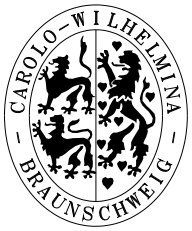
Strong Coupling Algorithms

Hermann G. Matthies

Institute of Scientific Computing
Technische Universität Braunschweig
Brunswick, Germany

acknowledgements to: J. Steindorf, M. Meyer, R. Niekamp, T. Srisupattarawanit

wire@tu-bs.de
http://www.wire.tu-bs.de



Fluid-Structure-Coupling (FSI)

Fluid—incompressible Newtonian fluid (i.e. Navier-Stokes eqns.) in ALE (**A**rbitrary **L**agrangean-**E**ulerian) formulation: (plus boundary conditions)

$$\text{in } \Omega_f: \quad \varrho_f \left(\dot{v} + (v - \dot{\chi}) \cdot \nabla v\right) - \operatorname{div} \sigma + \nabla p = r_f,$$

$$2\sigma = \nu(\nabla v + (\nabla v)^T) = 2\nu \nabla_s v, \qquad \operatorname{div} v = 0,$$

Solid—large deformation elastic St. Venant material in Lagrangean formulation: (plus boundary conditions)

$$\text{in } \Omega_s: \quad \varrho_s \ddot{u} - \operatorname{DIV}(FS) = r_s, \qquad F = I + \operatorname{GRAD} u$$

$$S = \lambda(\operatorname{tr} E) I + 2\mu E, \qquad 2E = (C - I), \qquad C = F^T F,$$

Arbitrary Lagrangean-Eulerian coordinate system:

$$\text{in } \Omega_f: \quad \mathcal{L}\chi = \beta_\Gamma u$$

FSI Interface

Conditions on interface Γ_I between Ω_f and Ω_s:

At spatial location $\chi(t) = \chi_0 + u(\chi_0, t) \in \Gamma_I$ continuity of velocities:

$$v(\chi(t), t) = \dot{u}(\chi_0, t).$$

Variational formulation for velocity condition:

$$\int_{\Gamma_I} \tau_I \cdot (v(\chi(t), t) - \dot{u}(\chi_0, t))\, d\Gamma_I = 0$$

Conservation of momentum—balance of tractions:

$$(\sigma - pI) \cdot n = -\frac{1}{J} F S F^T \cdot n\,, \qquad J = \det F\,.$$

Variational formulation for traction condition—treat like any other boundary traction, boundary traction is equal to Lagrange multiplier τ_I.

Coupled Problems

Coupled problems often combine the models of two or more physical systems, they are multi-physics modells. Different approaches:

- Monolithical approach, which means one global model of everything
  - Advantages: All encompassing theoretical and numerical treatment
  - Disadvantages: Treatment is ever more complex, every time completely new start, new algorithms, new software, does not scale, not modular

- Partitioned approach, which means separate models plus coupling
  - Advantages: Complexity constrained to one physical domain, theory may be well worked out, efficient numerical algorithms, existing sophisticated software for each subsystem, modular, scalable
  - Disadvantages: Subsystems have to be coupled together, new numerical and algorithmic problems, coupling software necessary

Pure Differential Coupling

The simplest case is pure differential coupling:

The first subsystem as evolution equation in some space $\mathcal{X}_1$:

$$\dot{x}_1 = f_1(x_1, x_2), \qquad x_1 \in \mathcal{X}_1 \ ,$$

The second subsystem as evolution equation in some space $\mathcal{X}_2$:

$$\dot{x}_2 = f_2(x_2, x_1), \qquad x_2 \in \mathcal{X}_2 \ ,$$

Combined nothing but a evolution equation for $(x_1, x_2) \in \mathcal{X}_1 \times \mathcal{X}_2$, direct identification of differential variables in both subsystems.

Might have been produced by the partition of monolithic system, or by combination of subsystems with identifiable variables

Pure Explicit Coupling

Assume that subsystems have been discretised in time (and in space if desired), assume for simplicity same time-step in both subsystems.

Approximation at time-step n denoted by $x_j^{(n)}$, $(j = 1, 2)$, with explicit or implicit time-discrete evolution φ_j, with functions $\Psi_j(n, t)$ to approximate evolution of variable x_j in $[t_n, t_{n+1}]$.

$$x_1^{(n+1)} = \varphi_1(x_1^{(n)}, \Psi_2(n, t)),$$
$$x_2^{(n+1)} = \varphi_2(x_2^{(n)}, \Psi_1(n, t)),$$

$\Psi_j(n, t)$ most easily produced by extrapolation of past values of $x_j^{(m)}$. Simplest is constant extrapolation—pure weak or loose coupling (switching):

$$\Psi_j(n, t) \equiv x_j^{(n)}, \quad (j = 1, 2)$$

Explicit Coupling—Switching

- Advantages:
  - Absolutely simple,
  - Can be performed in parallel.
- Disadvantages:
  - Critical time step will appear or may decrease,
  - In case of simple $\Psi_j(n,t)$ only first order accurate in Δt,
  - "Better" extrapolation $\Psi_j(n,t)$ for higher order decreases stability limit.

Resembles the block Jacobi or
additive Schwarz iteration for equation solution.

Explicit Coupling—Staggering

To achieve a better method—partly implicit—take as before

$$\Psi_2(n,t) \equiv x_2^{(n)} \text{ giving } x_1^{(n+1)},$$

but then—in a predictor-corrector fashion—to give the basic staggering method:

$$\Psi_1(n,t) = x_1^{(n+1)}.$$

This means subsystem 1 is solved as before,
but subsystem 2 gets response at new time level.

Resembles the block Gauss-Seidel or
multiplicative Schwarz iteration for equation solution.

Implicit Coupling

For stability reasons it may be advantageous to use implicit coupling,

$$x_1^{(n+1)} = \phi_1(x_1^{(n+1)}, x_1^{(n)}, \Psi_2) \,,$$

$$x_2^{(n+1)} = \phi_2(x_2^{(n+1)}, x_2^{(n)}, \Psi_1) \,,$$

$\Psi_j(n,t)$ including still unknown $x_j^{(n+1)}$ —strong or tight coupling—
simplest case purely constant extrapolation:

$$\Psi_j(n,t) \equiv x_j^{(n+1)}, \ (j = 1, 2),$$

requires global iteration—simplest case as before in time-stepping: in Jacobi or additive Schwarz fashion, or in Gauss-Seidel or multiplicative Schwarz fashion.

