Solving large scale flow acoustics time-harmonic problems in a HPC framework using domain decomposition

Philippe Marchner

Siemens Digital Industries Software
philippe.marchner@siemens.com

Rencontre JCJC ondes
November 29th, 2022

SIEMENS
Ingenuity for life

Outline

Outline

Outline

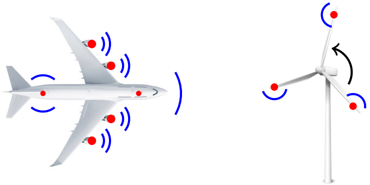
1. Introduction
   - Industrial context
   - Flow acoustics
   - Applications
2. Iterative domain decomposition approach
   - Non-overlapping Schwarz method
   - Building transmission conditions in the presence of flow
   - Domain decomposition in a straight waveguide
   - Variational formulation
   - Towards advanced test cases
3. Software implementation in a HPC framework
   - Implementation framework
   - Scalability assessment
   - The 3D turbofan engine benchmark
   - Reaching high frequency
4. Limitations and extensions

Industrial context

Long term perspective

Provide a cheap flow acoustics solver: noise from bodies in motion

Computational workflow



1. compute mean flow (e.g. RANS)
2. extract acoustic sources
3. compute sound propagation
4. find solutions (new material)

Objective

Provide a "*ready-to-use*" sound propagation simulation tool

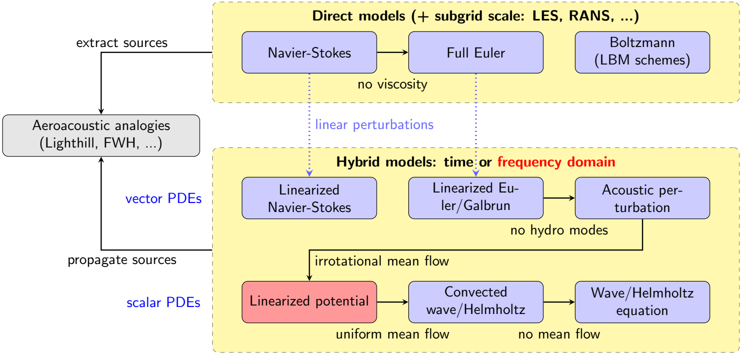
- suitable to modern computer architectures
- applicable to large, complex industrial problems

→ to be used in optimization

Outline

Physical models for flow acoustics

We focus on the time-harmonic regime (rotating machines)



Hybrid model - solve mean flow and acoustic perturbations separately

- Linearized Euler Equations provide a precise physical model, but are costly and exhibit hydrodynamic instabilities
- We rather focus on a self-adjoint, scalar operator : Linearized Potential/Pierce Operator [*Spieser, Bailly 2020*]

A self-adjoint flow acoustic operator

PDE for the acoustic velocity potential $\boldsymbol{v} = \nabla u$ and compact source f

Linearized Potential Equation (LPE)

$$\rho_0(\boldsymbol{x}) \frac{\mathrm{D}_0}{\mathrm{D}t} \left(\frac{1}{c_0(\boldsymbol{x})^2} \frac{\mathrm{D}_0 u}{\mathrm{D}t} \right) - \nabla \cdot (\rho_0(\boldsymbol{x}) \nabla u) = f, \quad \frac{\mathrm{D}_0}{\mathrm{D}t} = \mathrm{i}\omega + \boldsymbol{v}_0(\boldsymbol{x}) \cdot \nabla$$

Helmholtz-type problem with convection and heterogeneities

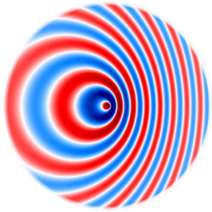
Mathematical difficulties

- oscillatory, non-local solution
- complex valued, strongly indefinite with ω
- unbounded domain
- local convection effects

Does not converge with classical iterative methods *[Ernst, Gander 2012]*

$\rightarrow$ **direct solver**

Point source in a uniform flow



$\mathrm{M} = \|\boldsymbol{v}_0\| / c_0 = 0.6$

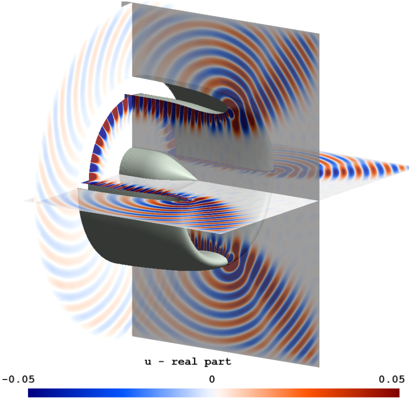
$\mathrm{M} < 1$ (Subsonic flow)

Outline

3D example: acoustic radiation of a turbofan engine

A typical problem: Compute the tonal radiation of an engine intake
Current solver: direct solver (MUMPS) + p-FEM approximation

$\omega_{\text{bpf}} \leftrightarrow \approx 25$ wavelengths



ω_{bpf}, $N_{\text{dofs}} = 10\text{M}$, $\text{nnz} = 730\text{M}$
Direct solver $\rightarrow$ 740 Gb of RAM

$\Downarrow$ increase ω ?

