Distributed Iterative Methods Applied in CFD Problems

Renato S. Silva

LNCC - Laboratório Nacional de Computação Científica

Rio de Janeiro - Brazil

Visiting Scholar from TICAM - Texas Institute for Computational and Applied Mathematics

The University of Texas at Austin

Motivation

- Solve "large" CFD problems

  - transport equation

  - compressible Euler and Navier-Stokes Eqs.

- Petrov-Galerkin Formulations

  - SUPG (test)

  - CAU (*Consistent Approximate Upwind*)

- Adaptive Methods

  - h

  - h-p

$$
A\,x = b \quad \left\{ \begin{array}{l} \text{- Nonlinear} \\ \text{- Nonsymmetric} \\ \text{- Sparse} \\ \text{- } \textit{Large} \end{array} \right.
$$

- Good Solver

  - reliable
  - fast

- Iterative Method

  - Adaptive Methods $\Longrightarrow$ $x_0^i \longleftarrow x^{i-1}$
  - Nonlinear Methods
  - Sparse Matrices $\Longrightarrow$ EBE
  - Save Memory

GMRES Method (Saad and Schultz, 1986)

$x = x_0 + z$

$min \, \|b - A(x_0 + z)\|$ over Krylov space:

$$span \left\{ r_0, Ar_0, A^2 r_0, \ldots, A^k r_0 \right\}$$

$$r_0 = b - Ax_0$$

Properties:

- $\|r^{i+1}\| \leq \|r^i\|$
- seldom breaks down
- computational work increase with k
- memory increase $\propto k^2$

Restarted GMRES(k)

after k orthogonalization steps

$$x_0^{i+1} \longleftarrow x_k^i$$

Drawbacks:

How to choose k?

(CFD problems: $5 \leq k \leq 30$)

Algorithm: GMRES(k) - with right Preconditioning

1. **Start:** Choose x_0 and a dimension k of the Krylov subspaces.
2. **Arnoldi process:**
   (a) Compute $r_0 = b - Ax_0,\ \beta = \|r_0\|$ and $v_1 = r_0/\beta$.
   (b) For $j = 1, \ldots, k$ do
   - Compute $z_j := M^{-1}v_j$
   - Compute $w := Az_j$
   - For $i = 1, \ldots, j,$ do $\begin{cases} h_{i,j} := (w, v_i) \\ w := w - h_{i,j}v_i \end{cases}$
   - Compute $h_{j+1,j} = \|w\|$ and $v_{j+1} = w/h_{j+1,j}$.

   (c) Define $V_k := [v_1, \ldots, v_k]$ and $\bar{H}_k = \{h_{i,j}\}_{1 \le i \le j+1; 1 \le j \le k}$.
3. **Form the approximate solution:** Compute $x_k = x_0 + M^{-1}V_k\ y_k$,
   where $y_k = argmin_y \|\beta e_1 - \bar{H}_k\ y\|$ and $e_1 = [1, 0, \ldots, 0]^t$.
4. **Restart:** If satisfied stop, else set $x_0 \longleftarrow x_k$ and go to 2.

Homogeneous Distributed Computer

Def.: Collection of computers interconnected by a network.

Advantages:

- cheap
- performance

PVM (Parallel Virtual Machine) - version 3.3.7

- portability

Homogeneous Distributed Computer (cont.)

Drawbacks:

- many users
- different machines

traffic

load inbalance

- network type:
  - ∗ Ethernet (10 Mbps)
    - all to all
    - only one message at time
  - ∗ FDDI (100Mbps)
- communication is very expensive

Parallel GMRES(k)

- matrix - vector products $O(n^2)$

  EBE $\Rightarrow$ nearest neighbor communication

  - reduce the number of neighbors

  $$\Downarrow$$

  unidimensional partition of the domain

- scalar products $O(n)$

  broadcast the scalar products results:

  - only function of the Krylov dimension

  - less expensive than exchange all vector

  $$\Downarrow$$

  constant communication cost by iteration

Additive Schwarz Method

- **Preconditioner**

$$A^{-1} \approx \sum_{i=1}^{N} R_i^t \; A_i^{-1} \; R_i = M_{ASM}^{-1}$$

$$A_i = R_i \; A \; R_i^t$$

R_i^t (Restriction matrix)

$R_i^t \; A_i^{-1} \; R_i \longrightarrow$ can be solved on separate processors

- Convergence

  - Overlap size (δ)