- Advantages: May be globally unconditionally stable, may be higher order in Δt without compromising stability, same results as monolithical approach.

- Disadvantages: Requires global iteration, but will converge for small Δt.

Differential-Algebraic Coupling

Assume that subsystems are differential-algebraic equations (DAEs) with local differential variables $x_1 \in \mathcal{X}_1$, local algebraic variables $y_1 \in \mathcal{Y}_1$:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, y_1, z) \,, \\ 0 &= g_1(x_1, x_2, y_1, z) \,, \end{aligned}$$

same for the second subsystem with local differential variables $x_2 \in \mathcal{X}_2$, local algebraic variables $y_2 \in \mathcal{Y}_2$:

$$\begin{aligned} \dot{x}_2 &= f_2(x_2, y_2, z) \,, \\ 0 &= g_2(x_2, x_1, y_2, z) \,. \end{aligned}$$

Coupling conditions formulated as "algebraic" constraints with global algebraic variables $z \in \mathcal{Z}$:

$$0 = h(x_1, x_2, y_1, y_2, z) \,.$$

Differential-Algebraic Regularity

Assume that each single subsystem, and also global system is an index-1 DAE. This means that the operator matrices

$$D_{y_j} g_j \ , \qquad \begin{bmatrix} D_{y_j} g_j & D_z g_j \\ D_{y_j} h & D_z h \end{bmatrix}, \ (j = 1, 2) \ , \qquad \begin{bmatrix} D_y g & D_z g \\ D_y h & D_z h \end{bmatrix}$$

have to be regular, where D_q is the partial derivative w.r.t. q, and we have set $g = (g_1, g_2)^T$ and $y = (y_1, y_2)^T$.

After time discretisation—we time-discretise DAEs with an implicit method—we have a global system of equations:

$$\begin{aligned} (x_1^{(n+1)}, y_1^{(n+1)})^T &= \Phi_1(x_1^{(n+1)}, x_1^{(n)}, y_1^{(n+1)}, y_1^{(n)}, z^{(n+1)}, z^{(n)}) \ , \\ (x_2^{(n+1)}, y_2^{(n+1)})^T &= \Phi_2(x_2^{(n+1)}, x_2^{(n)}, y_2^{(n+1)}, y_2^{(n)}, z^{(n+1)}, z^{(n)}) \ , \\ 0 &= h(x_1^{(n+1)}, x_2^{(n+1)}, y_1^{(n+1)}, y_2^{(n+1)}, z^{(n+1)}) \end{aligned}$$

Discrete Form of Fluid-Structure-Interaction

The Fluid and the ALE-Domain:

$$\begin{aligned} \boldsymbol{M}_f\dot{v} + \boldsymbol{N}(v-\dot{\chi})v + \boldsymbol{K}_f v + \boldsymbol{B}_f p &= r_f + \boldsymbol{T}_f^T \tau_I;, \\ \boldsymbol{B}_f^T v &= 0\,, \\ \boldsymbol{K}_g \chi &= \boldsymbol{A}u\,. \end{aligned}$$

The Solid:

$$\boldsymbol{M}_s\ddot{u} + \boldsymbol{K}_s(u)u = r_s - \boldsymbol{T}_s^T\tau_I\,.$$

The Interface:

$$\boldsymbol{T}_f v = \boldsymbol{T}_s\dot{u}\,.$$

Equality of interface tractions on coupling interface already included in terms $\boldsymbol{T}_f^T\tau_I$ and $\boldsymbol{T}_s^T\tau_I$.

The DAE Correspondence I

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, y_1, z) \,, \\ 0 &= g_1(x_1, x_2, y_1, z) \,, \end{aligned}$$

The Fluid and the ALE-Domain: (with $b := \dot{v}$, the fluid accel.) and $\psi = \dot{\chi}$:

$$x_1 := \begin{bmatrix} v \\ \chi \end{bmatrix} \,, \quad y_1 := \begin{bmatrix} b \\ p \\ \psi \end{bmatrix} \,, \quad f_1 := \begin{bmatrix} b \\ \psi \end{bmatrix} \,, \quad z := \tau_I \,,$$

$$g_1 := \begin{bmatrix} \boldsymbol{M}_f b + \boldsymbol{N}_f(v-\psi)v + \boldsymbol{K}_f v + \boldsymbol{B}_f p - r_f - \boldsymbol{T}_f^T \tau_I \\ -\boldsymbol{B}_f^T \boldsymbol{M}_f^{-1}(-\boldsymbol{N}_f(v-\psi)v - \boldsymbol{B}_f p - \boldsymbol{K}_f v + r_f + \boldsymbol{T}_f^T \tau_I) \\ \boldsymbol{K}_g \psi - \boldsymbol{A} w \end{bmatrix} \,,$$

The DAE Correspondence II

$$\begin{aligned} \dot{x}_2 &= f_2(x_2, y_2, z) \,, \\ 0 &= g_2(x_2, x_1, y_2, z) \,. \end{aligned}$$

The Solid: (with $a := \dot{w} = \ddot{u}$, the structural acceleration):

$$\begin{aligned} x_2 &:= \begin{bmatrix} u \\ w \end{bmatrix} \,, \quad y_2 := a \,, \quad f_2 := \begin{bmatrix} w \\ a \end{bmatrix} \,, \quad z := \tau_I \\ g_2 &:= \boldsymbol{M}_s a + \boldsymbol{K}_s(u)u - r_s + \boldsymbol{T}_s^T \tau_I; \,, \end{aligned}$$

$$0 = h(x_1, x_2, y_1, y_2, z) \,.$$

The Interface: $h := \boldsymbol{T}_f b - \boldsymbol{T}_s a$.