$2 \times \omega_{\text{bpf}}$, $N_{\text{dofs}} = 73\text{M}$, $\text{nnz} = 5\text{B}$
Direct solver $\approx$ 6 Tb of RAM ...

$\mathcal{O}(\omega^3)$ scaling in memory & time ...

Goal: compute tones up to 5 bpf !

Turbofan exhaust radiation

Turbofan exhaust: fan noise through dual-stream jet flow
Mean flow obtained through RANS computation $(\rho_0, c_0, \mathbf{v}_0)$

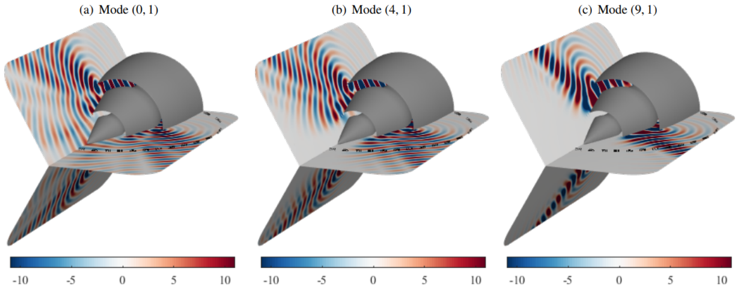


Figure: Real part of the acoustic pressure at 7497 Hz for various incident modes, from [*Hamiche et al. 2019*]

Memory limitation from $\approx$ 20-25 wavelengths in 3D

Objective

Industrial objective

Provide a (scalable) parallel solver to increase the upper frequency limit

Available tools at Siemens

Discretization

- high-order finite elements
  $\rightarrow$ reduce discretization error (interpolation & dispersion)
- *a-priori* error indicator - adaptive order [*Bériot et al. 2016*]
- efficient frequency sweep

Parallelization

- algebraic parallelization is hard for Helmholtz problems
- instead, "divide and conquer" at the continuous (PDE) level
  $\rightarrow$ domain decomposition
- lots of approaches, but common framework [*Gander, Zhang 2019*]

Selected solution

Extend the non-overlapping Schwarz domain decomposition framework [*Boubendir et al. 2012*]

Outline

Outline

Non-overlapping Schwarz method

Partition $\Omega = \bigcup_{i=1}^{N_{\text{dom}}} \Omega_i$ into subdomains, and solve the BVPs

Non-overlapping optimal Schwarz formulation

$$\begin{cases} \rho_0 \frac{D_0}{Dt}\left(\frac{1}{c_0^2}\frac{D_0 u_i}{Dt}\right) - \nabla \cdot (\rho_0 \nabla u_i) = 0 \text{ in } \Omega_i \text{ (volume PDE)} \\ \rho_0 \left(1 - M_{\boldsymbol{n}}^2\right) \left(\partial_{\boldsymbol{n}_i} u_i + \imath \widetilde{\Lambda}^+ u_i\right) = 0, \text{ on } \Gamma_i^\infty \text{ (radiation condition)} \\ \rho_0 \left(1 - M_{\boldsymbol{n}}^2\right) \left(\partial_{\boldsymbol{n}_i} u_i + \imath \mathcal{S}_i u_i\right) = g_{ij}, \text{ on } \Sigma_{ij} \text{ (interface condition)} \end{cases}$$

Introduce the interface coupling on Σ_{ij}

$$\begin{aligned} g_{ij} &= \rho_0 \left(1 - M_{\boldsymbol{n}}^2\right) \left(-\partial_{\boldsymbol{n}_j} u_j + \imath \mathcal{S}_i u_j\right) \\ &= -g_{ji} + \imath \rho_0 \left(1 - M_{\boldsymbol{n}}^2\right) (\mathcal{S}_i + \mathcal{S}_j) u_j := \mathcal{T}_{ji} g_{ji} + b_{ji} \end{aligned}$$