  - Number of domains

$\delta = 2$ elements				
$\#\Omega_i$	2	4	8	16
it.	1	2	6	10

Numerical Experiments

- SUPG Method
- GMRES: - $k = 5$

  - transport eq. $\longrightarrow tol = 10^{-6}$

  - Navier-Stokes eqs. $\longrightarrow tol = 10^{-4}$
- BLAS (Level 1, 2)
- Additive Schwarz Method:

  - LU decomposition - LAPACK
- Virtual Machine:

  - 16 RS 6000, Model 390

  - FDDI
- PVM: - DataInPlace

  - Dynamic TCP sockets
- Time: - gettimeofday

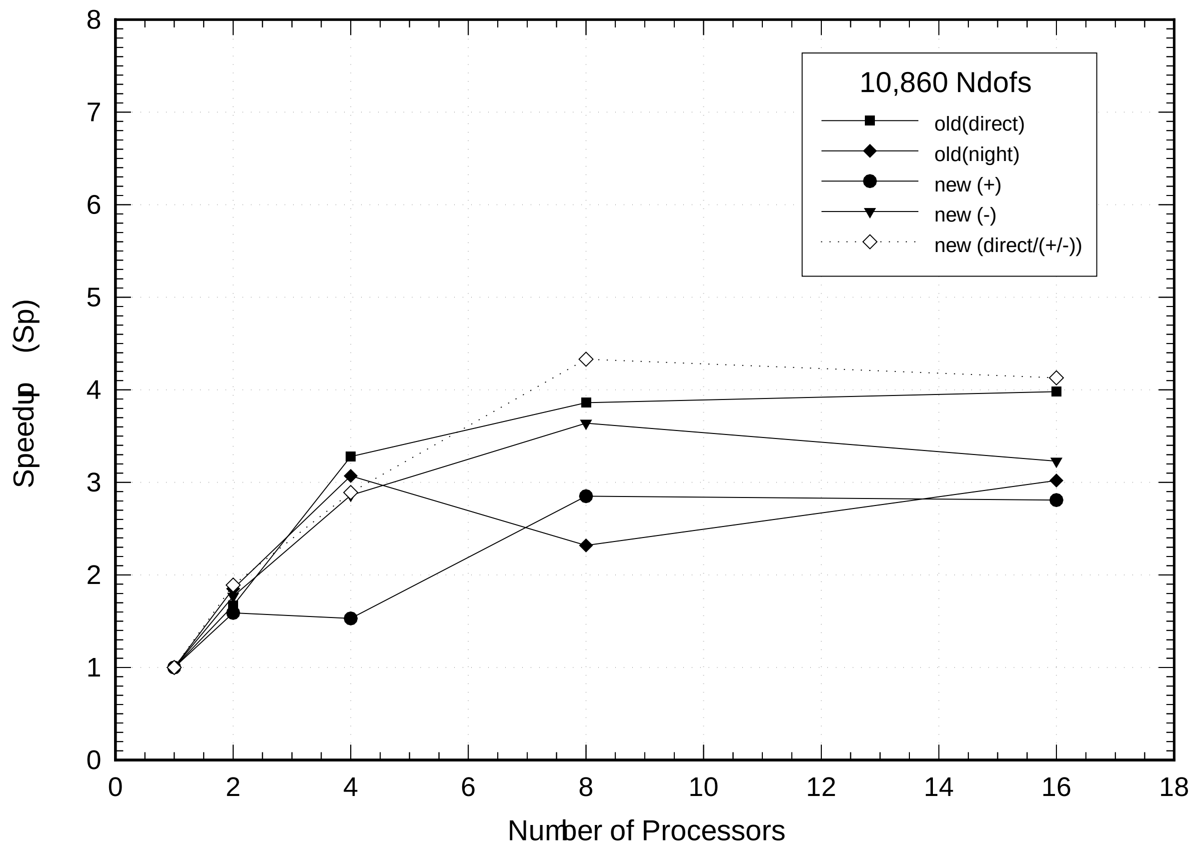
Example 1: Unidimensional Transport Equation