Equivalence of Iteration and Time-Stepping

Discretised dynamical system with state $\boldsymbol{x}^{(n)}$ at time levels $n \cdot \Delta t$
with the time advance operator $\boldsymbol{x}^{(n+1)} = \boldsymbol{\varphi}(\boldsymbol{x}^{(n)})$.

A stationary state or equilibrium $\boldsymbol{x}^*$ is fixed point for $\boldsymbol{x}^* = \boldsymbol{\varphi}(\boldsymbol{x}^*)$,
it is asymptot. stable ($\boldsymbol{x}^{(n)} \overset{n\to\infty}{\longrightarrow} \boldsymbol{x}^*$) iff $|\lambda_j| < 1$ for all eigenvalues of $D\boldsymbol{\varphi}(\boldsymbol{x}^*)$.

This is actually a method to compute the steady state or equilibrium,
often used in CFD (sometimes called dynamic relaxation).

An iteration method is an abstract dynamical system,
each iteration $\boldsymbol{x}_k + 1 = \boldsymbol{\psi}(\boldsymbol{x}_k)$ corresponds to one time step.
Iteration converges iff $|\lambda_j| < 1$ for all eigenvalues of $D\boldsymbol{\varphi}(\boldsymbol{x}^*)$.

Equivalent with stability of time-stepping scheme for given Δt.

Time-Stepping and Iteration

Solution process (iteration) for $\boldsymbol{x}^{(n+1)} = \boldsymbol{\phi}(\boldsymbol{x}^{(n+1)}, \boldsymbol{x}^{(n)})$ is a discrete dynamical system $\boldsymbol{x}_{\kappa}^{(n+1)} := \boldsymbol{\psi}(\boldsymbol{x}_{\kappa-1}^{(n+1)})$.

Will converge if $|\lambda_j| < 1$ for all eigenvalues of $D\boldsymbol{\psi}(\boldsymbol{x}^{(n+1)})$
$\Rightarrow \boldsymbol{\psi}$ is a contraction.

Convergence of this iteration and stability of time-stepping may be investigated with same theory.

Weak coupling/ simple switching corresponds (in simple case is equal to) strong coupling/ block-Jacobi iteration.

Weak coupling/ basic staggering corresponds (in simple case is equal to) strong coupling/ block-Gauss-Seidel iteration.

Global Equations for Strong Coupling

Coupling condition $h = 0$ usually adjoined to one subsystem.
Set $\boldsymbol{\xi} := (x_1, y_1, z)^T = (v, \chi, b, p, \psi, \tau_I)^T$
and $\boldsymbol{\zeta} := (x_2, y_2)^T = (u, w, a)^T$ to include interface in first equation,
otherwise include $z = \tau_I$ in $\boldsymbol{\zeta}$ and not in $\boldsymbol{\xi}$.

Assume that convergent iterative solvers for subsystems exist:

$$\boldsymbol{\xi}_\kappa = \boldsymbol{F}_1(\boldsymbol{\xi}_{\kappa-1}, \boldsymbol{\zeta}) \,, \quad \text{and} \quad \boldsymbol{\zeta}_\kappa = \boldsymbol{F}_2(\boldsymbol{\zeta}_{\kappa-1}, \boldsymbol{\xi}) \,, \quad \kappa = 1, 2, \ldots \,;$$

Simplest solution process is nonlinear block-Jacobi,
an additive or parallel Schwarz procedure (corresponds to simple switching):

$$\begin{aligned} \boldsymbol{\xi}_\kappa &= \boldsymbol{F}_1^{\nu_1}(\boldsymbol{\xi}_{\kappa-1}, \boldsymbol{\zeta}_{\kappa-1}) \,, \\ \boldsymbol{\zeta}_\kappa &= \boldsymbol{F}_2^{\nu_2}(\boldsymbol{\zeta}_{\kappa-1}, \boldsymbol{\xi}_{\kappa-1}) \,; \end{aligned}$$

Nonlinear block-Gauss-Seidel

Almost as simple is nonlinear block-Gauss-Seidel,
a multiplicative or serial Schwarz procedure (corresponds to basic staggering):

$$\boldsymbol{\xi}_\kappa = \boldsymbol{F}_1^{\nu_1}(\boldsymbol{\xi}_{\kappa-1}, \boldsymbol{\zeta}_{\kappa-1}) \,, \quad \text{and with new } \boldsymbol{\xi}_\kappa, \text{ do } \boldsymbol{\zeta}_\kappa = \boldsymbol{F}_2^{\nu_2}(\boldsymbol{\zeta}_{\kappa-1}, \boldsymbol{\xi}_\kappa) \,.$$

Theorem:[Arnold, Günther] *In block-G-S, let L be Lipschitz-constant of Ψ_j, and let*

$$\alpha = \max_{t\in[0,T]} \| \left(D_{y_2}\boldsymbol{g}_2\right)^{-1} D_z\boldsymbol{g}_2 \left(D_{y_1}\boldsymbol{h} \left(D_{y_1}\boldsymbol{g}_1\right)^{-1} D_z\boldsymbol{g}_1 \right)^{-1} D_{y_2}\boldsymbol{h} \| \,,$$

the iteration only converges if $\alpha < 1$, and if at least κ iterations are performed so that $L\alpha^\kappa < 1$, and the total global time-step error δ is bounded by

$$\delta < C(\mu^{\max\{0,\kappa-2\}}\psi(\boldsymbol{x}) + \mu^{\kappa-1}\psi(\boldsymbol{y})) + \varepsilon_1(\boldsymbol{x}) + \varepsilon_2(\boldsymbol{y}),$$

ψ is extrapolation error, ε_j is subsystem integrator error, and
$\mu = \alpha + O(\Delta t) < 1$.

Eliminating one Variable

One may see block-G-S in the following way:

$$\boldsymbol{F}_1^{\nu_1} : \boldsymbol{\zeta}_{\kappa-1} \longrightarrow \boldsymbol{\xi}_\kappa,$$

followed by: ($\boldsymbol{\xi}$ becomes "internal")

$$\boldsymbol{F}_2^{\nu_2} : \boldsymbol{\xi}_\kappa \longmapsto \boldsymbol{\zeta}_\kappa.$$

In toto, there is a mapping on $\boldsymbol{\zeta}$ alone:

$$\boldsymbol{S} : \boldsymbol{\zeta}_{\kappa-1} \longrightarrow \boldsymbol{\zeta}_\kappa$$

$\boldsymbol{\zeta}$ may be just the variables on interface.
Fixed-point of the map $\boldsymbol{S}$ is part of the solution
The fixed-point equation may be solved by some other method
(e.g. Newton-Raphson, preconditioned/ modified Newton, Quasi-Newton, etc.)