Rewrite the coupling as a linear system for $\boldsymbol{g} = (g_{ij}, g_{ji})^T$ over all Σ_{ij}

$$\underbrace{(\mathcal{I} - \mathcal{A})}_{\text{iteration matrix}} \underbrace{\boldsymbol{g}}_{\text{interface unknowns}} = \underbrace{\boldsymbol{b}}_{\text{physical sources}}, \quad \mathcal{A} = \begin{pmatrix} 0 & \mathcal{T}_{ji} \\ \mathcal{T}_{ij} & 0 \end{pmatrix}$$

$\mathcal{T}_{ij}$ and $\mathcal{T}_{ji}$ are iteration operators on Σ_{ij}

High-level algorithmic procedure

Surface iteration operators

$$\mathcal{T}_{ji} = \frac{\mathcal{S}_i - \widetilde{\Lambda}^+}{\mathcal{S}_j + \widetilde{\Lambda}^+}, \quad \mathcal{T}_{ij} = \frac{\mathcal{S}_j + \widetilde{\Lambda}^-}{\mathcal{S}_i - \widetilde{\Lambda}^-}$$

Iteration matrix eigenvalues: $\lambda_{(\mathcal{I}-\mathcal{A})} = 1 \pm \sqrt{\mathcal{T}_{ji}\mathcal{T}_{ij}}$

If we choose $\mathcal{S}_i = \widetilde{\Lambda}^+$ and $\mathcal{S}_j = -\widetilde{\Lambda}^-$, we have a direct method

Parallel iterative algorithm for the process i

Do in Ω_i at iteration $(n+1)$, $\forall j \in D_i$

1. given $g_{ij}^{(n)}$, solve $u_i^{(n+1)}$ in Ω_i,
2. update the $(n+1)$ neighbourhood data through $g_{ji}^{(n+1)} = -g_{ij}^{(n)} + \imath\rho_0 \left(1 - M_{\boldsymbol{n}}^2\right) (\mathcal{S}_i + \mathcal{S}_j) u_i^{(n+1)}$ on Σ_{ij},

High-level algorithmic procedure

Surface iteration operators

$$\mathcal{T}_{ji} = \frac{\mathcal{S}_i - \widetilde{\Lambda}^+}{\mathcal{S}_j + \widetilde{\Lambda}^+}, \quad \mathcal{T}_{ij} = \frac{\mathcal{S}_j + \widetilde{\Lambda}^-}{\mathcal{S}_i - \widetilde{\Lambda}^-}$$

Iteration matrix eigenvalues: $\lambda_{(\mathcal{I}-\mathcal{A})} = 1 \pm \sqrt{\mathcal{T}_{ji}\mathcal{T}_{ij}}$

If we choose $\mathcal{S}_i = \widetilde{\Lambda}^+$ and $\mathcal{S}_j = -\widetilde{\Lambda}^-$, we have a direct method

Parallel iterative algorithm for the process i

Do in Ω_i at iteration $(n+1)$, $\forall j \in D_i$

1. given $g_{ij}^{(n)}$, solve $u_i^{(n+1)}$ in Ω_i,
2. update the $(n+1)$ neighbourhood data through $g_{ji}^{(n+1)} = -g_{ij}^{(n)} + \imath\rho_0 \left(1 - M_{\boldsymbol{n}}^2\right) (\mathcal{S}_i + \mathcal{S}_j) u_i^{(n+1)}$ on Σ_{ij},

Problem: $(\widetilde{\Lambda}^+, -\widetilde{\Lambda}^-)$ are **non-local** DtN maps for the PDE

Idea: design **sparse approximations** $\mathcal{S}_i \approx \widetilde{\Lambda}^+$ and $\mathcal{S}_j \approx -\widetilde{\Lambda}^-$

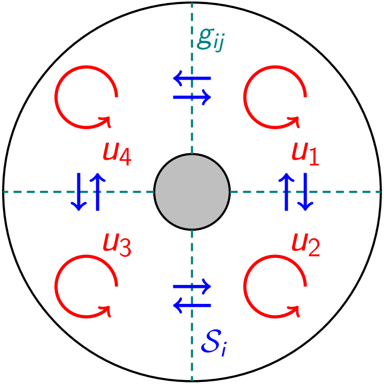
$\Leftrightarrow$ approximate Schur complements at the algebraic level

Illustration of the algorithm

Iterative solver for the interface problem $(\mathcal{I} - \mathcal{A})\mathbf{g} = \mathbf{b}$

Iterate until convergence

1. Solve the volume subproblems u_i with boundary conditions
2. update the interfaces unknowns $\boldsymbol{g} = (g_{ij}, g_{ji})$ through transmission conditions $(\mathcal{S}_i, \mathcal{S}_j)$



- Convergence ? [*Desprès 1991*]
- How to choose the operators $(\mathcal{S}_i, \mathcal{S}_j)$? → [*Gander et al. 2002*], numerous works...