$$-k\Delta\varphi + u\nabla\varphi = f$$

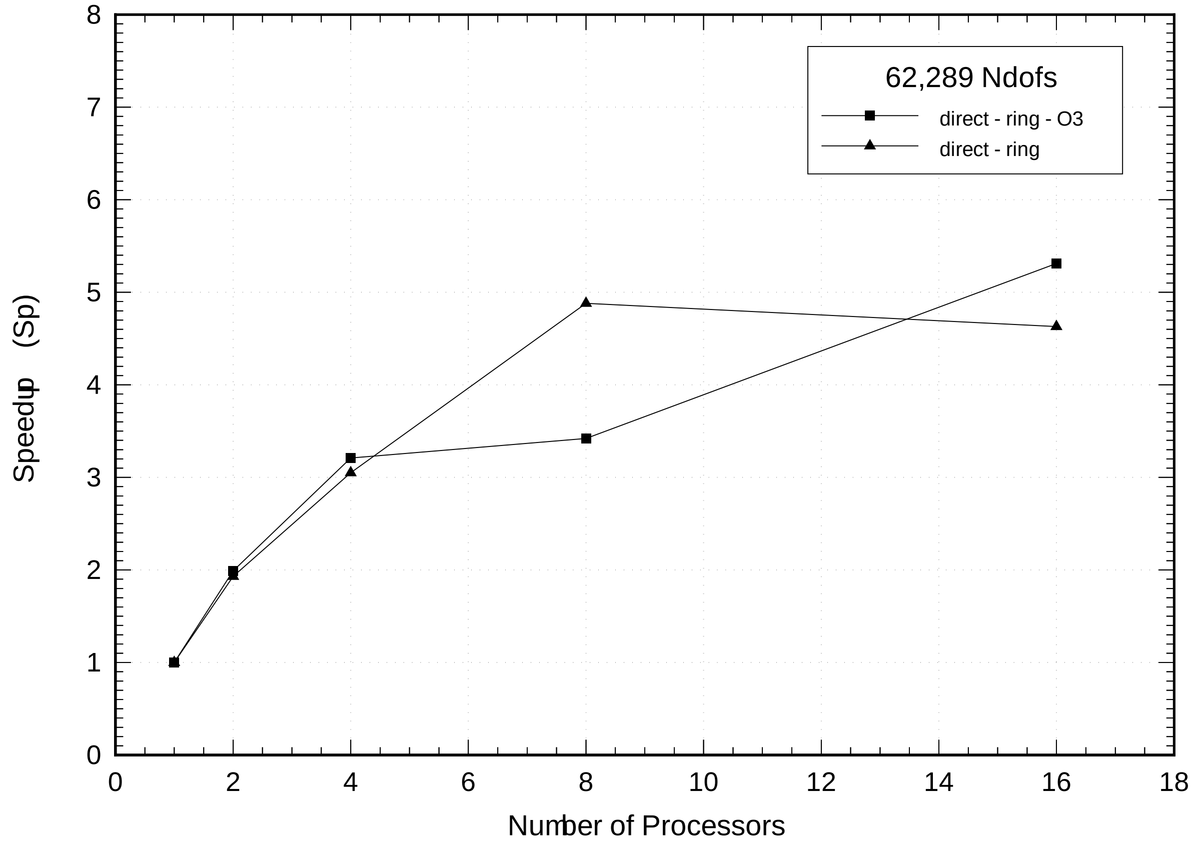
$$\varphi(0) = 0; \qquad \varphi(1) = 1$$

velocity field	$\lvert u \rvert = 1$
diffusion tensor	$k = 10^{-6}$
source term	$f = 1.0$