Different Possibilities for block-Gauss-Seidel

For different ordering and distribution of constraint, we have

- 1st fluid plus coupling, 2nd solid: $\alpha = \|\boldsymbol{M}_s^{-1}\boldsymbol{T}_s^T(\boldsymbol{T}_f\widehat{\boldsymbol{M}}^{-1}\boldsymbol{T}_f^T)^{-1}\boldsymbol{T}_s\|$,,
  where $\widehat{\boldsymbol{M}}^{-1} = \boldsymbol{M}_f^{-1}(\boldsymbol{M}_f - \boldsymbol{B}_f\widetilde{\boldsymbol{M}}_p\boldsymbol{B}_f^T)\boldsymbol{M}_f^{-1}$ is a Schur complement, and $\widetilde{\boldsymbol{M}}_p = (\boldsymbol{B}_f^T\boldsymbol{M}_f^{-1}\boldsymbol{B}_f)^{-1}$. Note $\alpha \propto \varrho_f/\varrho_s$.
- 1st structure plus coupling, 2nd fluid: $\alpha = \|\widehat{\boldsymbol{M}}^{-1}\boldsymbol{T}_f^T(\boldsymbol{T}_s\boldsymbol{M}_s^{-1}\boldsymbol{T}_s^T)^{-1}\boldsymbol{T}_f\|$.
  Note $\alpha \propto \varrho_s/\varrho_f$.
- 1st fluid, 2nd solid plus coupling: $\alpha = \|(\boldsymbol{T}_s\boldsymbol{M}_s^{-1}\boldsymbol{T}_s^T)^{-1}(\boldsymbol{T}_f\widehat{\boldsymbol{M}}^{-1}\boldsymbol{T}_f^T)\|$.
  Note $\alpha \propto \varrho_s/\varrho_f$.
- 1st solid plus coupling, 2nd fluid: $\alpha = \|(\boldsymbol{T}_f\widehat{\boldsymbol{M}}^{-1}\boldsymbol{T}_f^T)^{-1}(\boldsymbol{T}_s\boldsymbol{M}_s^{-1}\boldsymbol{T}_s^T)\|$.
  Note $\alpha \propto \varrho_f/\varrho_s$.

α depends on ratio of ϱ_f and ϱ_s.

Block-Newton

Desirable is an iteration scheme which will not depend on ordering and distribution of constraint: Block-Newton.

In each block-Newton iteration following system has to be solved:

$$\begin{bmatrix} \boldsymbol{I} - D_\xi \boldsymbol{F}_1 & D_\zeta \boldsymbol{F}_1 \\ D_\xi \boldsymbol{F}_2 & \boldsymbol{I} - D_\zeta \boldsymbol{F}_2 \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{\xi}_\kappa \\ \Delta \boldsymbol{\zeta}_\kappa \end{bmatrix} = - \begin{bmatrix} \boldsymbol{\xi}_\kappa - \boldsymbol{F}_1(\boldsymbol{\xi}_\kappa, \boldsymbol{\zeta}_\kappa) \\ \boldsymbol{\zeta}_\kappa - \boldsymbol{F}_2(\boldsymbol{\zeta}_\kappa, \boldsymbol{\xi}_\kappa) \end{bmatrix}.$$

Symbolic block-Gauss elimination:

$$\Delta \boldsymbol{\xi} = -(\boldsymbol{I} - D_\xi \boldsymbol{F}_1)^{-1}(\boldsymbol{\xi} - \boldsymbol{F}_1(\boldsymbol{\xi}, \boldsymbol{\zeta})) - \boldsymbol{C} \Delta \boldsymbol{\zeta} \,,$$

with the multiplier matrix $\boldsymbol{C} := (\boldsymbol{I} - D_\xi \boldsymbol{F}_1)^{-1}[D_\zeta \boldsymbol{F}_1]$.

Further with Schur complement matrix $\boldsymbol{S}$:
$\boldsymbol{S}\Delta\boldsymbol{\zeta} := (\boldsymbol{I} - [D_\zeta \boldsymbol{F}_2] - [D_\xi \boldsymbol{F}_2]\boldsymbol{C})\, \Delta\boldsymbol{\zeta} = -\boldsymbol{r}$,
with $\boldsymbol{r} := (\boldsymbol{\zeta} - \boldsymbol{F}_2(\boldsymbol{\zeta}, \boldsymbol{\xi})) + [D_\xi \boldsymbol{F}_2]\boldsymbol{q}$, $\boldsymbol{q} := -(\boldsymbol{I} - D_\xi \boldsymbol{F}_1)^{-1}(\boldsymbol{\xi} - \boldsymbol{F}_1(\boldsymbol{\xi}, \boldsymbol{\zeta}))$.

Solving the Block-Newton System

Solution proceeds by Krylov method (Bi-CGstab):

- Solving a system with $(\boldsymbol{I} - D_\xi \boldsymbol{F}_1)$: Apply iterative solver $\boldsymbol{F}_1$
- Same with $\boldsymbol{C}$, plus finite differences for $[D_\zeta \boldsymbol{F}_1]$
- Solving the Schur-complement system:
  - Use Bi-CGstab.
  - Compute $\boldsymbol{r}$ with iterating subsystem solver $\boldsymbol{F}_2$;
  - compute action of $\boldsymbol{S}$ by finite differences.

