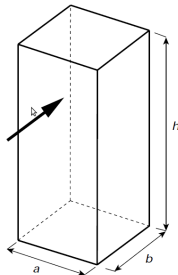
Fluid Structure Interaction

- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: **extreme winds** or tsunami impacts on coasts

Wind action (Eurocode I, P 2.4)

- Elementary geometry: A_{ref}

$$F = p_{\text{ref}} C_e C_z C_d A_{\text{ref}}$$



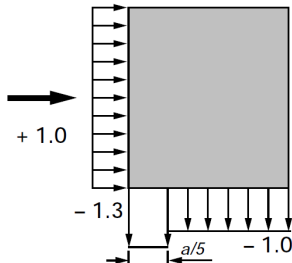
Fluid Structure Interaction

- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: **extreme winds** or tsunami impacts on coasts

Wind action (Eurocode I, P 2.4)

- Elementary geometry: A_{ref}
- Simplified Force actions:
 $p_{\text{ref}} C_e C_z$

$$F = p_{\text{ref}} C_e C_z C_d A_{\text{ref}}$$



Fluid Structure Interaction

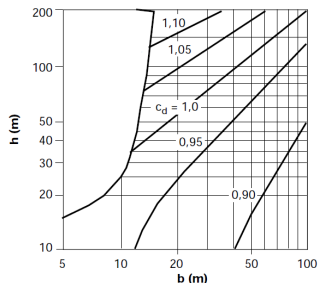
- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: **extreme winds** or tsunami impacts on coasts

Wind action (Eurocode I, P 2.4)

- Elementary geometry: A_{ref}
- Simplified Force actions:
 $p_{\text{ref}} C_e C_z$
- Simplified Interaction
wind / structure : C_d

Only the structure point of view

$$F = p_{\text{ref}} C_e C_z C_d A_{\text{ref}}$$

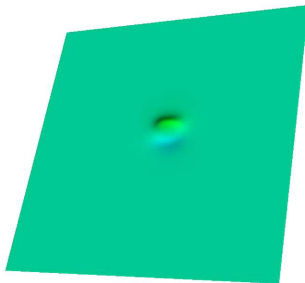


Fluid Structure Interaction

- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: extreme winds or tsunami impacts on coasts

Tsunami modeling

- Generation



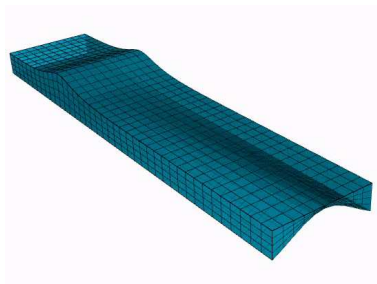
Source: CMLA-Cachan [Dutykh, 09]

Fluid Structure Interaction

- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: extreme winds or tsunami impacts on coasts

Tsunami modeling

- Generation
- Propagation



[Kassiotis, 07]

Fluid Structure Interaction

- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: extreme winds or tsunami impacts on coasts

Tsunami modeling

- Generation
- Propagation
- Run-up



Source: FBI (American Samoa office),
Samoa, September 2009

Fluid Structure Interaction

- Nearly every structure is surrounded by fluids
- Countless applications
- Among important issues: extreme winds or tsunami impacts on coasts

Tsunami modeling

- Generation
- Propagation
- Run-up

Run-up key issues

- Amplitude of the flood
- Resistance of buildings



Source: FBI (American Samoa office),
Samoa, September 2009

Goals

Coupled problem \Rightarrow Coupling approach

- Structures and fluids are two different scientific topics:
 - Different formulations: Lagrangian or Eulerian
 - Different discretization methods: FE or FV
 - Different softwares

Goals

Coupled problem \Rightarrow Coupling approach

- Structures and fluids are two different scientific topics:
 - Different formulations: Lagrangian or Eulerian
 - Different discretization methods: FE or FV
 - Different softwares
- Monolithical approach is **not a natural choice**

Monolithical approach in FSI:

- Finite Element based [Walhorn 02, Hübner *et al* 04]
- Finite Volume based [Mehl 08]

Goals

Coupled problem \Rightarrow Coupling approach

- Structures and fluids are two different scientific topics:
 - Different formulations: Lagrangian or Eulerian
 - Different discretization methods: FE or FV
 - Different softwares
- Monolithical approach is **not a natural choice**

Specifications

- Partitioned approaches
- Reach 3D computations
- Re-use dedicated and well-known codes for fluids and structures
 - Structures: *non-linear behaviors (cracking, reinforced concrete...)*
 - Fluids: *incompressibility, free surface flows, sloshing waves*

Goals

Coupled problem \Rightarrow Coupling approach

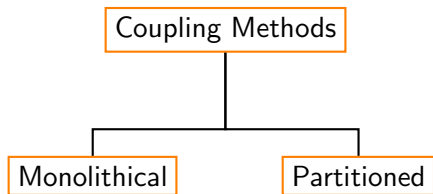
- Structures and fluids are two different scientific topics:
 - Different formulations: Lagrangian or Eulerian
 - Different discretization methods: FE or FV
 - Different softwares
- Monolithical approach is **not a natural choice**

Specifications

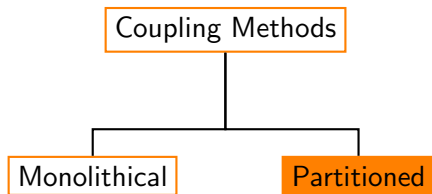
- Partitioned approaches
- Reach 3D computations
- Re-use dedicated and well-known codes for fluids and structures
 - Structures: *non-linear behaviors (cracking, reinforced concrete...)*
 - Fluids: *incompressibility, free surface flows, sloshing waves*

Software component technology

Approaches to solve FSI coupled problems



Approaches to solve FSI coupled problems

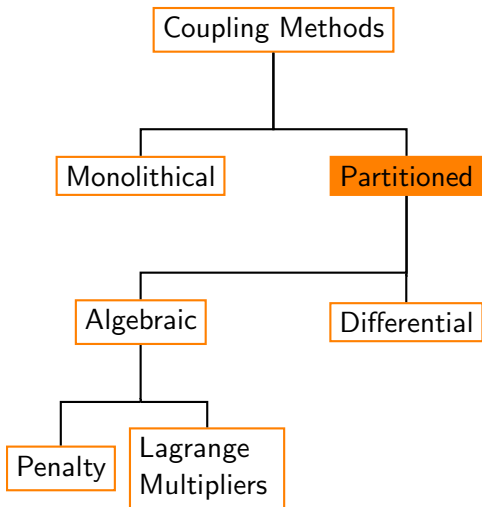


Partitioned Approach

- Introducing interface unknowns
- Advantages:
 - Independent subsystem
 - Different discretization and integration schemes
- Drawbacks
 - More unknowns
 - Stability? Convergence?

[Park & Felippa 77, Wall 99, Matthie & Steindorf 04, Vergnault 09, Gerbeau & Vidrascu 03, Fernández *et al* 07, Deparis & Quateroni, 06]

Approaches to solve FSI coupled problems

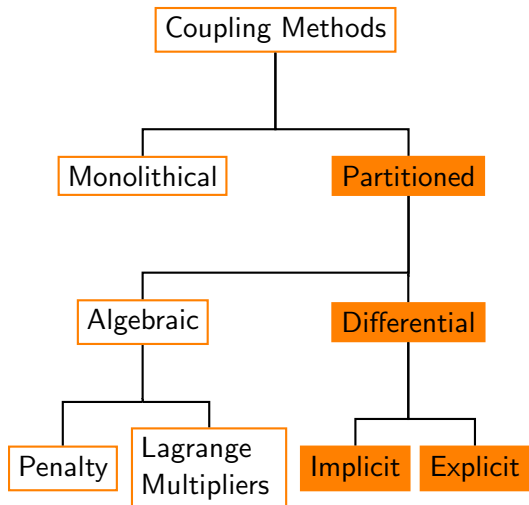


Algebraic Approach

- Minimization under an algebraic constraint (interface)
- Applied to acoustic fluids
- Advantages
 - Genericity and parallelization
 - Large coupling windows
- Drawbacks
 - Computational cost
 - Data transfer

[Park, Felippa, Ohayon, 04]

Approaches to solve FSI coupled problems



Differential approach – DFMT

- Direct Force-Motion Transfer [Ross & Felippa 09]
- Advantages
 - Simplicity
 - Data exchange
 - Few computations outside existing codes
- Drawbacks
 - Smaller coupling windows
 - Conditional stability

[Perić & Dettmer 03-07, Wall *et al* 99-09, Steindorf 04. . .]

Outline

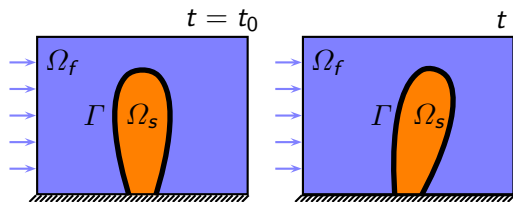
- 1 Fluid structure interaction framework
 - Structure and fluid subproblems
 - Explicit and implicit coupling algorithms for FSI
 - Convergence and stability of coupling algorithms
- 2 Software implementation and validation
 - Component architecture cops
 - Lid driven-cavity with a flexible bottom
 - Oscillating appendix in a flow
- 3 Applications: 3D computations and interaction with free surface flows
 - Three dimensional computing and paralleling
 - Solving free surface flows
 - Examples: free-surface flows impacting structures

Outline

- 1 Fluid structure interaction framework
 - Structure and fluid subproblems
 - Explicit and implicit coupling algorithms for FSI
 - Convergence and stability of coupling algorithms
- 2 Software implementation and validation
 - Component architecture cops
 - Lid driven-cavity with a flexible bottom
 - Oscillating appendix in a flow
- 3 Applications: 3D computations and interaction with free surface flows
 - Three dimensional computing and paralleling
 - Solving free surface flows
 - Examples: free-surface flows impacting structures

Structure and fluid subproblems

Continuum mechanics equations

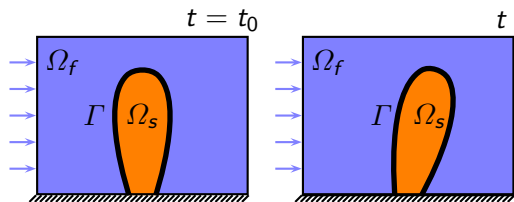


Equilibrium equation:

- Structure (Lagrangian): $\rho \partial_t^2 \mathbf{u} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$ in Ω_s
- Fluid (Eulerian) in Ω_f :
 - Equilibrium: $\rho \partial_t \mathbf{v} + \mathbf{v} \cdot \nabla \mathbf{v} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$
 - Incompressibility : $\nabla \cdot \mathbf{v} = 0$

Structure and fluid subproblems

Continuum mechanics equations

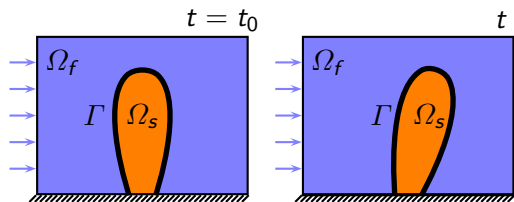


Equilibrium equation:

- Structure (Lagrangian): $\rho \partial_t^2 \mathbf{u} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$ in Ω_s
- Fluid (ALE) in $\Omega_f(t)$:
 - Equilibrium: $\rho \partial_t \mathbf{v} + (\mathbf{v} - \partial_t \mathbf{u}) \cdot \nabla \mathbf{v} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$
 - Incompressibility : $\nabla \cdot \mathbf{v} = \mathbf{0}$
 - Fluid domain motion: $\mathbf{u} = \text{Ext}(\mathbf{u}|_{\Gamma})$

Structure and fluid subproblems

Continuum mechanics equations

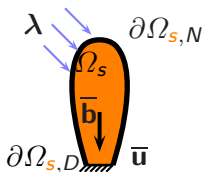


Equilibrium equation:

- Structure (Lagrangian): $\rho \partial_t^2 \mathbf{u} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$ in Ω_s
- Fluid (ALE) in $\Omega_f(t)$:
 - Equilibrium: $\rho \partial_t \mathbf{v} + (\mathbf{v} - \partial_t \mathbf{u}) \cdot \nabla \mathbf{v} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$
 - Incompressibility : $\nabla \cdot \mathbf{v} = \mathbf{0}$
 - Fluid domain motion: $\mathbf{u} = \text{Ext}(\mathbf{u}|_{\Gamma})$

How to solve each of this subproblems?