Outline

Approximation of the DtN map

We follow the idea to find local approximations of the DtN maps $(\widetilde{\Lambda}^+, -\widetilde{\Lambda}^-)$ for outgoing waves and use them as transmission conditions

There are several ways to do so:

- Absorbing boundary condition (ABC),
- Infinite element (IE),
- Perfectly Matched Layer (PML),
- etc.

In this talk we focus on absorbing boundary condition.

1. it's a boundary treatment: easy set up of the surfacic problem in a non-overlapping context
2. we need to account for the entire frequency spectrum (Fourier analysis)

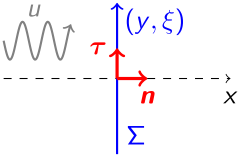
Remark: the extension of ABC, PML and IE techniques for flow acoustics is not straightforward !
$\rightarrow$ we focus on ABC construction for a uniform axial mean flow

DtN map for flow acoustics

Idea: Find an exact form of the DtN map for a half-space problem

DtN operator on Σ

$$\widetilde{\Lambda^+} : \begin{cases} H^{1/2}(\Sigma) \to H^{-1/2}(\Sigma) \\ u_{|\Sigma} \mapsto \partial_{\boldsymbol{n}} u_{|\Sigma} = -i\widetilde{\Lambda^+} u_{|\Sigma} \end{cases}$$



General case: use pseudo-differential calculus

[*Engquist and Majda 1977, 1979*] [*Antoine et al. 1999*]

Example: 2D convected Helmholtz operator ($|M_x| < 1$, $M_y = 0$)

$$\mathcal{L} = (1 - M_x^2)\partial_x^2 + \partial_y^2 - 2\imath\omega M_x \partial_x + \omega^2$$

Question: can we factorize the operator $\mathcal{L}$ on Σ ?

$$\mathcal{L} \underset{?}{=} \left(\partial_x + \imath\widetilde{\Lambda^-}\right)\left(\partial_x + \imath\widetilde{\Lambda^+}\right) \quad \text{on } \Sigma$$

Waveguide case

For the half-space problem with uniform flow $|M_x| < 1$, we have an exact solution:

$$\widetilde{\Lambda}^{\pm} = \omega \frac{-M_x \pm \sqrt{1+Z}}{1 - M_x^2}, \quad Z = \left(1 - M_x^2\right) \frac{\Delta_\Sigma}{\omega^2},$$

Localization of $\widetilde{\Lambda}^{+}$: high-frequency approx. for $\sqrt{1+Z}$, $(\omega \to +\infty)$.

However we want an approximation for all the Fourier modes of Δ_Σ
$\to$ rotate branch-cut and use complex valued approximations of

$$f_\alpha(Z) = e^{\imath\alpha/2}\sqrt{1+\hat{Z}}, \quad \hat{Z} = [e^{-\imath\alpha}(1+Z) - 1],$$

Taylor approximation (N, α)

$$f_\alpha(Z) \approx e^{\imath\alpha/2} \sum_{\ell=0}^{N} \binom{1/2}{\ell} \left(e^{-\imath\alpha}(1+Z) - 1\right)^{\ell}$$

Padé approximation (N, α)

$$f_\alpha(Z) \approx K_0(\alpha) + \sum_{\ell=1}^{N} A_\ell(\alpha) Z (1 + B_\ell(\alpha) Z)^{-1}$$

Square-root function approximation

Let us plot f_α and some approximations along the real line $\Im(Z) = 0$

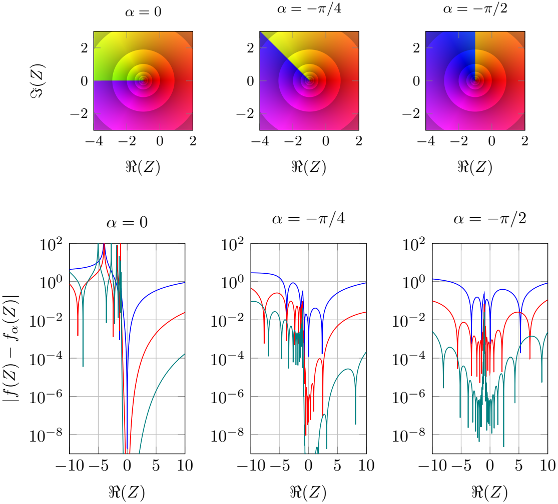
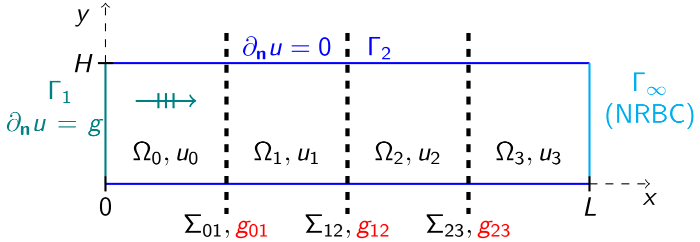