Theorem:[Mackens, Voss] *If the single system solvers are quadratically convergent (or enough iterations are made in the approximative steps), the global iteration is also quadratically convergent.*

Quasi-Newton

Quasi-Newton methods are generalisations of secant method:

$$\boldsymbol{H}_\kappa \begin{bmatrix} \Delta\boldsymbol{\xi}_\kappa \\ \Delta\boldsymbol{\zeta}_\kappa \end{bmatrix} = - \begin{bmatrix} \boldsymbol{\xi}_\kappa - \boldsymbol{F}_1(\boldsymbol{\xi}_\kappa, \boldsymbol{\zeta}_\kappa) \\ \boldsymbol{\zeta}_\kappa - \boldsymbol{F}_2(\boldsymbol{\zeta}_\kappa, \boldsymbol{\xi}_\kappa) \end{bmatrix}.$$

Easy to solve with $\boldsymbol{H}_\kappa$ (explicit inverse $\boldsymbol{H}_\kappa^{-1}$).
$\boldsymbol{H}_\kappa$ changes by low rank only from step to step.

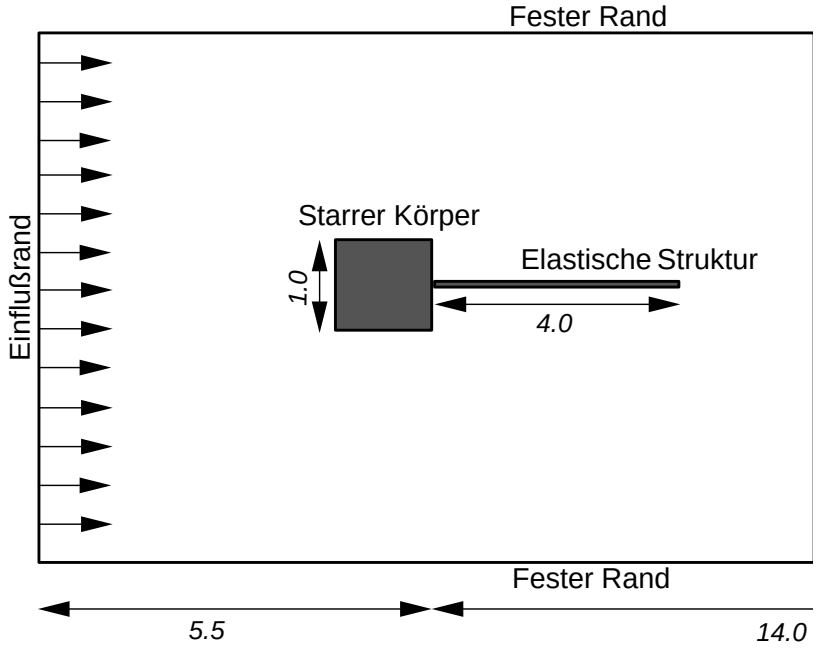
$$\boldsymbol{H}_\kappa^{-1} = \boldsymbol{H}_{\kappa-1}^{-1} + \boldsymbol{a}_\kappa \cdot \boldsymbol{b}_\kappa^T$$

a rank one update—or—a rank two update

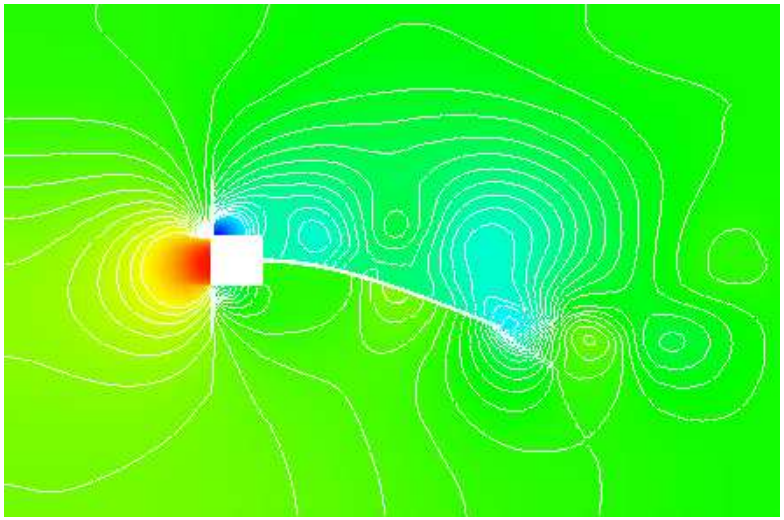
$$\boldsymbol{H}_\kappa^{-1} = \boldsymbol{H}_{\kappa-1}^{-1} + \boldsymbol{a}_\kappa \cdot \boldsymbol{a}_\kappa^T + + \boldsymbol{b}_\kappa \cdot \boldsymbol{b}_\kappa^T$$

$\boldsymbol{a}_\kappa, \boldsymbol{b}_\kappa$ are easy to compute from known data