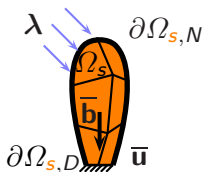
Structure and fluid subproblems



Structure discretization

- Weak formulation

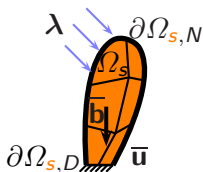
Structure and fluid subproblems



Structure discretization

- Weak formulation
- Finite Element Method [Zienkewicz, Taylor]
- Continuous elementwise polynomial functions

Structure and fluid subproblems

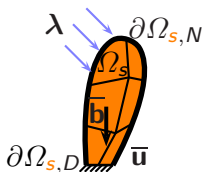


Structure discretization

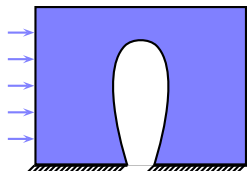
- Weak formulation
- Finite Element Method [Zienkewicz, Taylor]
- Continuous elementwise polynomial functions

Poincaré-Steklov operator: $\mathcal{S}_s^{-1} : \boldsymbol{\lambda} \longrightarrow \mathbf{u}$ [Simone, Deparis, Quateroni, 03]

Structure and fluid subproblems



Poincaré-Steklov operator: $\mathcal{S}_s^{-1} : \lambda \longrightarrow \mathbf{u}$ [Simone, Deparis, Quateroni, 03]



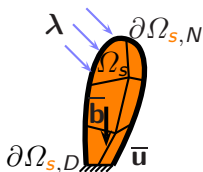
Structure discretization

- Weak formulation
- Finite Element Method [Zienkewicz, Taylor]
- Continuous elementwise polynomial functions

Fluid discretization

- Weak formulation

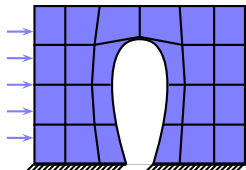
Structure and fluid subproblems



Structure discretization

- Weak formulation
- Finite Element Method [Zienkewicz, Taylor]
- Continuous elementwise polynomial functions

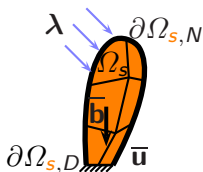
Poincaré-Steklov operator: $\mathcal{S}_s^{-1} : \boldsymbol{\lambda} \longrightarrow \mathbf{u}$ [Simone, Deparis, Quateroni, 03]



Fluid discretization

- Weak formulation
- FEM or Finite Volume Method [Ferziger, Perić]
- Discontinuous elementwise constant functions

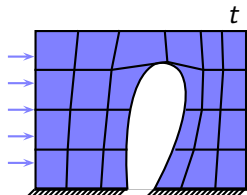
Structure and fluid subproblems



Structure discretization

- Weak formulation
- Finite Element Method [Zienkewicz, Taylor]
- Continuous elementwise polynomial functions

Poincaré-Steklov operator: $\mathcal{S}_s^{-1} : \boldsymbol{\lambda} \longrightarrow \mathbf{u}$ [Simone, Deparis, Quateroni, 03]



Fluid discretization

- Weak formulation
- FEM or Finite Volume Method [Ferziger, Perić]
- Discontinuous elementwise constant functions

Steklov-Poincaré operator: $\mathcal{S}_f : \mathbf{u} \longrightarrow \boldsymbol{\lambda} = p\mathbf{n} + \nu_f \mathbf{D}(\mathbf{v})\mathbf{n}$

Coupling equation

Steklov-Poincaré operators

Solid: $\mathcal{S}_s : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_s$

Fluid: $\mathcal{S}_f : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_f$

- Defined on $\Gamma \times [0, T]$
- Can be computed with existing tools
- Require (non-linear) computation on the whole domain Ω_s and Ω_f

Coupling equation

Steklov-Poincaré operators

Solid: $\mathcal{S}_s : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_s$

Fluid: $\mathcal{S}_f : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_f$

- Defined on $\Gamma \times [0, T]$
- Can be computed with existing tools
- Require (non-linear) computation on the whole domain Ω_s and Ω_f

Interface equations

- Displacement continuity: $\mathbf{u}_f = \mathbf{u}_s = \mathbf{u}$
- Stress equilibrium: $\boldsymbol{\sigma} \mathbf{n}_s + \boldsymbol{\sigma} \mathbf{n}_f = \mathbf{0}$

Coupling equation

Steklov-Poincaré operators

Solid: $\mathcal{S}_s : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_s$

Fluid: $\mathcal{S}_f : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_f$

- Defined on $\Gamma \times [0, T]$
- Can be computed with existing tools
- Require (non-linear) computation on the whole domain Ω_s and Ω_f

Interface equations

- Displacement continuity: $\mathbf{u}_f = \mathbf{u}_s = \mathbf{u}$
- Stress equilibrium: $\mathcal{S}_s(\mathbf{u}) + \mathcal{S}_f(\mathbf{u}) = \mathbf{0}$

Coupling equation

Steklov-Poincaré operators

Solid: $\mathcal{S}_s : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_s$

Fluid: $\mathcal{S}_f : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_f$

- Defined on $\Gamma \times [0, T]$
- Can be computed with existing tools
- Require (non-linear) computation on the whole domain Ω_s and Ω_f

Interface equations

- Displacement continuity: $\mathbf{u}_f = \mathbf{u}_s = \mathbf{u}$
- Stress equilibrium: $\mathcal{S}_s(\mathbf{u}) + \mathcal{S}_f(\mathbf{u}) = \mathbf{0}$

Solve FSI coupled problem:

- Find roots of equation: $\mathbf{u} - \mathcal{S}_s^{-1}(-\mathcal{S}_f(\mathbf{u})) = \mathbf{0}$
- Find fix-points of equation: $\mathbf{u} = \mathcal{S}_s^{-1}(-\mathcal{S}_f(\mathbf{u}))$

Coupling equation

Steklov-Poincaré operators

Solid: $\mathcal{S}_s : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_s$

Fluid: $\mathcal{S}_f : \mathbf{u} \rightarrow \boldsymbol{\lambda} = \boldsymbol{\sigma} \mathbf{n}_f$

- Defined on $\Gamma \times [0, T]$
- Can be computed with existing tools
- Require (non-linear) computation on the whole domain Ω_s and Ω_f

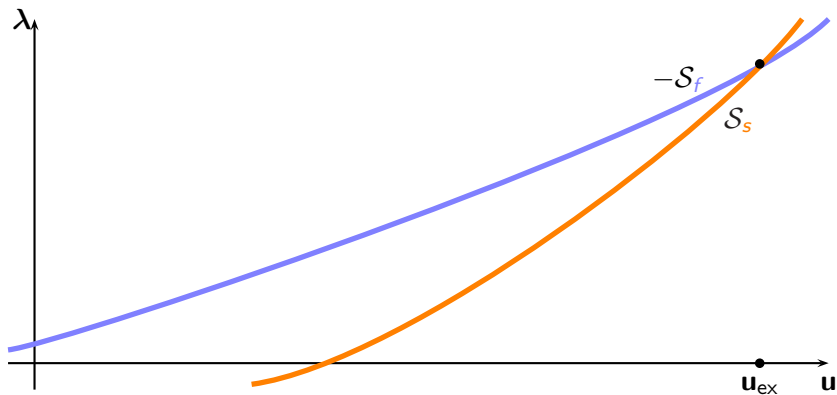
Interface equations

- Displacement continuity: $\mathbf{u}_f = \mathbf{u}_s = \mathbf{u}$
- Stress equilibrium: $\mathcal{S}_s(\mathbf{u}) + \mathcal{S}_f(\mathbf{u}) = \mathbf{0}$

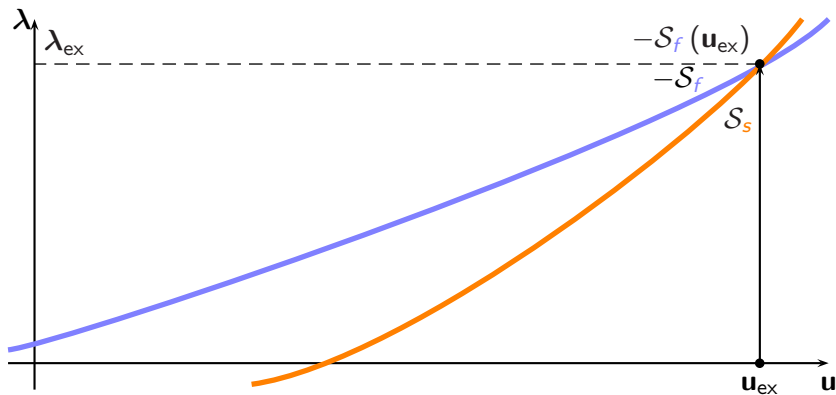
Solve FSI coupled problem:

- Find roots of equation: $\mathbf{u} - \mathcal{S}_s^{-1}(-\mathcal{S}_f(\mathbf{u})) = \mathbf{0}$
- Find fix-points of equation: $\mathbf{u} = \mathcal{S}_s^{-1}(-\mathcal{S}_f(\mathbf{u}))$