Figure: Absolute error along the real line with Padé approximants: $N = 1$, $N = 2$, $N = 8$

Outline

Domain decomposition in a straight waveguide

Domain partition: $\Omega = \bigcup_{i=0}^{N_{\text{dom}}-1} \Omega_i$, $\Sigma_{ij} = \overline{\partial\Omega_i \bigcap \partial\Omega_j}$, $j \neq i$



Straight waveguide with uniform flow $|M_x| < 1$: the convergence radius has the explicit form:

$$\rho(\xi) = \left|\sqrt{\mathcal{T}_{ji}\mathcal{T}_{ij}}\right| = \left|\frac{(f - f_\alpha)(-2M_x + f - f_\alpha)}{(-2M_x + f + f_\alpha)(f + f_\alpha)}\right|$$

where $f = \sqrt{1 + (1 - M_x^2)(\xi/\omega)^2}$, f_α is the square-root approximation and ξ is the Fourier mode for Δ_Σ

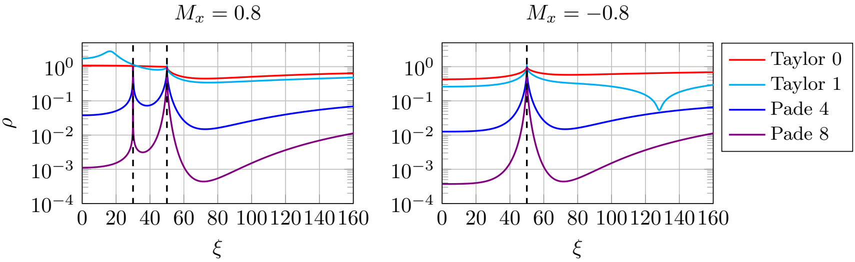


Figure: Convergence radius for various approximations, $\alpha = -\pi/2, \omega = 30$. For $M_x = 0.8$ and $\xi \in [30, 50]$, the wave has negative phase velocity.

Remarks:

- The Taylor approximations can not ensure $\rho < 1$ for all modes,
- In practice, only Padé approximations converge with a Jacobi solver,
- The Taylor 1 approximation results in a 2nd order surface operator, and is straightforward to integrate in an existing code

Outline

High-order FEM implementation

Discretization on a conformal, high-order H^1-basis for an arbitrary flow

Weak formulation for the linearized potential equation

$\forall v \in V \subseteq H^1(\Omega)$,

$$\int_\Omega \left[\rho_0 \nabla u \cdot \overline{\nabla v} - \frac{\rho_0}{c_0^2} \frac{D_0 u}{Dt} \overline{\frac{D_0 v}{Dt}}\right] d\Omega + \imath \int_\Sigma \mathcal{G} u \bar{v} \, d\Sigma = \int_\Omega f \bar{v} d\Omega$$

The boundary operator $\mathcal{G}$ takes the same form as in the Helmholtz case

$$\int_\Sigma \mathcal{G} u \, \bar{v} \, d\Sigma = \int_\Sigma e^{\imath\alpha/2} \rho_0 k_0 \sqrt{1 + \hat{Z}} \, u \, \bar{v} \, d\Sigma$$

with

$\hat{Z} = e^{-\imath\alpha} \left(1 - 2\imath M_{\boldsymbol{\tau}} \frac{\nabla_\Sigma}{k_0} + \left(1 - \mathrm{M}^2\right) \frac{\Delta_\Sigma}{k_0^2}\right) - 1, \; \mathrm{M} = \sqrt{M_{\boldsymbol{n}}^2 + M_{\boldsymbol{\tau}}^2}, \; k_0 = \omega / c_0$