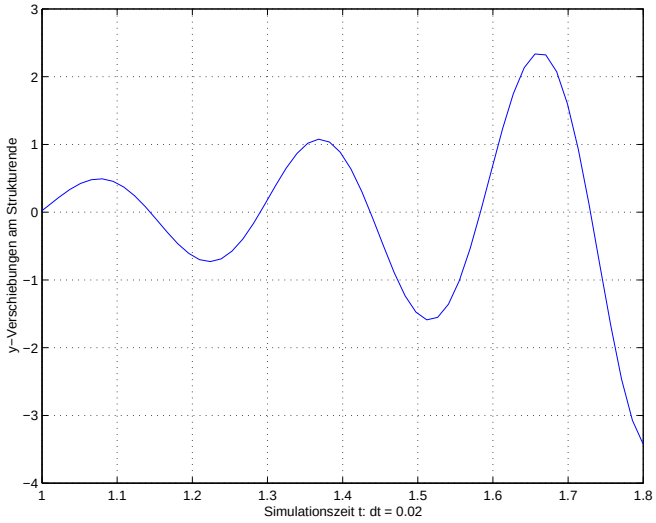
A Simple Example



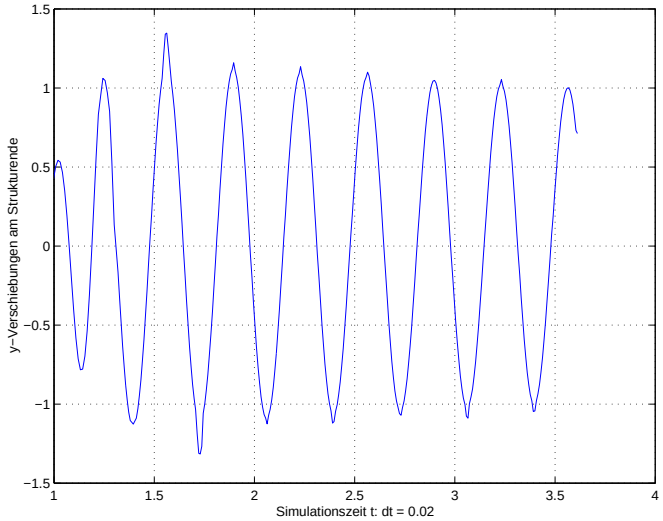
Movement and Pressure Distribution



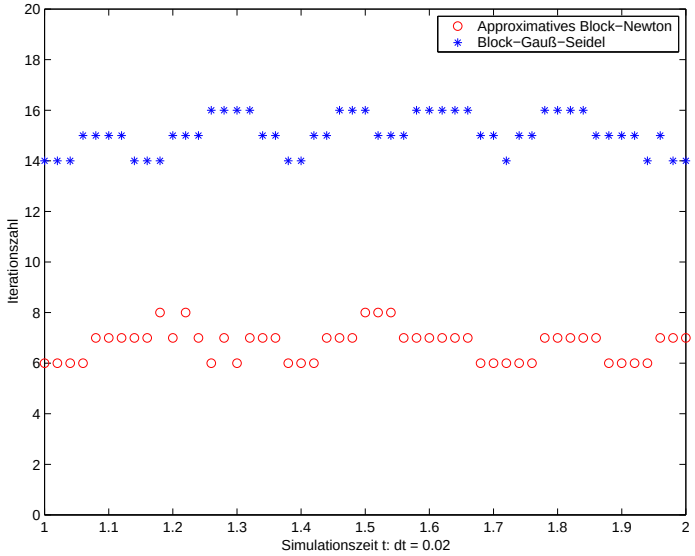
Tip Displacement Response Weak Coupling



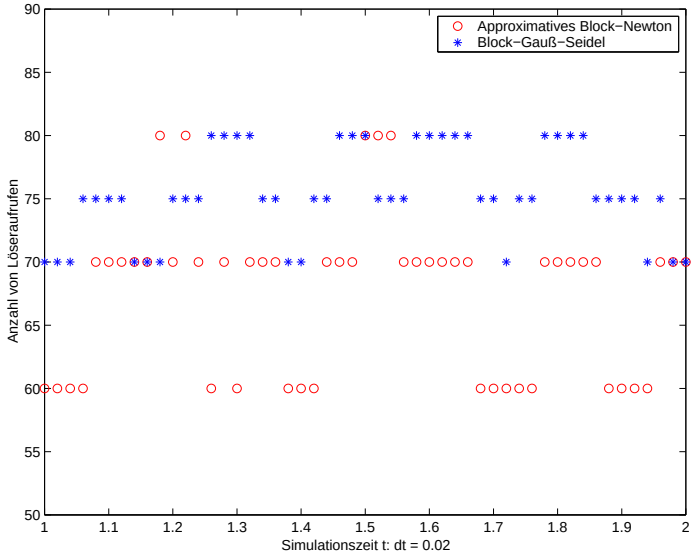
Tip Displacement Response Strong Coupling