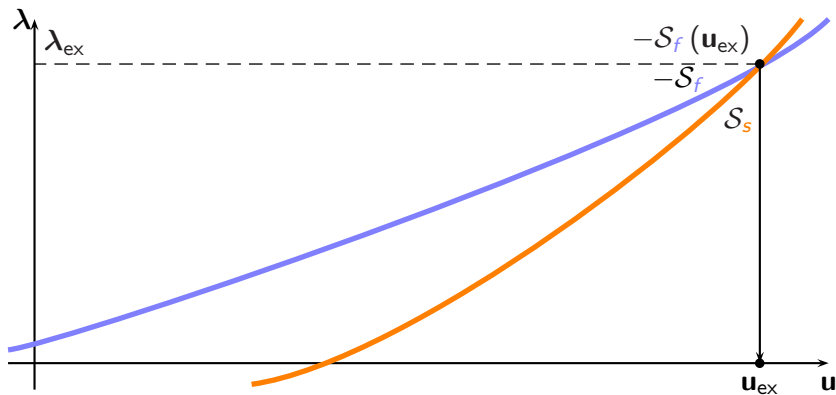
DFMT coupling algorithms – Explicit



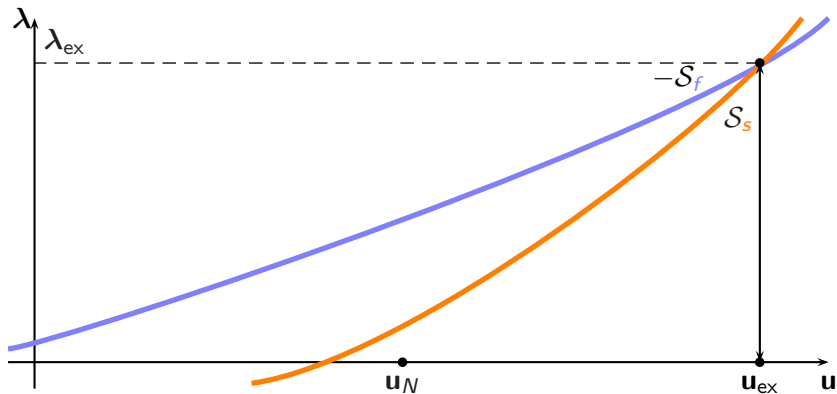
DFMT coupling algorithms – Explicit



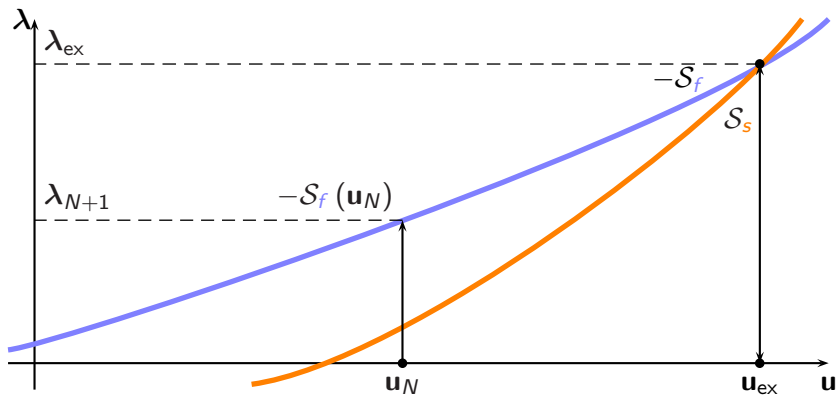
DFMT coupling algorithms – Explicit



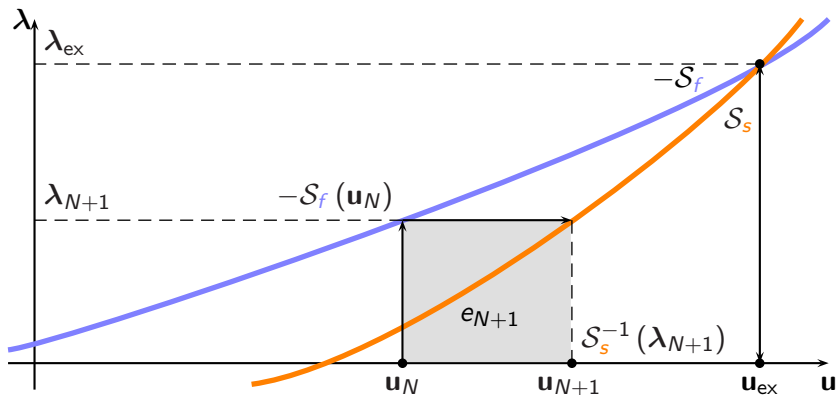
DFMT coupling algorithms – Explicit



DFMT coupling algorithms – Explicit

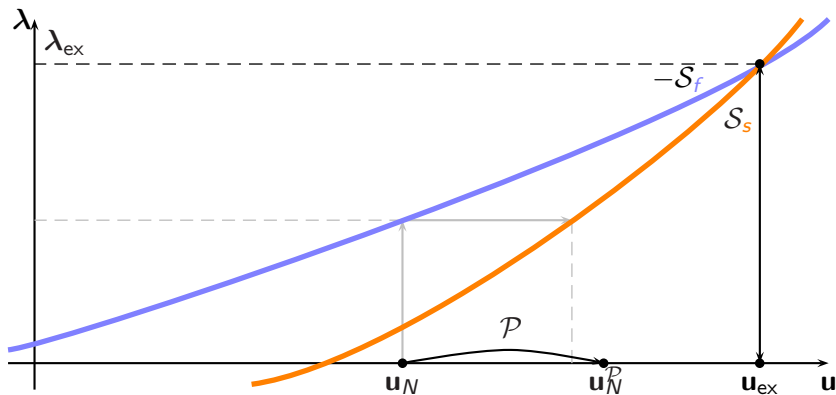


DFMT coupling algorithms – Explicit



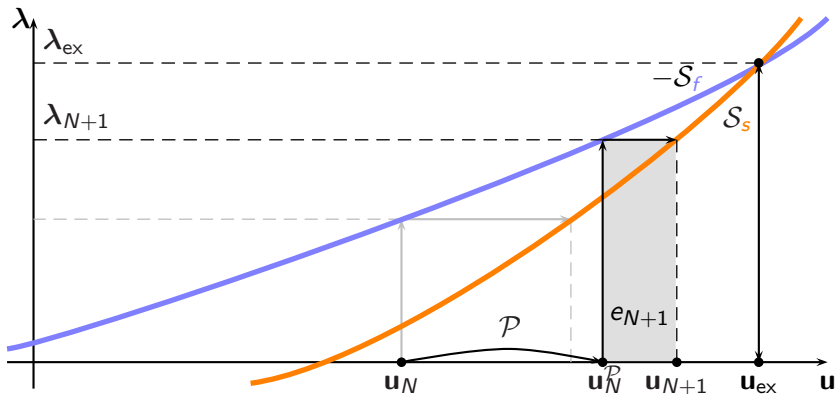
- Spurious numerical energy at the interface

DFMT coupling algorithms – Explicit



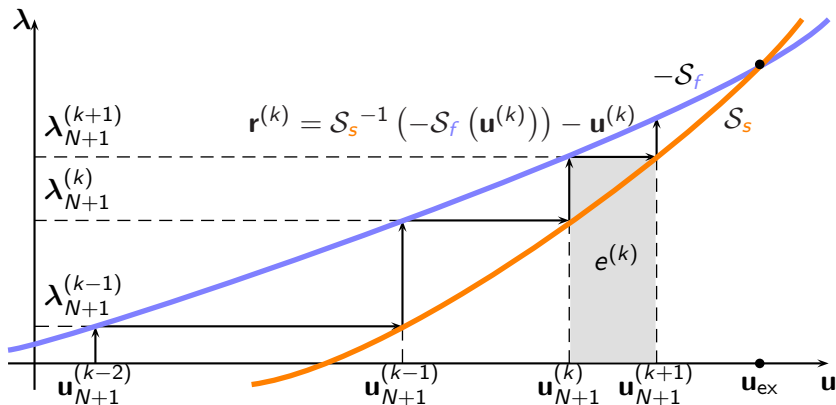
- Spurious numerical energy at the interface
- Cheap predictor computed at the interface

DFMT coupling algorithms – Explicit



- Spurious numerical energy at the interface
- Cheap predictor computed at the interface
- Function of window size, subproblem time integration schemes and predictors [Piperno & Farhat 99-03]

DFMT coupling algorithms – Implicit Block-Gauß-Seidel



- Iterations of the explicit coupling strategy
- Predictor can be used to reduce the number of iteration
- No information used for search direction (subproblem tangent terms)

Stability of the coupling algorithm ?

Stability of the coupling algorithm (DFMT-BGS)

Stability proof

- Criterion: [Arnold, 01; Steindorf, 04] *Compressible flow*

$$\left\| \mathbf{M}_s^{-1} \mathbf{M}_f \right\| \leq 1$$

- \mathbf{M}_s structure mass matrix
- \mathbf{M}_f fluid mass matrix

Stability of the coupling algorithm (DFMT-BGS)

Stability proof

- Criterion: [Arnold, 01; Steindorf, 04] *Incompressible flow*

$$\left\| \mathbf{M}_s^* \mathbf{M}_f \right\| \leq 1$$

- “Added Mass” effect [Le Tallec 01, Causin *et al.* 05, Forster *et al.* 07] :
 - No explicit coupling
 - Difficulty to make DFMT-BGS algorithm converge

- \mathbf{M}_s structure mass matrix
- \mathbf{M}_f fluid mass matrix
- $\mathbf{M}_s^* = \mathbf{M}_s (\mathbf{1} - \mathcal{F}(\mathbf{M}_f, \mathbf{B}_f))$
- \mathbf{B}_f fluid gradient matrix (associated to pressure)

Stability of the coupling algorithm (DFMT-BGS)

Stability proof

- Criterion: [Arnold, 01; Steindorf, 04] *Incompressible flow*

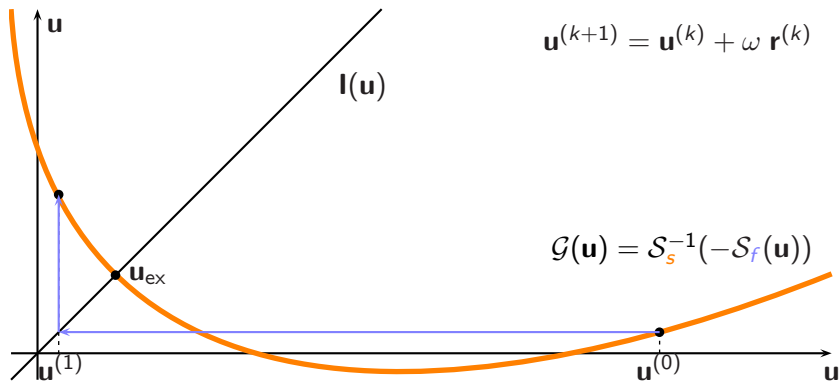
$$\left\| \mathbf{M}_s^{*-1} \mathbf{M}_f \right\| \leq 1$$

- “Added Mass” effect [Le Tallec 01, Causin *et al.* 05, Forster *et al.* 07] :

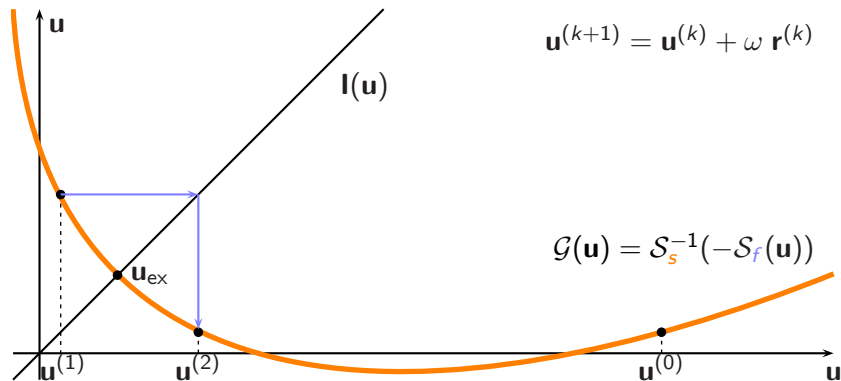
When the criterion is not fulfilled ?

- Re-ordering [Arnold, 01]
- Relaxation: *Aitken, steepest descent* [Küttler *et al.* 08]
- Preconditioning [Quateroni *et al.* 04]
- Other algorithm: *(In)-Exact Block-Newton* [Matthies 06, Dettmer & Perić, Gerbeau 03, Fernández 07]

Relaxation strategy



Relaxation strategy

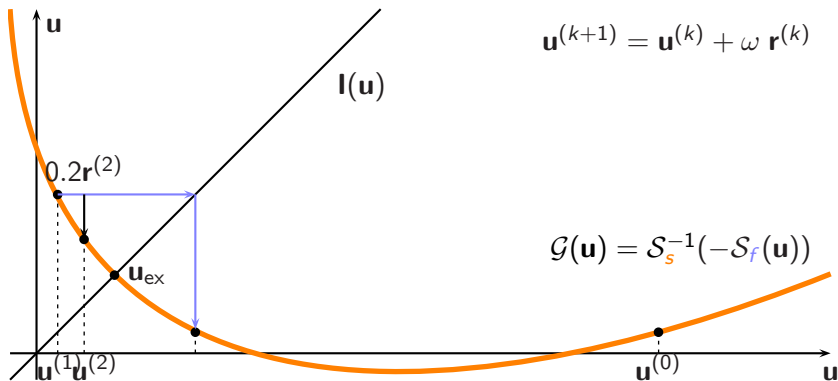


- No relaxation

Relaxation strategy

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \omega \mathbf{r}^{(k)}$$

$$\mathcal{G}(\mathbf{u}) = \mathcal{S}_s^{-1}(-\mathcal{S}_f(\mathbf{u}))$$

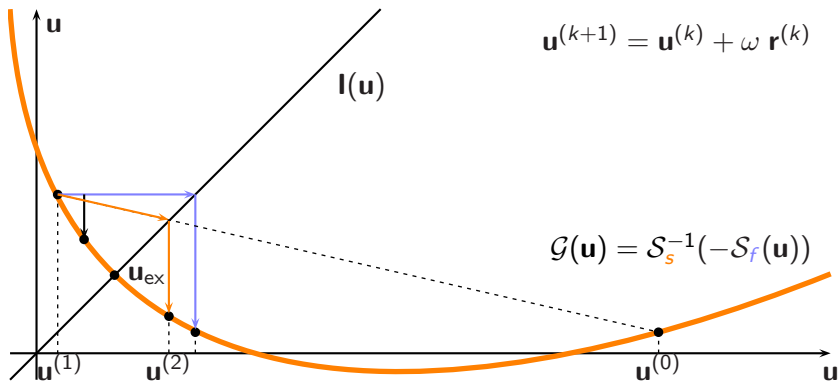


- No relaxation
- Fixed relaxation (used in pressure-velocity coupling)

Relaxation strategy

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \omega \mathbf{r}^{(k)}$$

$$\mathcal{G}(\mathbf{u}) = \mathcal{S}_s^{-1}(-\mathcal{S}_f(\mathbf{u}))$$



- No relaxation
- Fixed relaxation (used in pressure-velocity coupling)
- Aitken's relaxation (secant) [Küttler & Wall, 08]

First summary

Partitioned procedure for FSI

Fluid, structure and interface

- Structure: FEM discretized Lagrangian formulation
- Fluid: FVM discretized ALE formulation
- Interface: primal variable continuity and dual variable equilibrium

First summary

Partitioned procedure for FSI

Fluid, structure and interface

- Structure: FEM discretized Lagrangian formulation
- Fluid: FVM discretized ALE formulation
- Interface: primal variable continuity and dual variable equilibrium

Partitioned strategy for FSI

- Use of Steklov-Poincaré operators based on existing discretization
- Direct Force-Motion Transfer (DFMT) algorithms
- Block Gauss-Seidel (BGS) solver

First summary

Partitioned procedure for FSI

Fluid, structure and interface

- Structure: FEM discretized Lagrangian formulation
- Fluid: FVM discretized ALE formulation
- Interface: primal variable continuity and dual variable equilibrium