High-order FEM implementation

Discretization on a conformal, high-order H^1-basis for an arbitrary flow

Weak formulation for the linearized potential equation

$\forall v \in V \subseteq H^1(\Omega)$,

$$\int_\Omega \left[\rho_0 \nabla u \cdot \overline{\nabla v} - \frac{\rho_0}{c_0^2} \frac{D_0 u}{Dt} \overline{\frac{D_0 v}{Dt}}\right] d\Omega + \imath \int_\Sigma \mathcal{G} u \overline{v} \, d\Sigma = \int_\Omega f \overline{v} d\Omega$$

The boundary operator $\mathcal{G}$ takes the same form as in the Helmholtz case

$$\int_\Sigma \mathcal{G} u \, \overline{v} \, d\Sigma = \int_\Sigma e^{\imath\alpha/2} \rho_0 k_0 \sqrt{1 + \hat{Z}} \, u \, \overline{v} \, d\Sigma$$

with

$\hat{Z} = e^{-\imath\alpha} \left(1 - 2\imath M_{\boldsymbol{\tau}} \frac{\nabla_\Sigma}{k_0} + \left(1 - \mathrm{M}^2\right) \frac{\Delta_\Sigma}{k_0^2}\right) - 1, \ \mathrm{M} = \sqrt{M_{\boldsymbol{n}}^2 + M_{\boldsymbol{\tau}}^2}, \ k_0 = \omega / c_0$

Example : 2nd order Taylor approximation of the square-root:

$$\int_\Sigma \mathcal{G} u \, \overline{v} \, d\Sigma = \cos(\alpha/2) \int_\Sigma \rho_0 k_0 \, u \, \overline{v} \, d\Sigma + e^{-\imath\alpha/2} \left(\int_\Sigma \rho_0 M_{\boldsymbol{\tau}} \nabla_\Sigma u \, \overline{v} \, d\Sigma - \int_\Sigma \rho_0 \frac{(1 - \mathrm{M}^2)}{2k_0} \nabla_\Sigma u \, \nabla_\Sigma \overline{v} \, d\Sigma \right)$$

Outline

Extending transmission conditions

Such DtN approximations can be extended to

- regular convex shaped boundaries
- non-uniform flows, density and speed of sound

The methodology is to expand the DtN operator into its **symbols**: the leading term encodes uniform flow and straight boundary.

Convergence difficulties are expected in the inverse upstream regime for the 2nd order Taylor based transmission condition
$\rightarrow$ try instead a coercive 2nd order approximation ?

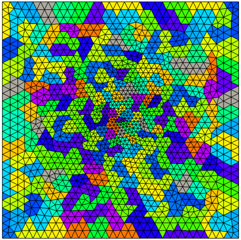
The extension to Pierce operator is direct: same variational formulation ($\rho_0^{-1}(x) \leftrightarrow \rho_0(x)$): encodes richer physics (more complex mean flows)

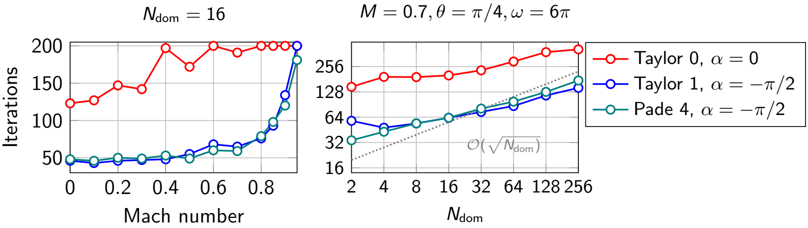
Towards realistic cases

Automatic partitioning

- Cross-points
  $\rightarrow$ harder to design ABCs
- Good load balancing between subproblems
- Shorter connectivity graph - $\mathcal{O}(\sqrt{N_{\text{dom}}})$

$N_{\text{dom}} = 256$





We choose Taylor 1, $\alpha = -\pi/2$ for arbitrary decomposition

Outline

Introduction

Goal: propose a proof of concept of a DDM solver with industrial constraints:

- minimize the implementation overhead from a given FEM code,
- parallelism must be hidden to the user,
- switch to a parallel solver when necessary (frequency criterion),
- can be tested/validated easily,

Outline

Implementation framework

C++ distributed memory implementation

- Gmsh: mesh generation, partitioning (METIS)
- GmshFEM: Finite element library, subdomain solver (MUMPS)
- GmshDDM: Interface problem, communication (MPI), iterative solver

First, create a mesh partition that minimizes the size of the interface problem.