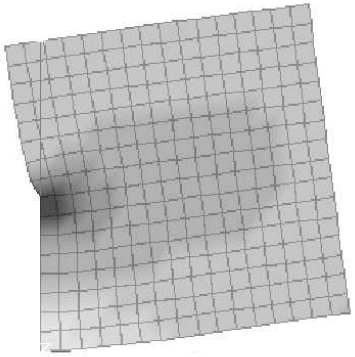
Iteration Count

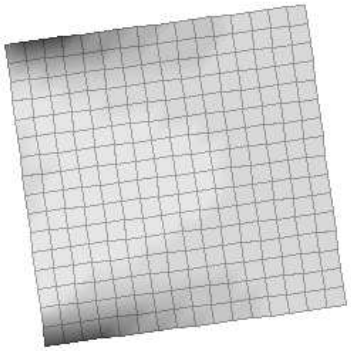


Solver Calls



Another Example





Material St.Venant: $\eta = 0.2, E = E_l$ Navier-Lamé: $\eta = 0.2, E = E_r$

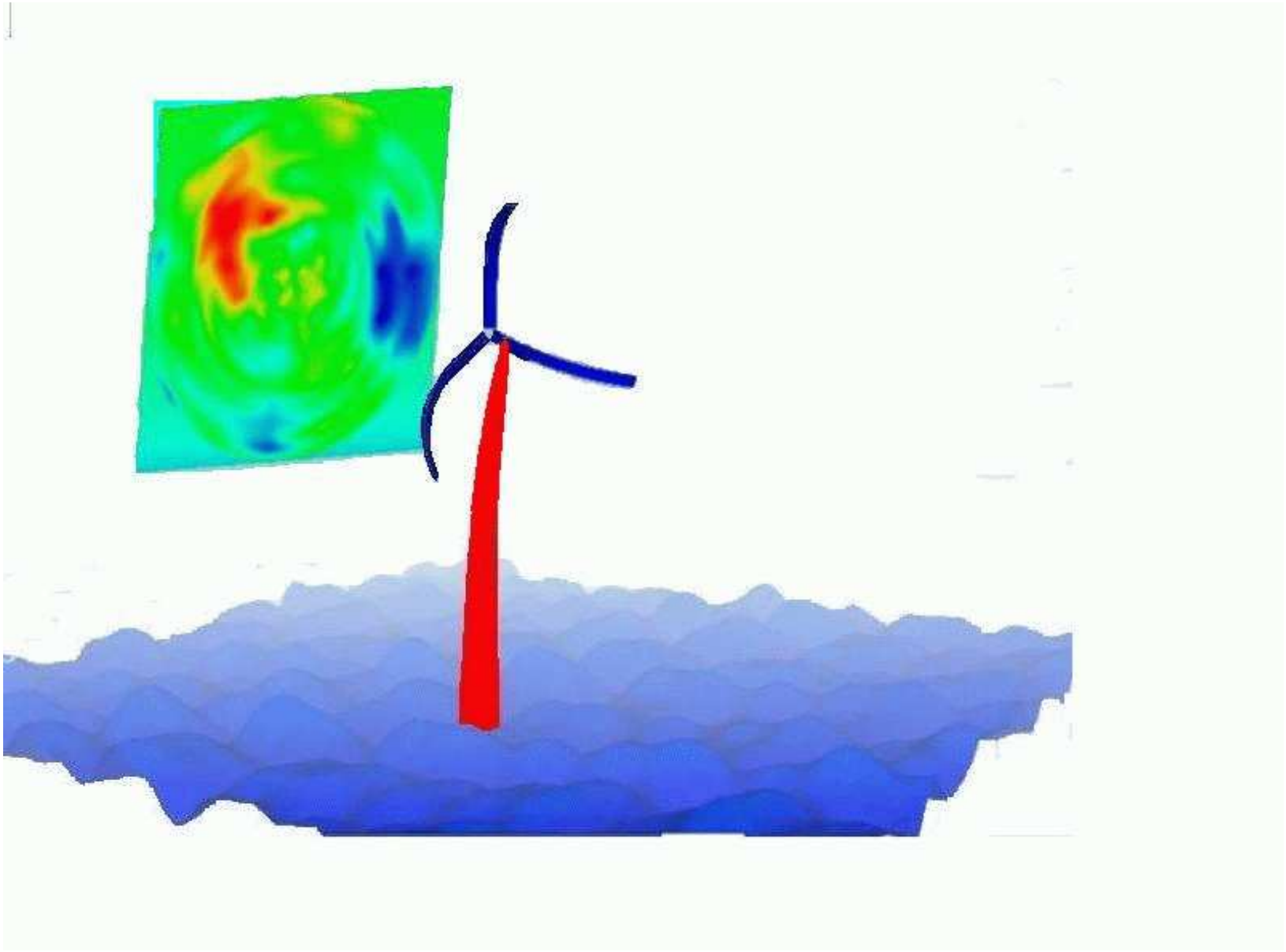
Iteration Count

$E_l = 10^5, E_r = 5 * 10^4$		
	iter	cpu[t]
Jacobi	18	10.9
Gauss-Seidel	14	7.9
BFGS	12	6.8
Newton	3	13.5

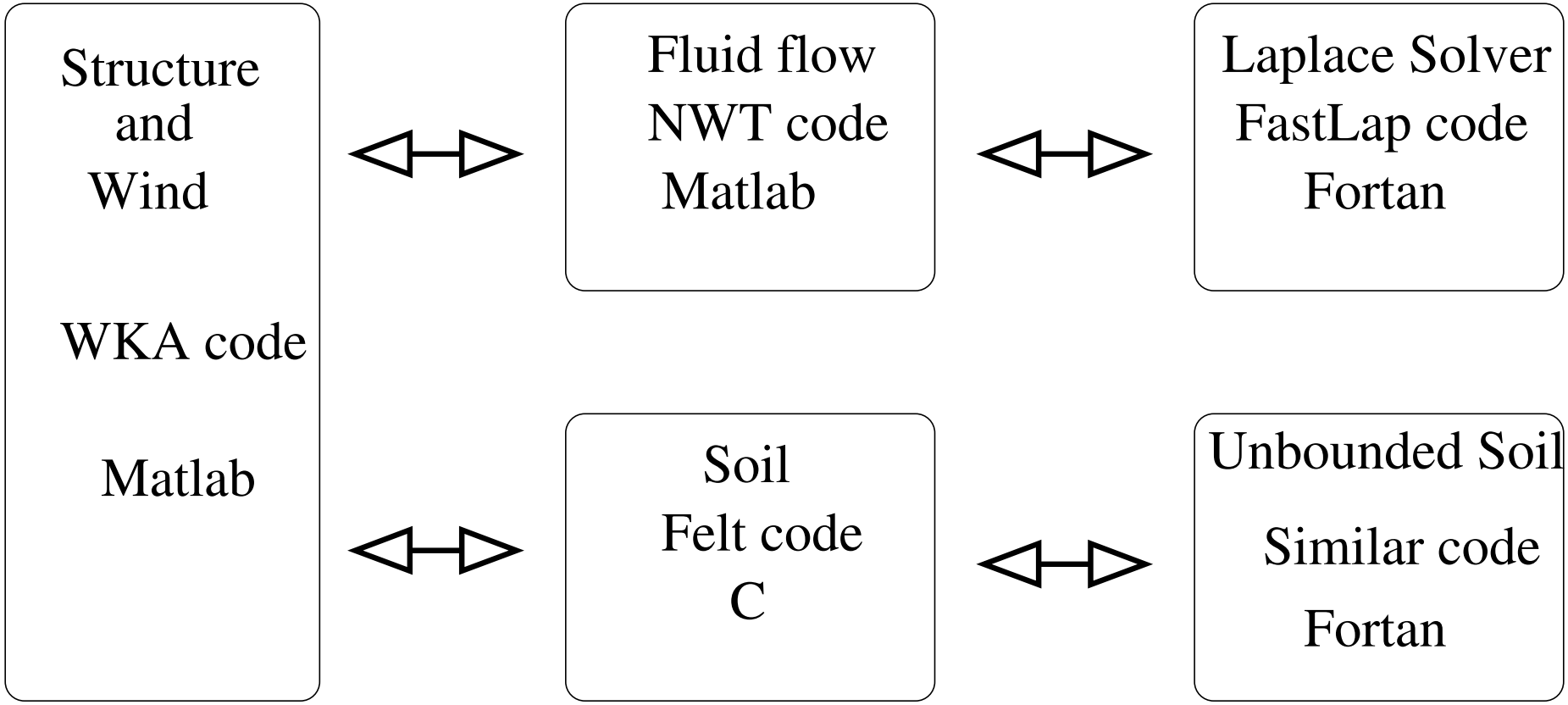
$E_l = 10^5, E_r = 10^6$		
	iter	cpu[t]
Jacobi	∞	-
GS	28	16.0
BFGS	8	4.8
Newton	3	13.7

$E_l = 5 * 10^4, E_r = 10^6$		
	iter	cpu[t]
Jacobi	∞	-
GS	∞	-
BFGS	25	13.4
Newton	12	52.0

A More Involved Example—OWECS



Some of the Subsystems



Subsystems of Offshore-Wind-Turbine I

Wind Stochastic wind model in frequency domain.