Partitioned strategy for FSI

- Use of Steklov-Poincaré operators based on existing discretization
- Direct Force-Motion Transfer (DFMT) algorithms
- Block Gauss-Seidel (BGS) solver
- **Stability criterion for coupling incompressible flows and structures**
- Conditional stability improved by dynamic relaxation

First summary

Partitioned procedure for FSI

Fluid, structure and interface

- Structure: FEM discretized Lagrangian formulation
- Fluid: FVM discretized ALE formulation
- Interface: primal variable continuity and dual variable equilibrium

Partitioned strategy for FSI

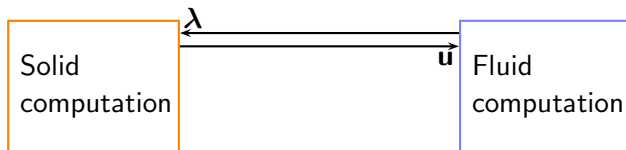
- Use of Steklov-Poincaré operators based on existing discretization
- Direct Force-Motion Transfer (DFMT) algorithms
- Block Gauss-Seidel (BGS) solver
- Stability criterion for coupling incompressible flows and structures
- Conditional stability improved by dynamic relaxation

Partitioned approach implementation and use of component technology

Outline

- 1 Fluid structure interaction framework
 - Structure and fluid subproblems
 - Explicit and implicit coupling algorithms for FSI
 - Convergence and stability of coupling algorithms
- 2 Software implementation and validation
 - Component architecture cops
 - Lid driven-cavity with a flexible bottom
 - Oscillating appendix in a flow
- 3 Applications: 3D computations and interaction with free surface flows
 - Three dimensional computing and paralleling
 - Solving free surface flows
 - Examples: free-surface flows impacting structures

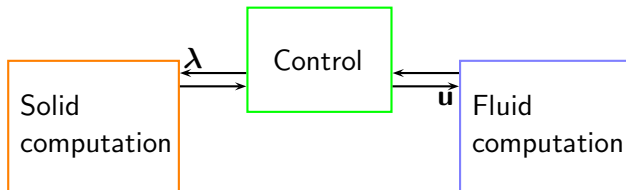
Programming context for partitioned solution procedure



FSI software implementation

- Data exchange between fluid and structure computations

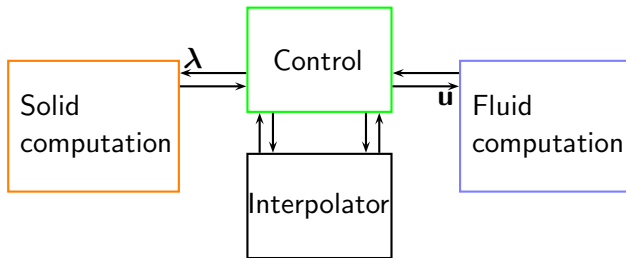
Programming context for partitioned solution procedure



FSI software implementation

- Data exchange between fluid and structure computations
- Implementation of a master code

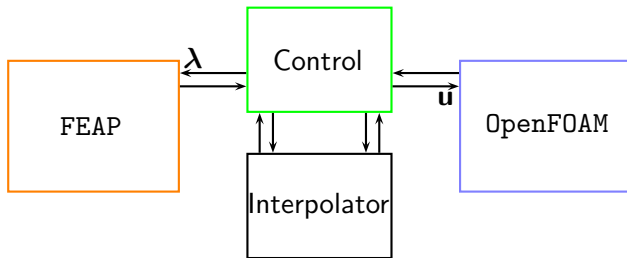
Programming context for partitioned solution procedure



FSI software implementation

- Data exchange between fluid and structure computations
- Implementation of a master code
- Non matching meshes handled by the Interpolator

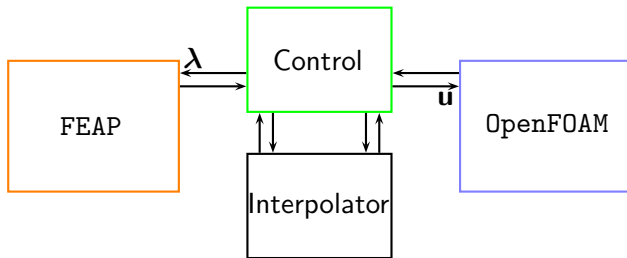
Programming context for partitioned solution procedure



FSI software implementation

- Data exchange between fluid and structure computations
- Implementation of a master code
- Non matching meshes handled by the Interpolator
- Re-using existing fluid and structure codes

Programming context for partitioned solution procedure

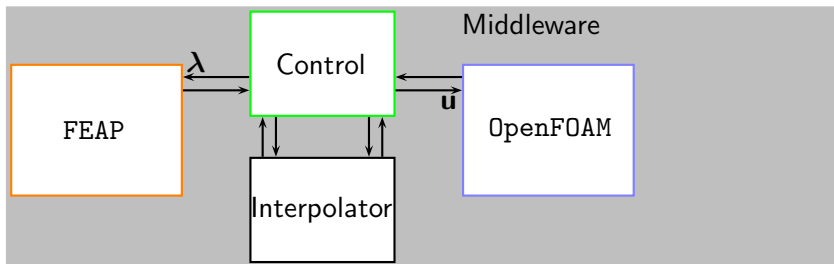


FSI software implementation

- Data exchange between fluid and structure computations
- Implementation of a master code
- Non matching meshes handled by the Interpolator
- Re-using existing fluid and structure codes

Minimum requirement: a communication protocol

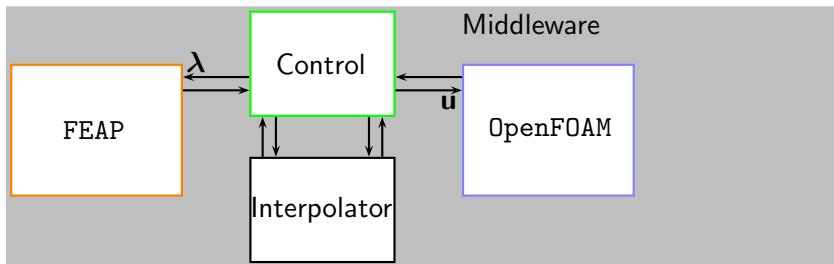
Programming context for partitioned solution procedure



Middleware – Software component technology

- “Between” software and hardware
- Computer science community [Mac Ilroy 68, Szyperski & Meeserschmitt 98]
- Each software: a **component**
- Generalization of OOP to software: encapsulated / interface
- Middleware in charge of communication and data types

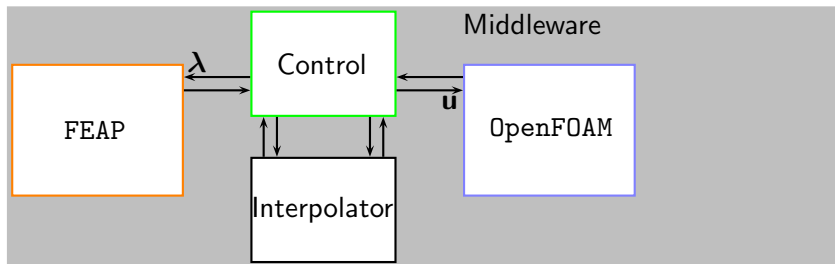
Programming context for partitioned solution procedure



Middleware – for scientific computing

- Available middleware: Corba, Java-RMI, MS.net ...
- Communication Template Library (CTL): C++ [Niekamp, 02]

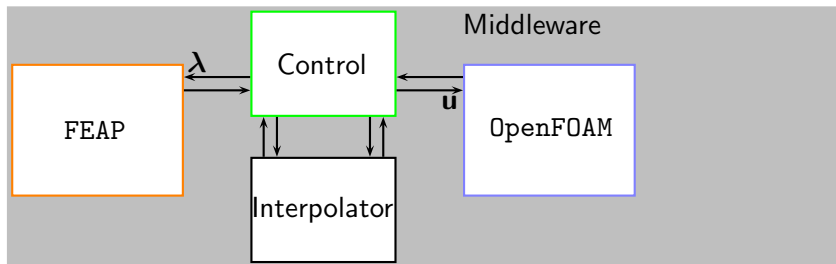
Programming context for partitioned solution procedure



Middleware – for scientific computing

- Available middleware: [Corba](#), Java-RMI, MS.net ...
- [Communication Template Library \(CTL\)](#): C++ [Niekamp, 02]
- Scientific computing: requires good performances [Niekamp, 05]

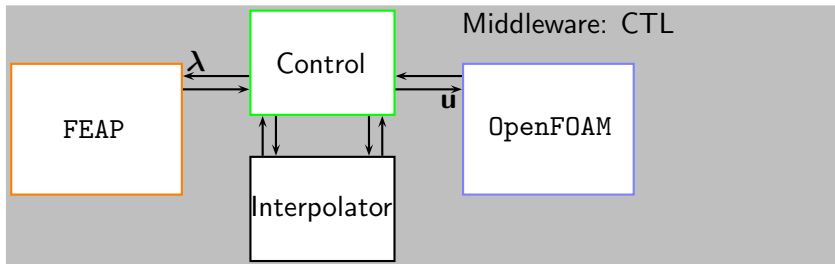
Programming context for partitioned solution procedure



Middleware – for scientific computing

- Available middleware: [Corba](#), Java-RMI, MS.net ...
- [Communication Template Library \(CTL\)](#): C++ [Niekamp, 02]
- Scientific computing: requires good performances [Niekamp, 05]
- Salomé platform (ÉDF R&D)

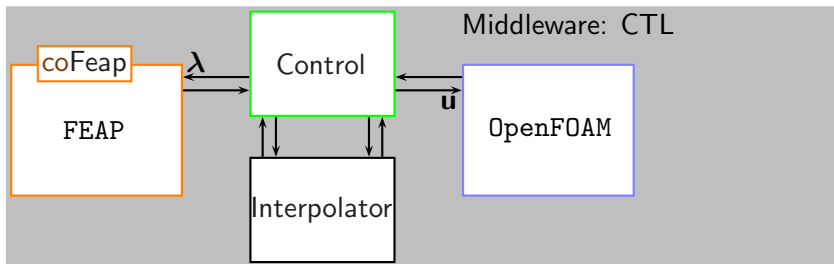
Programming context for partitioned solution procedure



Middleware – for scientific computing

- Available middleware: Corba, Java-RMI, MS.net ...
- **Communication Template Library (CTL): C++** [Niekamp, 02]
- Scientific computing: requires good performances [Niekamp, 05]
- Salomé platform (ÉDF R&D)
- Software development made by non-programmers

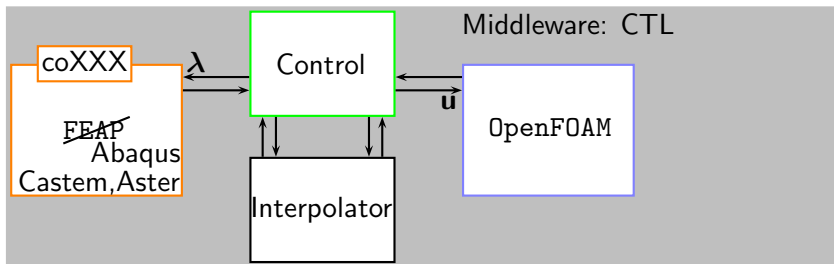
Components implementation and use



Structure component: **coFeap** [Kassiotis & Hautefeuille 08]

- Interface definition `simu.ci`

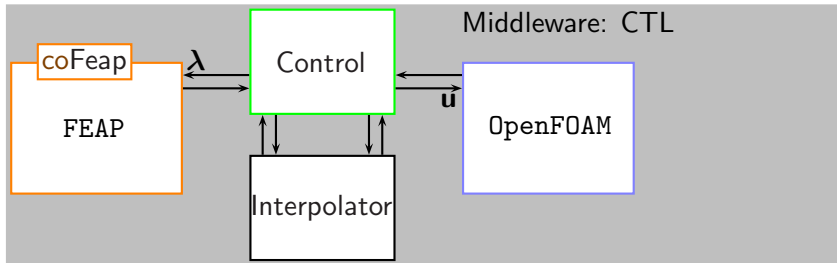
Components implementation and use



Structure component: `coFeap` [Kassiotis & Hautefeuille 08]

- Interface definition `simu.ci` (Genericity)

Components implementation and use



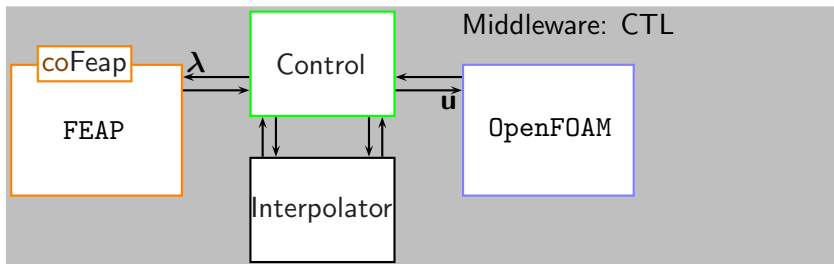
Structure component: coFeap [Kassiotis & Hautefeuille 08]

- Interface definition `simu.ci` (Genericity)
- Methods declaration

```
#define CTL_Method6 void, set_load,
                    (const array<scalar1>/*value*/), 1
```

- Methods implementation in Fortran

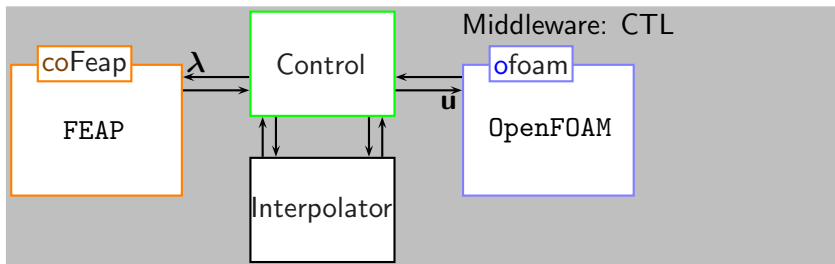
Components implementation and use



Structure component: coFeap [Kassiotis & Hautefeuille 08]

- Compilation gives:
 - A library: call like a `lib`, `thread` (asynchronous calls)
 - An executable: remote call with `tcp`, `pipe`, `MPI`...
- Use: *Multiscale* [Hautefeuille 09], *EFEM* [Benkemoun 09], *Stochastic* [Krosche 09], *Thermomechanics* [Kassiotis 06], *Mass transfer* [De Sa 08] ...