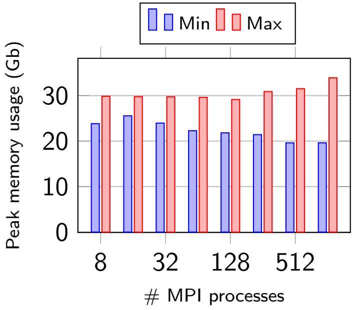
Domain decomposition algorithm for the i-th process linked to the subdomain Ω_i

1. **Initialization: read mesh, map mean flow, initialize interface problem**
2. **Assembly:** assemble the finite element matrix,
3. **Factorization:** call the external MUMPS solver via PETSc and run a sparse LU decomposition for the volume subproblem,
4. **Surface Assembly:** assemble the surface interface problem,
5. **Iterative solver:** enter the iterative solver (PETSc GMRES) for the interface problem $(\mathcal{I} - \mathcal{A})g = f$. Do until convergence:
   5.1 receive g_{ij} and send the updated data g_{ji} to the connected subdomains,
   5.2 compute the local matrix-vector product,
6. **Post-process:** save the solution.

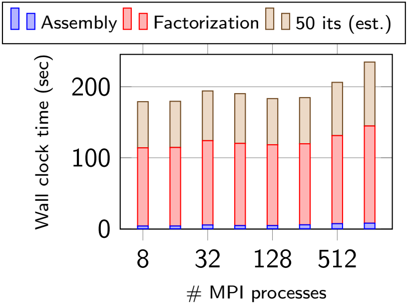
Outline

Weak scalability assessment - Helmholtz case

Evaluate time and memory usage: assembly, factorization, cost per iteration.
Weak scalability up to 700M dofs - 1024 MPI processes: ≈ 80% efficiency
We assign 1 MPI process per subdomain



Min/max peak memory usage over all MPIs.



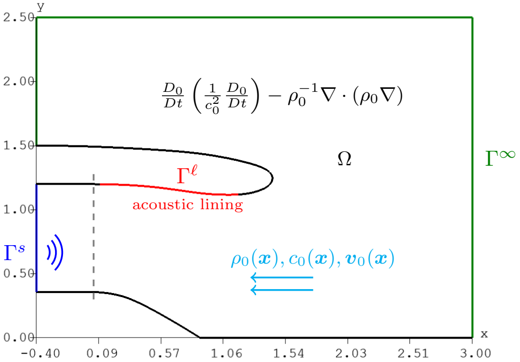
Cumulated wall time.

- ≈ 1M dofs per subdomain leads a reasonable factorization cost
- each process takes advantage of multi-threading
- load balancing is affected by the number of subdomains

Outline

The boundary value problem

Given a flight configuration (mean flow), predict the radiated noise from the fan, at multiples of the blade passing frequency $\omega_{\mathrm{bpf}}/(2\pi) = 1300$ Hz



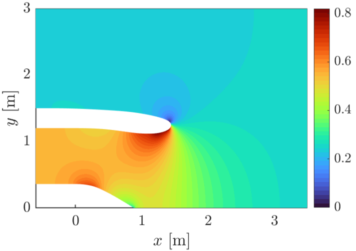
Boundary conditions

- Ingard-Myers on Γ_ℓ
- PML (active) on Γ^s
- Fixed annular Bessel mode on Γ^s
- PML (passive) on Γ^∞

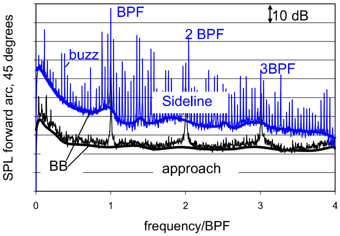
The mean flow is pre-computed (non-linear Poisson) and mapped on the acoustic mesh

Physical interpretation

The problem models the **blade passing frequency** (bpf) of the fan: links the annular mode numbers (m, n) with the input frequency ω_{bpf}.



Mach number $M = \|\mathbf{v}_0\| / c_0$ for a typical mean flow.



Sound pressure level for an engine intake from a static engine test.

Computational domain: 3D cylinder (radius 2.5m, length 3.4m).

- $n \times$bpf $\leftrightarrow$ $n \times 25$ wavelengths in the domain
- mesh size and p-FEM order are fixed to have at least 5 dofs per λ

Acoustic lining boundary condition

The acoustic lining is defined by a complex-valued impedance $\mathcal{Z}(\omega)$, which can be modeled by a boundary operator on Γ_ℓ

$$\int_{\Gamma_\ell} -\rho_0 \frac{\partial u}{\partial \boldsymbol{n}_\ell} \overline{v} \, d\Gamma_\ell$$

$$= \int_{\Gamma_\ell} \frac{\rho_0}{\mathcal{Z}(\omega) c_0} \left(\mathrm{i}\omega u \overline{v} + (\boldsymbol{v}_0 \cdot \nabla_\Gamma u) \, \overline{v} - u \, (\boldsymbol{v}_0 \cdot \nabla_\Gamma \overline{v}) - \frac{1}{\mathrm{i}\omega} (\boldsymbol{v}_0 \cdot \nabla_\Gamma u) \, (\boldsymbol{v}_0 \cdot \nabla_\Gamma \overline{v}) \right) \, d\Gamma_\ell.$$

It results in broader modal content of the numerical solution (evanescent modes, reflected waves, etc.).