Wind/ Blade-Structure Blade element theory for the turbine blades.
Simple force and torque balance in decoupled rotor disk annuli.

Blade/ Nacelle & Tower Manages non-inertial rotating co-ordinate systems, one for each blade, one each for nacelle, tower. Structure-structure coupling. (Plus generator, control, etc.).

Random Waves/ Fluid-Fluid Coupling To prevent parasitic reflected waves and get the right random properties $\Rightarrow$ three wave domains.
Deep water, only linear incident waves—shallower water, nonlinear incident and linearised refracte waves—shallow water, fully nonlinear waves and FSI.

Subsystems of Offshore-Wind-Turbine II

Deep Water Linear Airy waves for deep water in spectral description.

Shallower Water Nonlinear incident wave, linear perturbation refracted wave. Potential flow. FE-model for free surface, fast multipole BEM solver for water below.

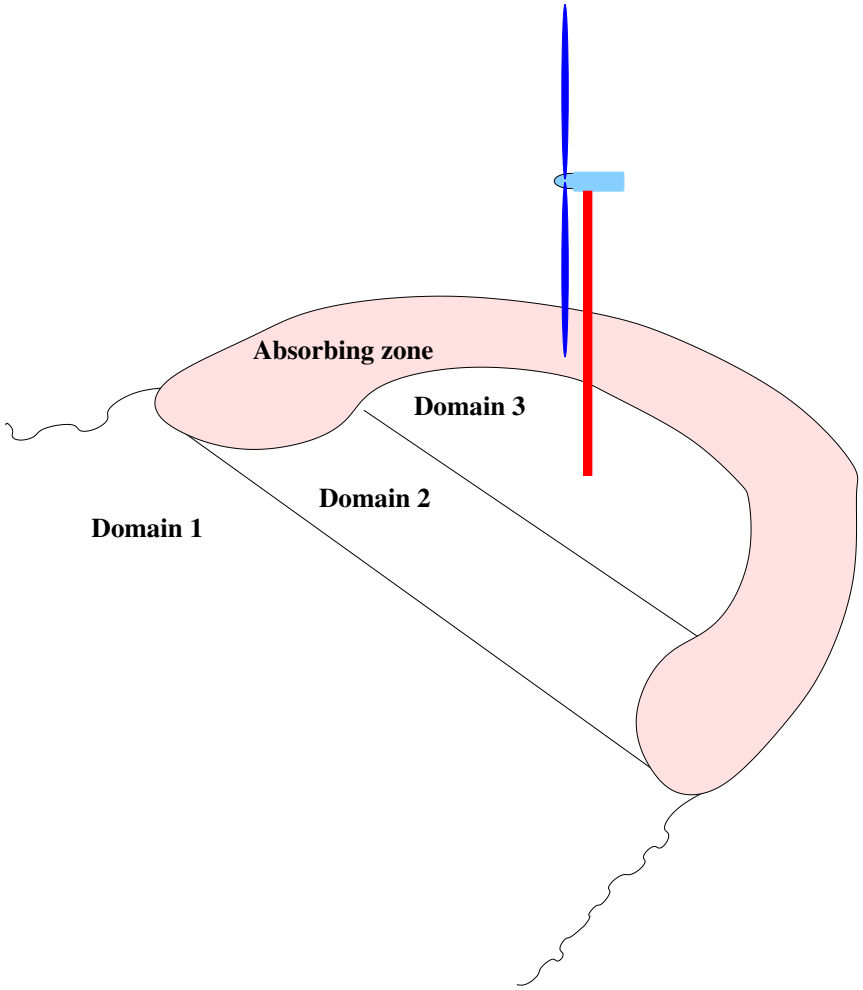
Shallow Water Fully nonlinear wave, both for incident and refracted wave. Potential flow. FE-model for free surface, fast multipole BEM solver for water below.

Wave/ Tower Potential Flow—Free Surface/ Tower Coupling.

Tower/ Soil Coupling tower/ pile FE-model with a near field FE-model of the soil.

Soil/ Soil Near field FE-model of soil coupled with far-field SBFEM code.

Wave Domains



Some of the OWECS Models

Structure description is standard large displacement beam model, in non-inertial (co-rotational) reference systems. Soil is linear elastic solid.

Blade element theory is simple mass flow—momentum and rotational momentum , and mass conservation in each rotor annulus independently, together with measured / previously calculated profile data (C_L, C_D, C_M).

Waves—Potential flow with free surface:
Fluid velocity $\boldsymbol{v}(\boldsymbol{x}, t) = \nabla \Phi(\boldsymbol{x}, t)$ with potential Φ, and for all t: $\nabla^2 \Phi = 0$.

Free surface described by level set function $F(\boldsymbol{x}, t) = 0$.
Eikonal equation for free surface $\partial_t F / |\nabla F|$ movement has to match normal velocity of fluid $\boldsymbol{v} \cdot \boldsymbol{n}$ with $\boldsymbol{n} = -\nabla F / |\nabla F|$:

$$\boldsymbol{v} \cdot \boldsymbol{n} = -\nabla \Phi \cdot \frac{\nabla F}{|\nabla F|} \stackrel{!}{=} \frac{\partial_t F}{|\nabla F|} \quad \Leftrightarrow \quad \frac{DF}{dt} := \partial_t F + \boldsymbol{v} \cdot \nabla F = 0.$$

Conclusions

- staggering algorithms may introduce critical time step
- coupling may introduce algebraic constraints
- DAEs are different from pure differential coupling
- block-Gauss-Seidel depends strongly on ordering
  - in purely differential case may be made convergent with small Δt
  - in DAE case may be unconditionally unstable
- Newton-like methods are more robust (and may be faster)