Components implementation and use



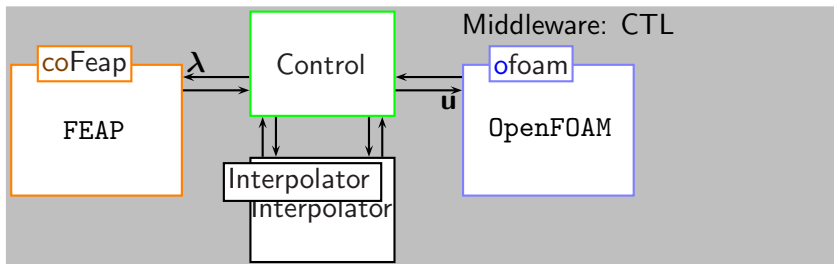
Fluid component: `ofoam` [Krosche 07, Kassiotis 09]

- Interface definition can be derivated from `simu.ci`: `CFDsimu.ci`
- Methods declaration

```
#define CTL_Method2 void, get,
    ( const string /*name*/, array<real8> /*v*/ ) const, 2
```

- Methods implementation in C++

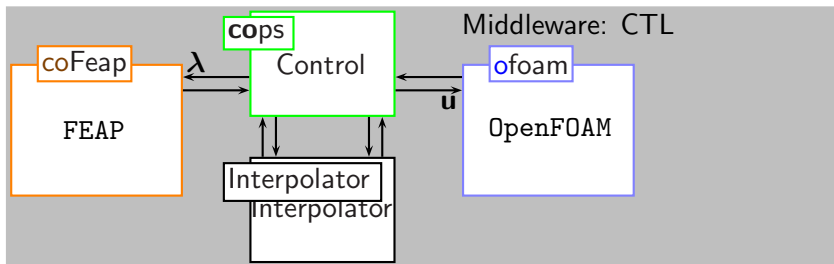
Components implementation and use



Interpolation component: Interpolator [Jürgens 09]

- C++ component
- Interpolation with radial basis functions [Beckert & Wendland 01]
- Full matrices
- Solve: coupled with the Lapack library

Components implementation and use

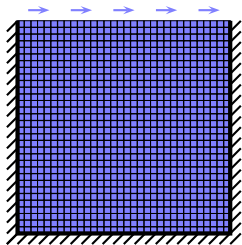


COupling COmponents by a Partitioned Strategy: **cops**

- Coupling components as templates
- Implementation of DFMT coupling algorithm
- Explicit coupling: collocated and non-collocated
- Implicit coupling: BGS
- Predictors (order 0 to 2), fixed and dynamic Aitken's relaxation

Lid-driven cavity with a flexible bottom

Problem parameters

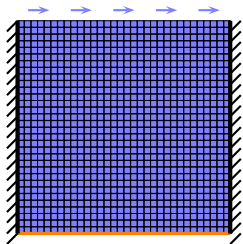


Fluid problem

- Material properties:
 $\rho_f = 1\text{kg}\cdot\text{m}^{-3}$, $\nu_f = 0.01\text{m}\cdot\text{s}^{-2}$.
- Boundary conditions:
 - only ∇p required
 - $\mathbf{v} \cdot \mathbf{e}_x = 1 - \cos(2\pi t/T_{\text{char}})$
- Accurate discretization when $\mathcal{R}e \leq 300$

Lid-driven cavity with a flexible bottom

Problem parameters



Fluid problem

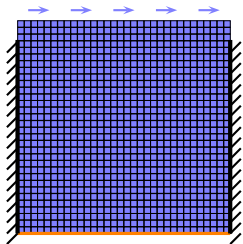
- Material properties:
 - $\rho_f = 1\text{kg}\cdot\text{m}^{-3}$, $\nu_f = 0.01\text{m}\cdot\text{s}^{-2}$.
- Boundary conditions:
 - only ∇p required
 - $\mathbf{v} \cdot \mathbf{e}_x = 1 - \cos(2\pi t/T_{\text{char}})$
- Accurate discretization when $\text{Re} \leq 300$

Modification for the FSI case

- Structure problem: $\rho_s = 500\text{kg}\cdot\text{m}^{-3}$, $E_s = 250\text{Pa}$ and $\nu_s = 0$

Lid-driven cavity with a flexible bottom

Problem parameters



Fluid problem

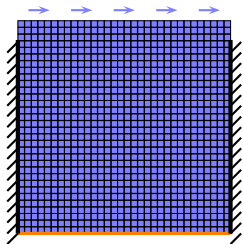
- Material properties:
 - $\rho_f = 1 \text{ kg} \cdot \text{m}^{-3}$, $\nu_f = 0.01 \text{ m} \cdot \text{s}^{-2}$.
- Boundary conditions:
 - only ∇p required
 - $\mathbf{v} \cdot \mathbf{e}_x = 1 - \cos(2\pi t / T_{\text{char}})$
- Accurate discretization when $\text{Re} \leq 300$

Modification for the FSI case

- Structure problem: $\rho_s = 500 \text{ kg} \cdot \text{m}^{-3}$, $E_s = 250 \text{ Pa}$ and $\nu_s = 0$
- No incompressibility dilemma [Wall et al. 98, Gerbeau & Vidrascu 03]
- Pressure fix (different from [Bathe & Zhang 09])

Lid-driven cavity with a flexible bottom

Results



Discretization

- Fluid: 32x32 cells.
- Structure: 16 quadratic elements.
- Time step: $\Delta t = 0.1s$.

Lid-driven cavity with a flexible bottom

Results



Discretization

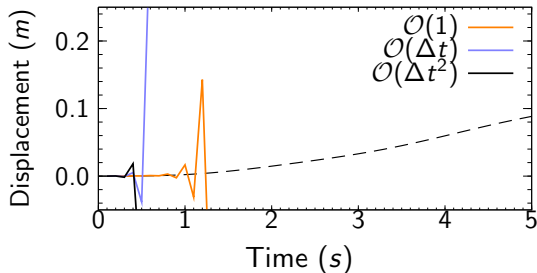
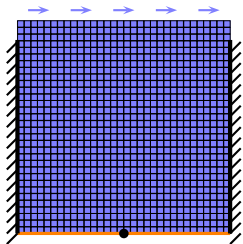
- Fluid: 32x32 cells.
- Structure: 16 quadratic elements.
- Time step: $\Delta t = 0.1s$.

Perfect benchmark for FSI

- Mesh simplicity
- Computational time: $T_s^{\text{CPU}} = 2.95 \times 10^{-3}s$ and $T_f^{\text{CPU}} = 1.08 \times 10^{-1}s$
- Harmonic solution quickly reached

Lid-driven cavity with a flexible bottom

Explicit results

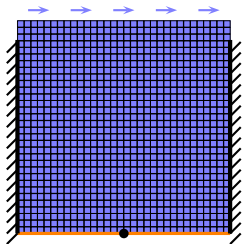


Influence of numerical parameters

- Order of predictor
- Time step size
- Time integration of the fluid problem
- Non-collocated schemes

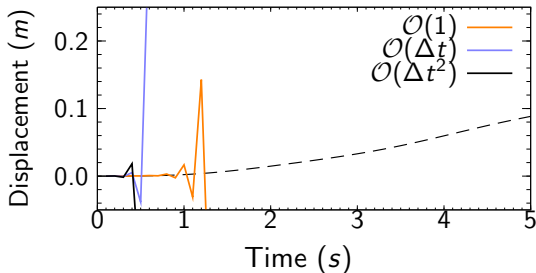
Lid-driven cavity with a flexible bottom

Explicit results



Added mass effect

no explicit coupling when
incompressible flow
interacts with structure

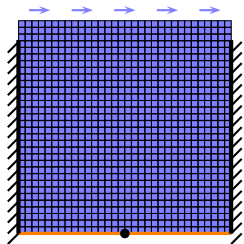


Influence of numerical parameters

- Order of predictor
- Time step size
- Time integration of the fluid problem
- Non-collocated schemes

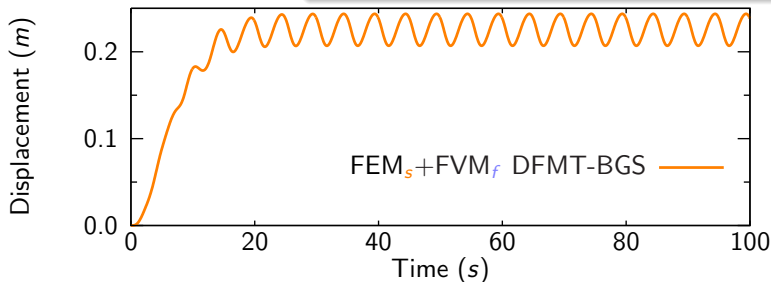
Lid-driven cavity with a flexible bottom

Implicit results



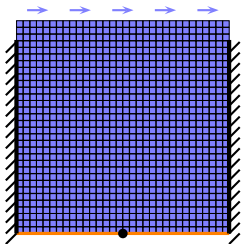
Numerical parameters

- Interface residual: $\|\mathbf{r}_N^{(k)}\|_2 \leq 1 \times 10^{-7}$
- All converged computations: same results



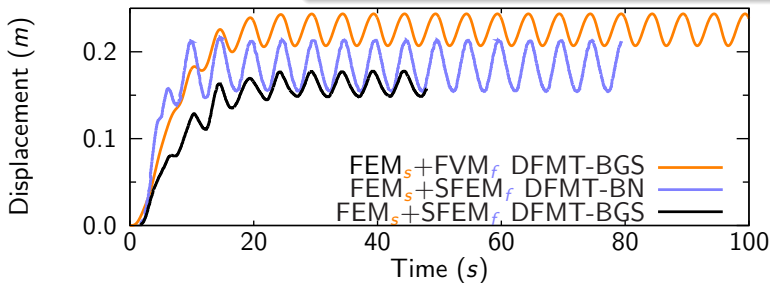
Lid-driven cavity with a flexible bottom

Implicit results



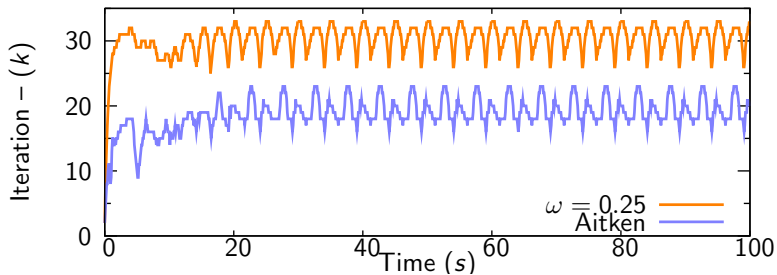
Numerical parameters

- Interface residual: $\|\mathbf{r}_N^{(k)}\|_2 \leq 1 \times 10^{-7}$
- All converged computations: same results
- Results with other methods [Gerbeau & Vidrascu 03, Wall & Mok 99]