The implementation has been validated by a mode-matching method

Outline

Reaching high frequency

1 bpf: $\approx$ 10M dofs $\rightarrow$ 740 Gb RAM direct solver (25 λ)
2 bpf: $\approx$ 70M dofs $\rightarrow$ 6 Tb RAM direct solver (50 λ)

Nic5 Uliège cluster - (AMD Epyc Rome 7542 CPUs 2.9 GHz)

Cores	Total dofs	nnz	peak memory	pre-pro	GMRES	Iterations
64×1	73 M	5 B	26 Gb	45min	4h30	535

Run at 2bpf with 64 subdomains (64 MPI) and acoustic lining. p-FEM$=4$

IDRIS Jean-Zay CPU partition – 256 nodes (2 x 20 Intel Cascade Lake 6248 2,5Ghz)

Cores	Total dofs	nnz	peak memory	pre-pro	GMRES	Iterations
512×20	565M	85B	70Gb	24min	8h20	3000

Run at 4bpf with 512 subdomains (512 MPI) and acoustic lining. p-FEM$=6$

GMRES stopped after 3000 Iterations at a residual $r_l = 2.7 \times 10^{-6}$

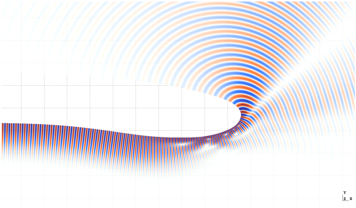
Reaching high frequency

IDRIS Jean-Zay CPU partition – 512 nodes (2 x 20 Intel Cascade Lake 6248 2,5Ghz)

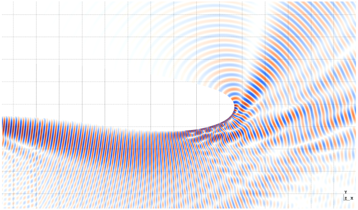
Cores	Total dofs	nnz	peak memory	pre-pro	GMRES	Iterations
1024×20	1.1B	167B	70Gb	24min	6h10	2253

Run at 5bpf with 1024 subdomains (1024 MPI) and acoustic lining. p-FEM$= 6$

125 wavelengths in the domain - validation with the axisymmetric case



Mode $(48, 1)$ without liner



Mode $(48, 1)$ with acoustic liner

Visualization at 5 bpf

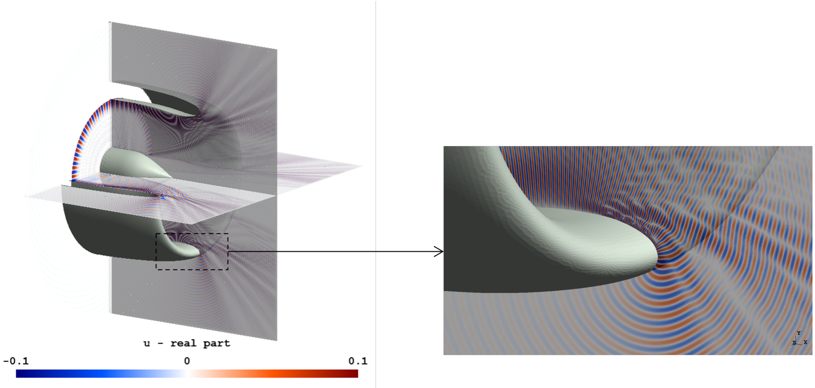


Figure: Real part of the acoustic velocity potential for the mode $(48, 1)$ at 5 bpf.

Outline

Limitations and extensions

We proposed a scalable high frequency distributed memory solver for realistic flow acoustics radiation problems.

- the number of iterations depends of the frequency, mesh size and N_{dom}: a coarse space would be highly beneficial
- transmission conditions are PDE based: hard to extend to Maxwell, elastic waves, etc.
- cross-point treatment is hard with mean flow anisotropy
- subdomain factorization cost could be reduced with `MUMPS` block low rank feature
- the next step is the extension to Pierce equation (turbofan exhaust)