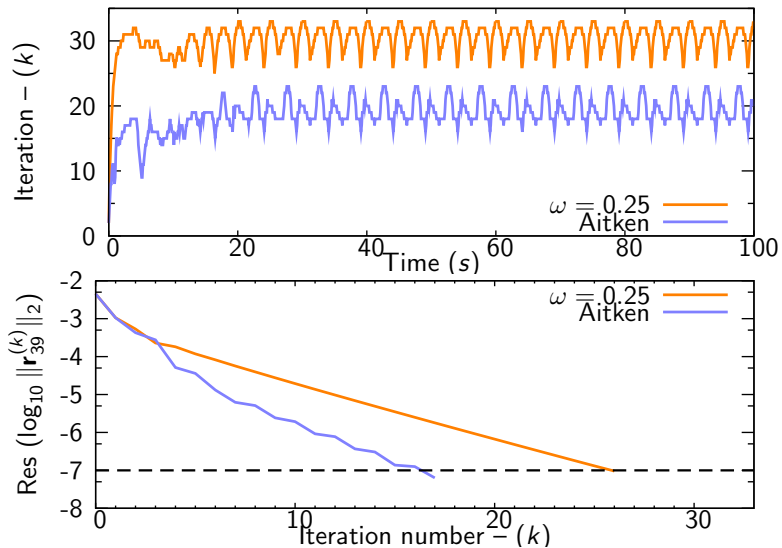
Lid-driven cavity with a flexible bottom

Implicit results – Aitken's relaxation



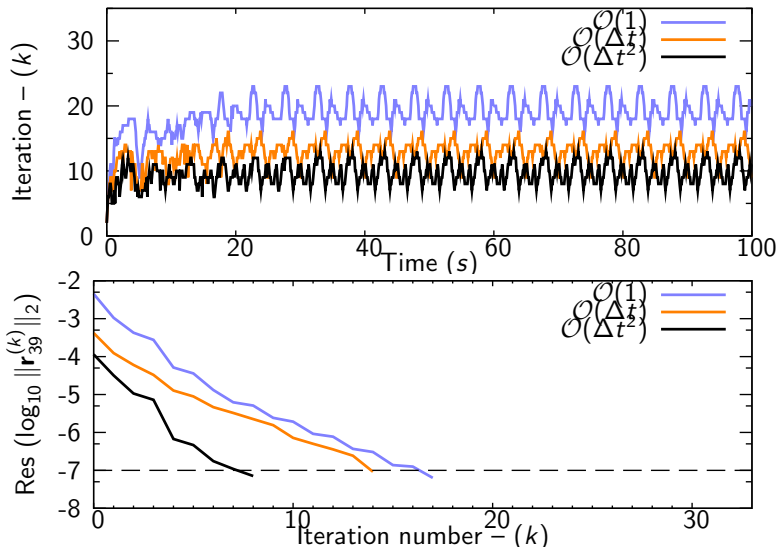
Lid-driven cavity with a flexible bottom

Implicit results – Aitken's relaxation



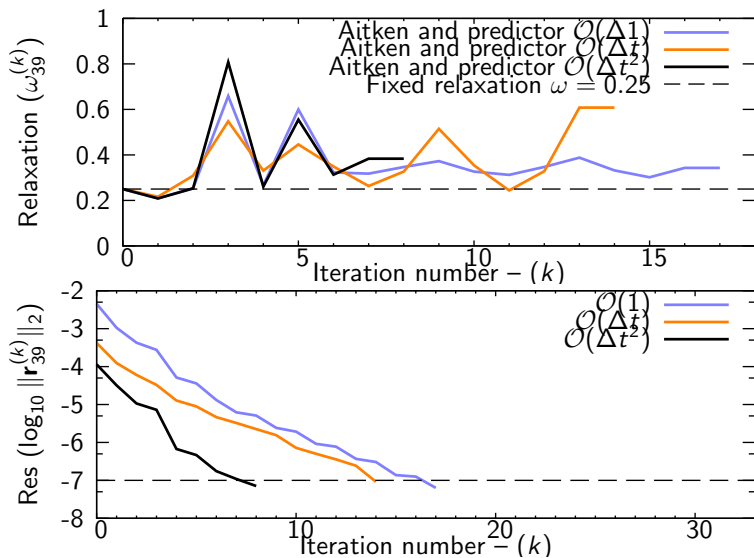
Lid-driven cavity with a flexible bottom

Implicit results – Predictors



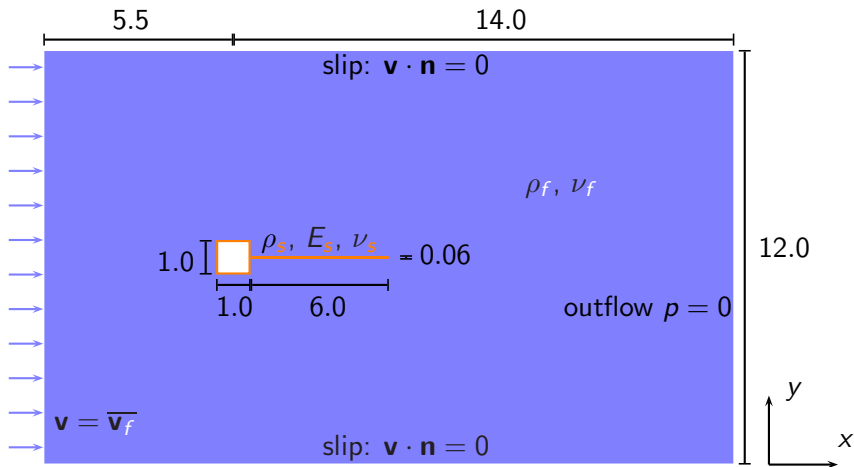
Lid-driven cavity with a flexible bottom

Implicit results – Predictors



Oscillating appendix

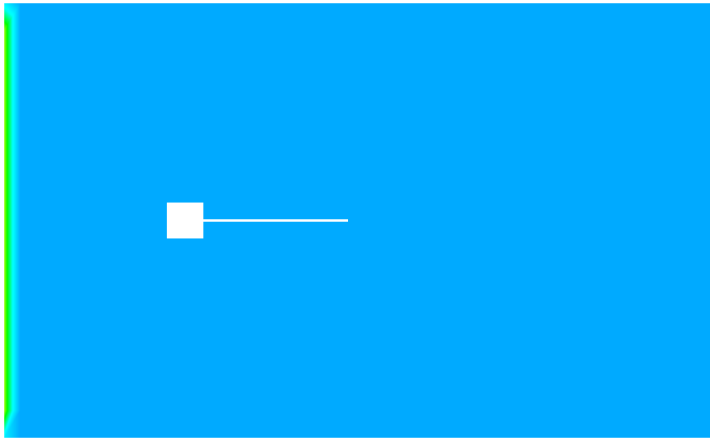
Problem presentation



Implicit/Explicit coupling

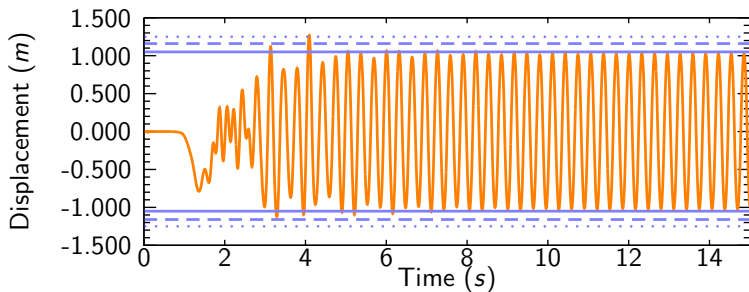
Oscillating appendix

Results



Oscillating appendix

Computation results



Comparison with other works (Maximum amplitude motion)

- $FEM_s + FVM_f$ DFMT-BGS ————
- $FEM_s + SFEM_f$ DFMT-BGS - - - - [Wall & Ramm 99]
- $FEM_s + SFEM_f$ DFMT-BN ———— [Steindorf & Matthies 02]
- $FEM_s + SFEM_f$ Monolithical ······ [Dettmer & Perić 07]

Second summary

From a partitioned solution procedure to a component architecture

Software implementation

- Suited for partitioned strategy with high performance data transfers
- Middleware CTL simplifies communication
- Component technology: re-use of existing codes

Second summary

From a partitioned solution procedure to a component architecture

Software implementation

- Suited for partitioned strategy with high performance data transfers
- Middleware CTL simplifies communication
- Component technology: re-use of existing codes

Validation and comparison with other strategies

- Full definition of an adapted benchmark to validate FSI implementation
- Implicit coupling required for incompressible flows interacting with structures required
- Behavior of DMFT-BGS with dynamic relaxation validated
- Comparison with other approaches gives similar qualitative results

Second summary

From a partitioned solution procedure to a component architecture

Software implementation

- Suited for partitioned strategy with high performance data transfers
- Middleware CTL simplifies communication
- Component technology: re-use of existing codes

Validation and comparison with other strategies

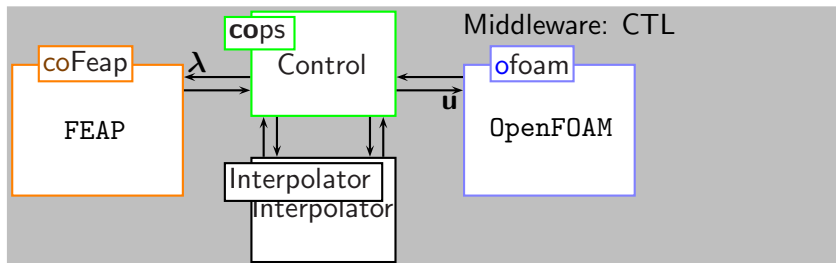
- Full definition of an adapted benchmark to validate FSI implementation
- Implicit coupling required for incompressible flows interacting with structures required
- Behavior of DMFT-BGS with dynamic relaxation validated
- Comparison with other approaches gives similar qualitative results

Advantages of re-using: efficient solvers and advanced models

Outline

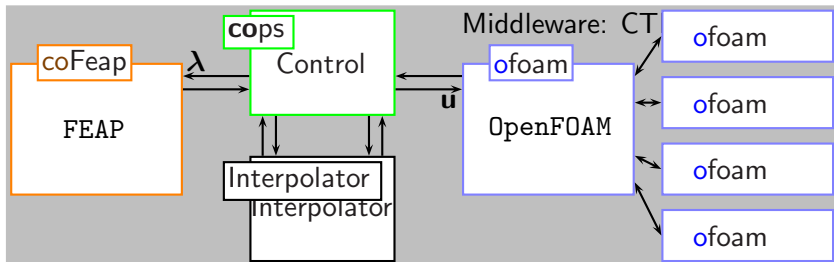
- 1 Fluid structure interaction framework
 - Structure and fluid subproblems
 - Explicit and implicit coupling algorithms for FSI
 - Convergence and stability of coupling algorithms
- 2 Software implementation and validation
 - Component architecture cops
 - Lid driven-cavity with a flexible bottom
 - Oscillating appendix in a flow
- 3 Applications: 3D computations and interaction with free surface flows
 - Three dimensional computing and paralleling
 - Solving free surface flows
 - Examples: free-surface flows impacting structures

Performances and paralleling



Lid-cavity T^{CPU} : Structure 3%, Fluid 96% and Interpolation 1%.

Performances and paralleling

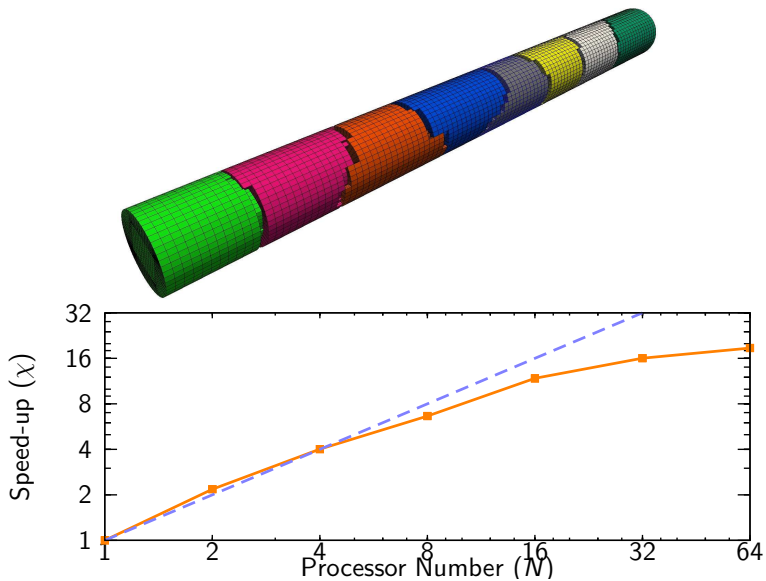


Lid-cavity T^{CPU} : Structure 3%, Fluid 96% and Interpolation 1%.

A parallel version of ofoam

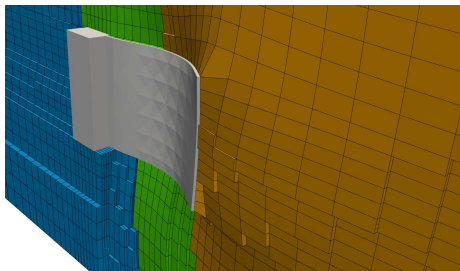
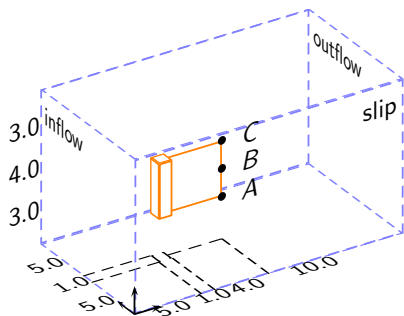
- Based on OpenFOAM inner paralleling (MPI)
- Derive a parallel interface `CFDsиму.pi` from standard interface
- Group of workers instantiation and communication handled by CTL
- Call parallel version transparent for client

Performances and paralleling



Three-dimensional “flag” in the wind

Problem parameters

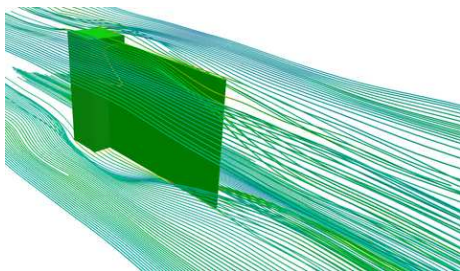
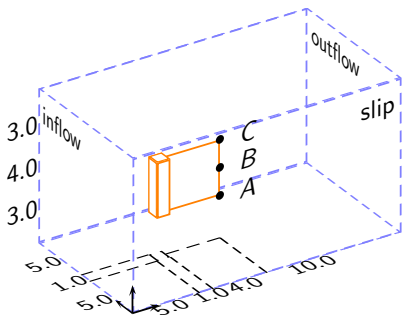


Numerical parameters

- Implicit DFMT-BGS coupling
- Interface: $\|\mathbf{r}_N^{(k)}\|_2 \leq 1 \times 10^{-7}$
- Discretization: 150×10^3 or 1.2×10^6 *d-o-f*, 6×10^3 *time step*
- Paralleling of the fluid sub-problem

Three-dimensional “flag” in the wind

Problem parameters

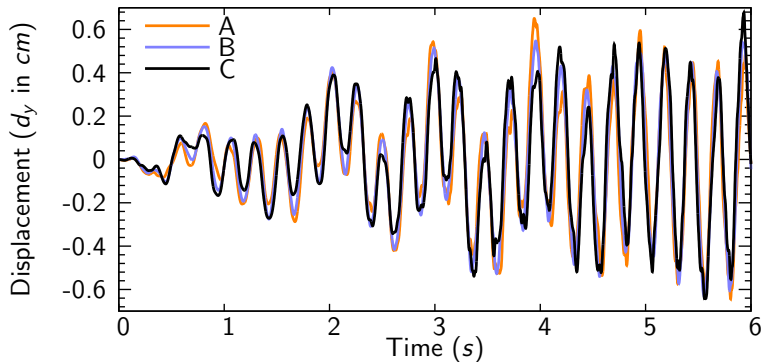


Numerical parameters

- Implicit DFMT-BGS coupling
- Interface: $\|\mathbf{r}_N^{(k)}\|_2 \leq 1 \times 10^{-7}$
- Discretization: 150×10^3 or 1.2×10^6 *d-o-f*, 6×10^3 *time step*
- Paralleling of the fluid sub-problem

Three-dimensional “flag” in the wind

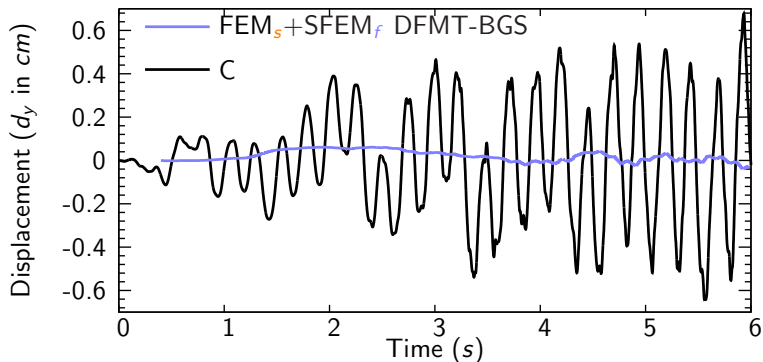
Computation results



- First flexion mode

Three-dimensional “flag” in the wind

Computation results



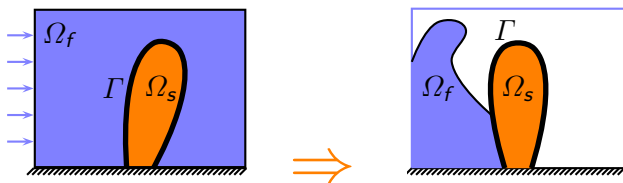
- First flexion mode
- Different from the torsional mode observed [von Scheven, 09]
- Complex flow, different structure model, sensitivity to initial condition. . .

Outline

- 1 Fluid structure interaction framework
 - Structure and fluid subproblems
 - Explicit and implicit coupling algorithms for FSI
 - Convergence and stability of coupling algorithms
- 2 Software implementation and validation
 - Component architecture cops
 - Lid driven-cavity with a flexible bottom
 - Oscillating appendix in a flow
- 3 Applications: 3D computations and interaction with free surface flows
 - Three dimensional computing and paralleling
 - Solving free surface flows
 - Examples: free-surface flows impacting structures

Structure and free surface flow subproblems

Continuum mechanics equations

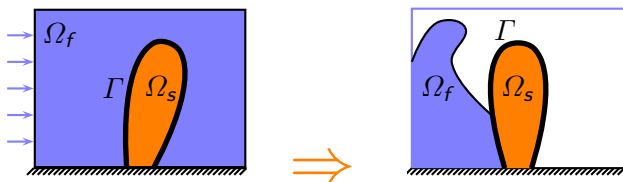


Problem equations:

- Structure (Lagrangian): $\rho \partial_t^2 \mathbf{u} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$ dans Ω_s
- Fluid (ALE) in Ω_f :
 - Equilibrium: $\rho \partial_t \mathbf{v} + (\mathbf{v} - \partial_t \mathbf{u}) \cdot \nabla \mathbf{v} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$
 - Incompressibility : $\nabla \cdot \mathbf{v} = 0$
 - Fluid domain motion: $\mathbf{u} = \text{Ext}(\mathbf{u}|_Gamma)$

Structure and free surface flow subproblems

Continuum mechanics equations

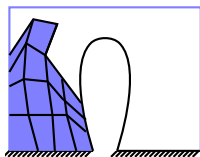


Problem equations:

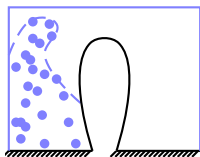
- Structure (Lagrangian): $\rho \partial_t^2 \mathbf{u} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \mathbf{0}$ dans Ω_s
- Fluid (ALE) in $\Omega_f(t)$:
 - Equilibrium: $\rho \partial_t \mathbf{v} + (\mathbf{v} - \partial_t \mathbf{u}) \cdot \nabla \mathbf{v} - \nabla \cdot \boldsymbol{\sigma} - \mathbf{f} = \sigma \kappa \delta_\Gamma \mathbf{n} + \rho \mathbf{g}$
 - Incompressibility : $\nabla \cdot \mathbf{v} = 0$
 - Fluid domain motion: $\mathbf{u} = \text{Ext}(\mathbf{u}|_\Gamma)$
 - Characteristic function: $\partial_t \iota + (\mathbf{v} - \partial_t \mathbf{u}) \cdot \nabla \iota = 0$ and normal $\mathbf{n} = \nabla \iota$

Structure and fluid subproblems

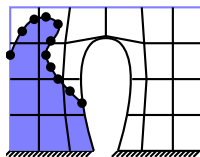
Discretization



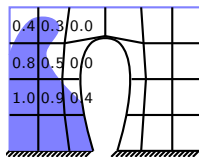
1



2



3



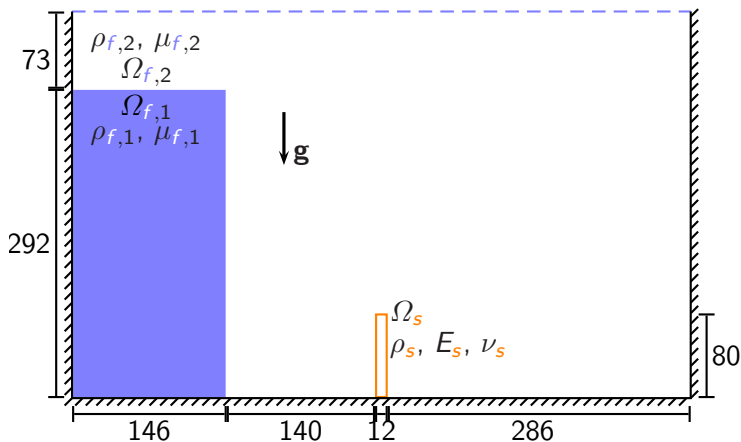
4

Discretization strategies

- ① Moving grid method: *PFEM* [Idelsohn 04]
- ② Meshless method: *SPH* [Monaghan 88, Fries 05]
- ③ Tracking surface method: *Surface fitted method* [Ferziger & Perić 96]
- ④ Tracking volume method: *V.O.F.* [Ghidaglia 01, Rusche 02, Duthyk 08]

Two-dimensional dam-break problem

Problem parameters



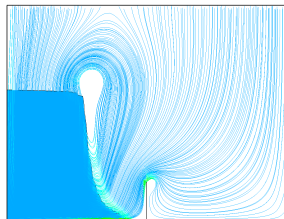
Structure neo-Hookean $E_s = 1 \times 10^6 \text{ Pa}$, $\nu_s = 0$, $\rho_s = 2500 \text{ kg} \cdot \text{m}^{-3}$.

Fluid $\rho_{f,1} = 1 \times 10^3 \text{ kg} \cdot \text{m}^{-3}$, $\nu_{f,1} = 1 \times 10^6 \text{ m} \cdot \text{s}^{-1}$,
 $\rho_{f,2} = 1 \text{ kg} \cdot \text{m}^{-3}$, $\nu_{f,2} = 1 \times 10^5 \text{ m} \cdot \text{s}^{-1}$.

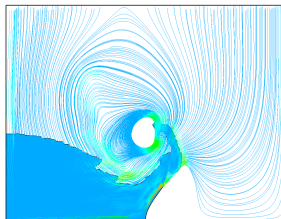
Two-dimensional dam-break problem

Results

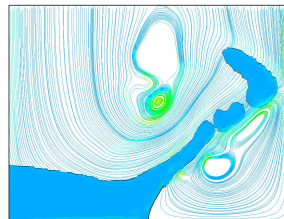
$t = 0.1s$



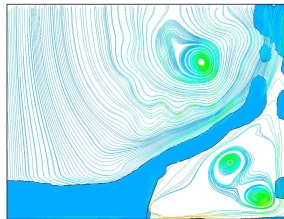
$t = 0.2s$



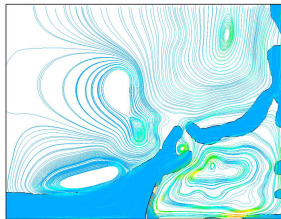
$t = 0.3s$



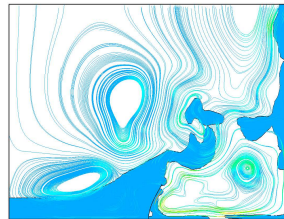
$t = 0.4s$



$t = 0.5s$

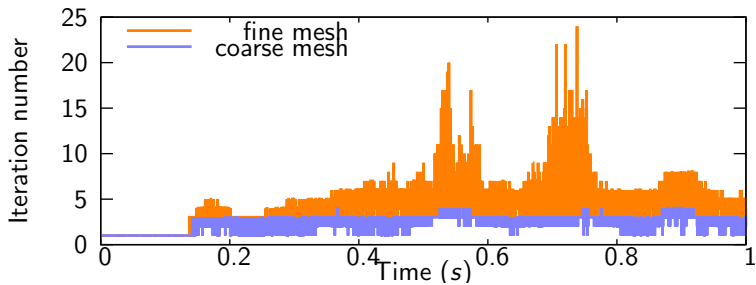
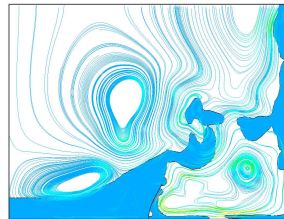
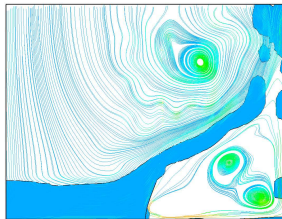
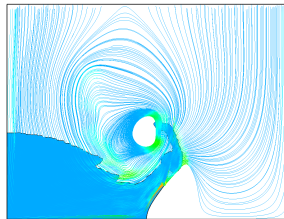


$t = 0.6s$



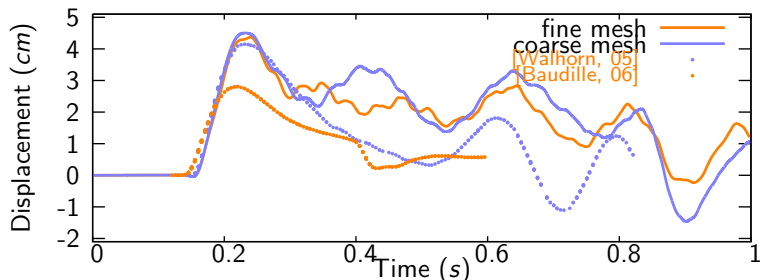
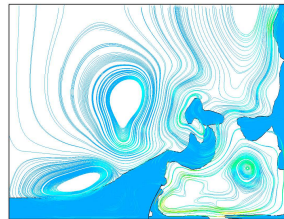
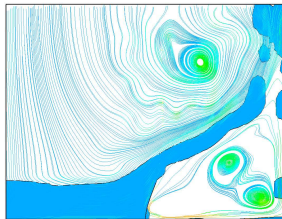
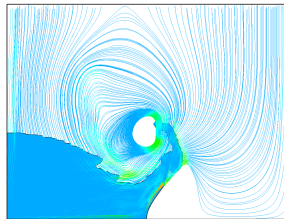
Two-dimensional dam-break problem

Computation results

 $t = 0.2s$ $t = 0.4s$ $t = 0.6s$ 

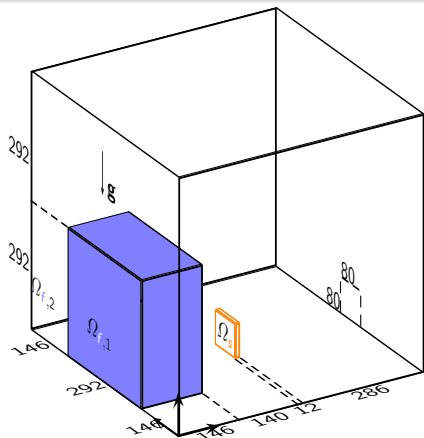
Two-dimensional dam-break problem

Computation results

 $t = 0.2s$ $t = 0.4s$ $t = 0.6s$ 

Three-dimensional dam-break problem

Problem parameters



Parameters

- Free-outflow boundaries
- Discretization:
 - 64×10^3 or 526×10^3 *d-o-f*
 - 1×10^5 *time step*
- Multigrid solver for the fluid part
- Interface: $\|\mathbf{r}_N^{(k)}\|_2 \leq 1 \times 10^{-6}$

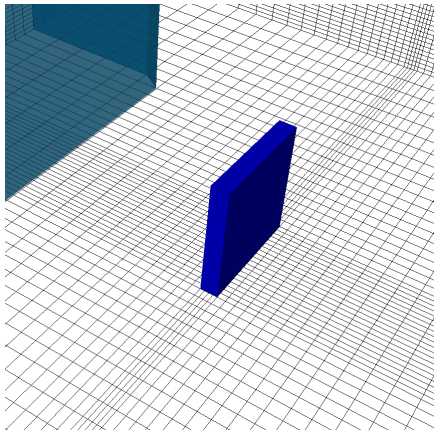
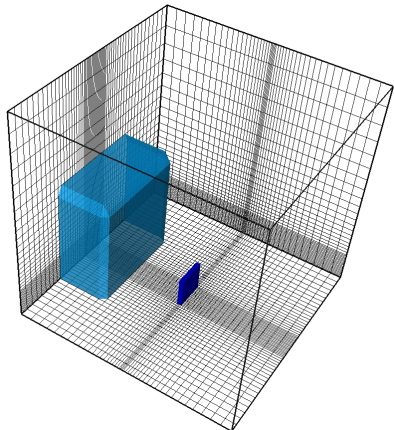
Structure *neo-Hookean* $E_s = 1 \times 10^6 \text{ Pa}$, $\nu_s = 0$, $\rho_s = 2500 \text{ kg} \cdot \text{m}^{-3}$.

Fluid $\rho_{f,1} = 1 \times 10^3 \text{ kg} \cdot \text{m}^{-3}$, $\nu_{f,1} = 1 \times 10^6 \text{ m} \cdot \text{s}^{-1}$,
 $\rho_{f,2} = 1 \text{ kg} \cdot \text{m}^{-3}$, $\nu_{f,2} = 1 \times 10^5 \text{ m} \cdot \text{s}^{-1}$.

Three-dimensional dam-break problem

Results

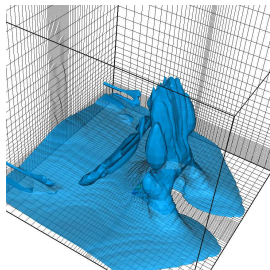
Isosurface $\iota = 0.5$



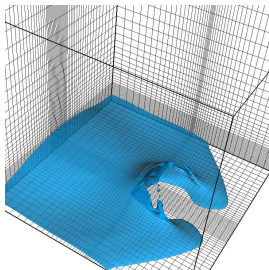
Three-dimensional dam-break problem

Results

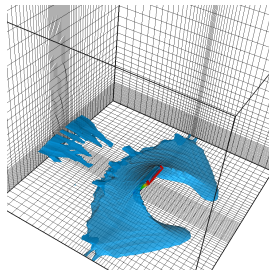
Free-surface representation



● $t = 0.01$



● $t = 0.50$

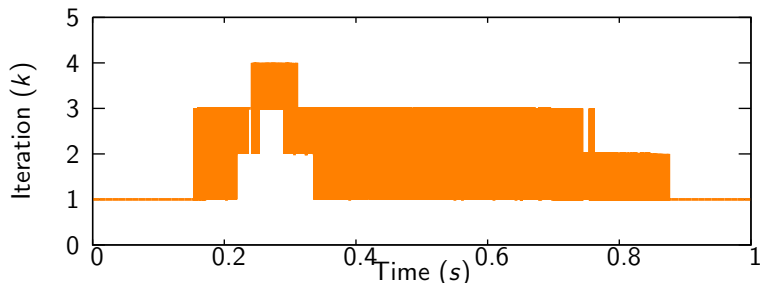


● $t = 0.99$

- Visualization of a qualitative free-surface
- Water mass is conserved

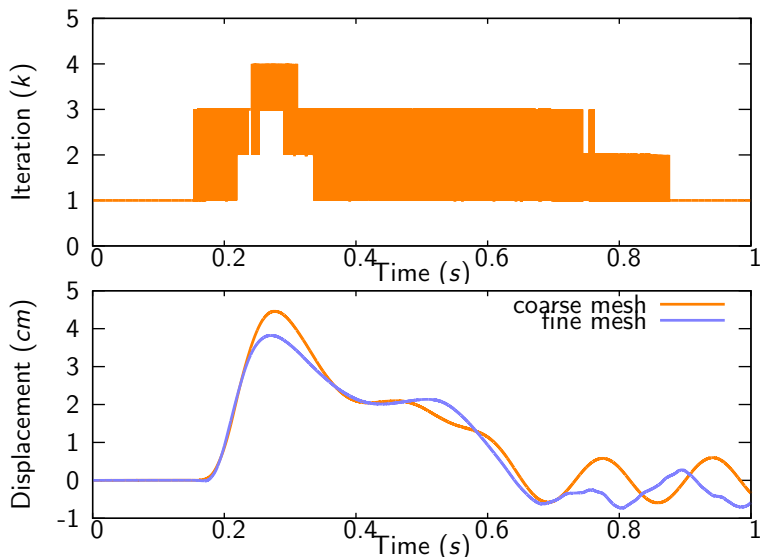
Three-dimensional dam-break problem

Computation results



Three-dimensional dam-break problem

Computation results



Coupling with concrete civil engineering structures

Localization limiters / Crack representation

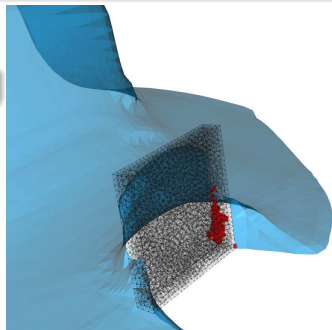
- Smearred crack model [Hidelborg *et al* 77]
- Cohesive zone model [Barenblatt, 62]
- Non-local approach [Pijaudier-Cabot and Bažant, 87]
- EFEM [Wells & Sluys, 00] / XFEM [Moës *et al*, 99]

Coupling with concrete civil engineering structures

Localization limiters / Crack representation

- Smearred crack model [Hidelborg *et al* 77]
- Cohesive zone model [Barenblatt, 62]
- Non-local approach [Pijaudier-Cabot and Bažant, 87]
- EFEM [Wells & Sluys, 00] / XFEM [Moës *et al*, 99]

Lattice truss model [Benkemoun *et al* 09]



Coupling with concrete civil engineering structures

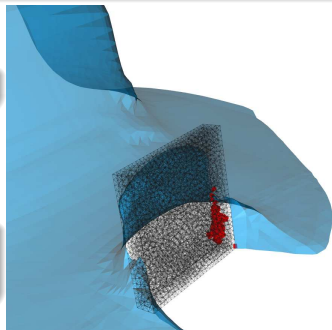
Localization limiters / Crack representation

- Smearred crack model [Hidelborg *et al* 77]
- Cohesive zone model [Barenblatt, 62]
- Non-local approach [Pijaudier-Cabot and Bažant, 87]
- EFEM [Wells & Sluys, 00] / XFEM [Moës *et al*, 99]

Lattice truss model [Benkemoun *et al* 09]

Crack opening \Rightarrow softening response

Force control: open question



Conclusions and Outlooks

Software implementation

cops component based implementation

- Flexible implementation
- Use of the middleware CTL
- Re-use existing code and libraries: FEAP, OpenFOAM
- Development of components: coFeap, ofoam, cops
- Parallel features for fluid subproblems (bottleneck) allows to reach 3D
- Transfer operation handled independently: Interpolator

Conclusions and Outlooks

Software implementation

cops component based implementation

- Flexible implementation
- Use of the middleware CTL
- Re-use existing code and libraries: FEAP, OpenFOAM
- Development of components: coFeap, ofoam, cops
- Parallel features for fluid subproblems (bottleneck) allows to reach 3D
- Transfer operation handled independently: Interpolator

Outlooks

- Transfer operator based on compact support radial basis functions
- Parallel features for the solid subproblem
- Coupling with other softwares (e.g. conuwata for wave propagation, other fluid and structure solvers)

Conclusions and Outlooks

Coupling algorithm for FSI

DFMT-BGS with Aitken's relaxation

- easy implementation and cheap computation outside existing codes
- coupling **incompressible** fluid and structure
- efficiency of Aitken's relaxation

Conclusions and Outlooks

Coupling algorithm for FSI

DFMT-BGS with Aitken's relaxation

- easy implementation and cheap computation outside existing codes
- coupling **incompressible** fluid and structure
- efficiency of Aitken's relaxation

Outlooks

- automatic choice for time step size
- decrease iteration number: better approximation of the tangent terms (still partitioned)
- expensive first iterations: model reduction

Conclusions and Outlooks

Models and discretization

Advantages of component technology and software re-use

- Popular FEM and FVM for fluid and structure part
- Efficient to use already developed models

Free surface flow computations

- VOF: selection of an appropriate model in `ofoam`
- Suitable for sloshing waves
- Full representation of the two-phase flow (water and air)

Conclusions and Outlooks

Models and discretization

Advantages of component technology and software re-use

- Popular FEM and FVM for fluid and structure part
- Efficient to use already developed models

Outlooks

- Fluid:
turbulence, non-newtonian flows, different representation (wave propagation and sloshing)
- Structure:
more advance models, multi-scale representation of the structure (MuSCAd), concrete structures
- Use FSI to model cement based material at small scales