Example 1: Unidimensional Transport Equation



Example 1: Unidimensional Transport Equation

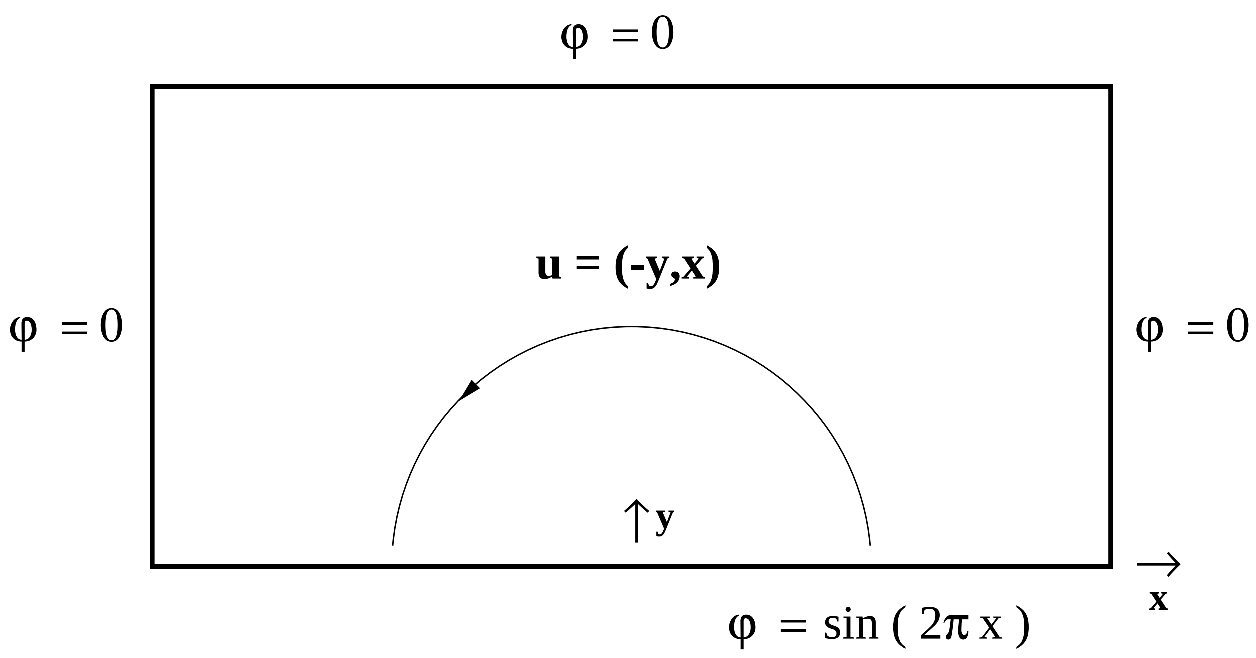


Example 2: Two-dimensional Transport Equation

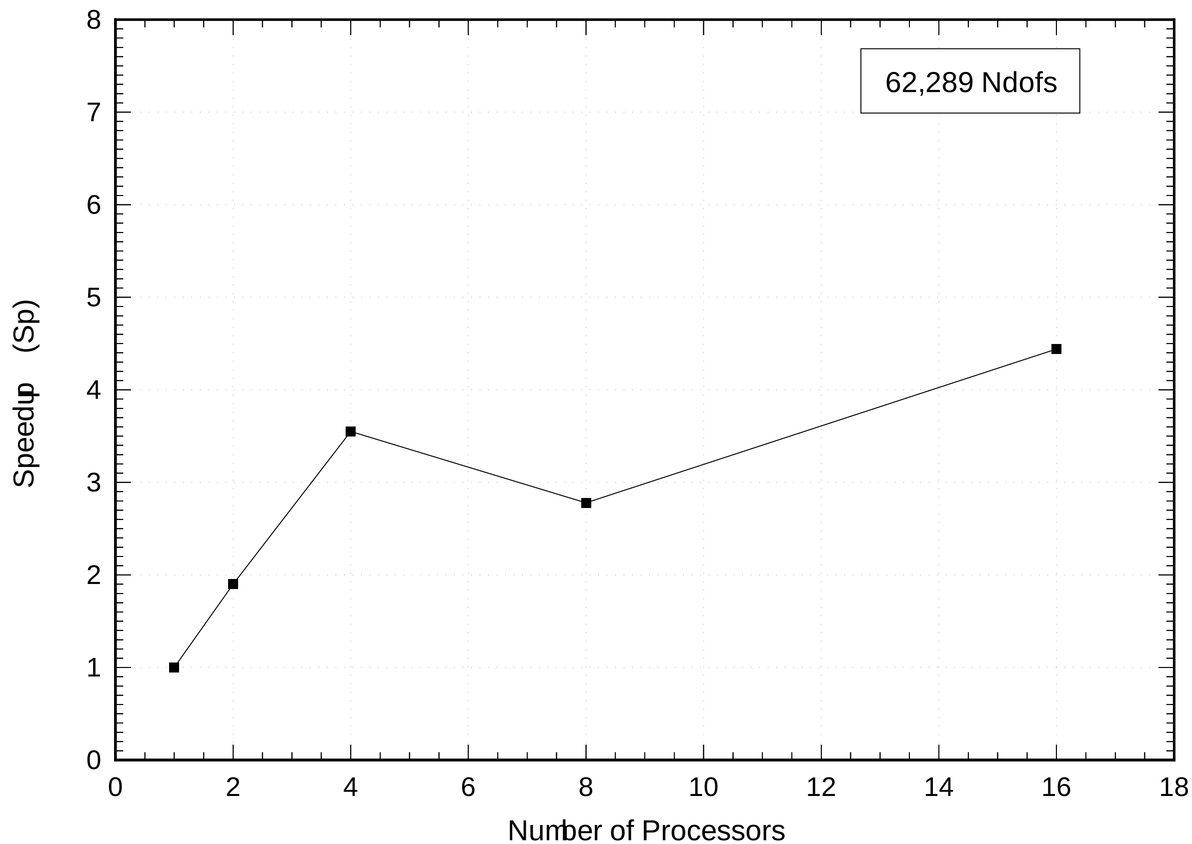
$|u| = 1$

$k = 10^{-6}$

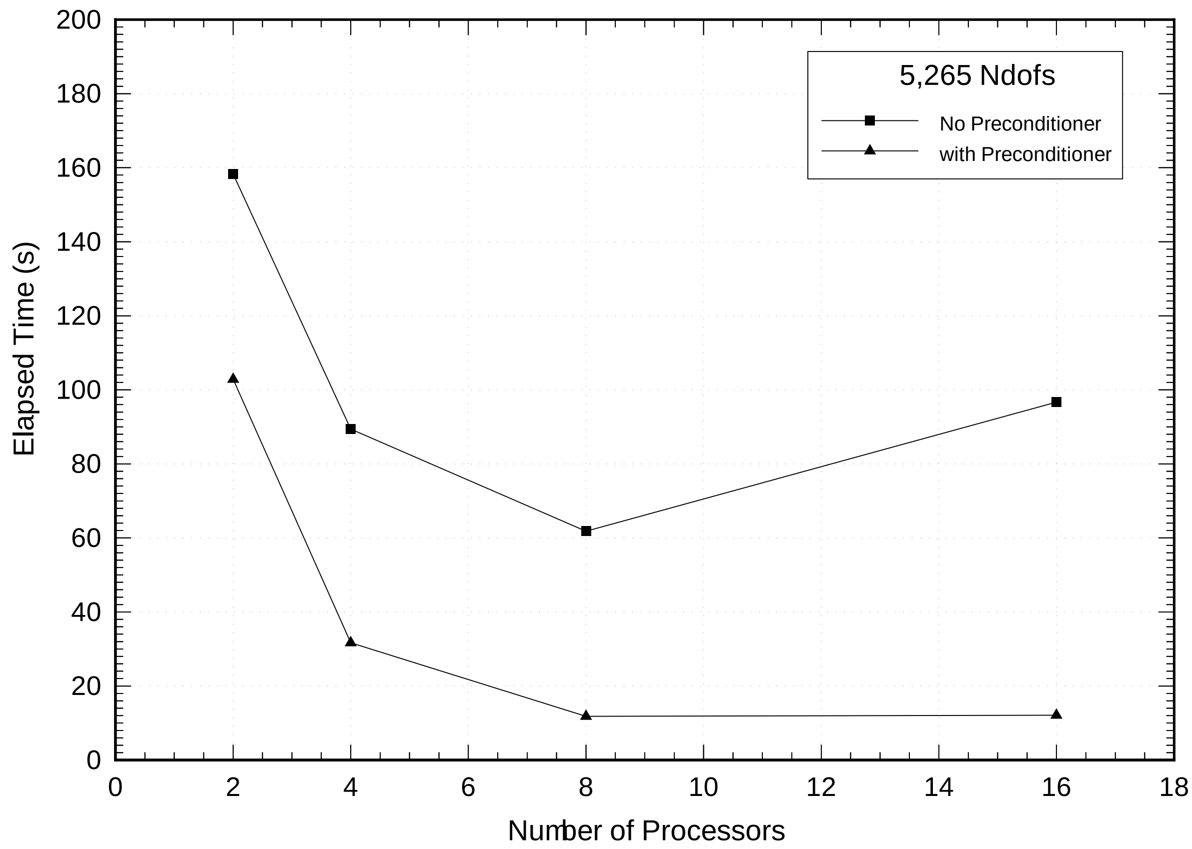
$f = 0.$



Example 2: Two-dimensional Transport Equation



Example 2: Two-dimensional Transport Equation



Navier-Stokes Equations

$$\boxed{U_{,t} \; + \; F_{i,i} \; = \; F^{\nu}_{i,i} \; + \; F^{h}_{i,i} \; + \; F}$$

conservation variables

$$U = \begin{Bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{Bmatrix} = \rho \begin{Bmatrix} 1 \\ u_1 \\ u_2 \\ e \end{Bmatrix}$$

Euler flux

$$F_i = \rho u_i \begin{Bmatrix} 1 \\ u_1 \\ u_2 \\ e \end{Bmatrix} + p \begin{Bmatrix} 0 \\ \delta_{1i} \\ \delta_{2i} \\ u_i \end{Bmatrix}$$

viscous flux

$$F^{\nu}_i = \begin{Bmatrix} 0 \\ \tau_{1i} \\ \tau_{2i} \\ \tau_{ij} u_j \end{Bmatrix}$$

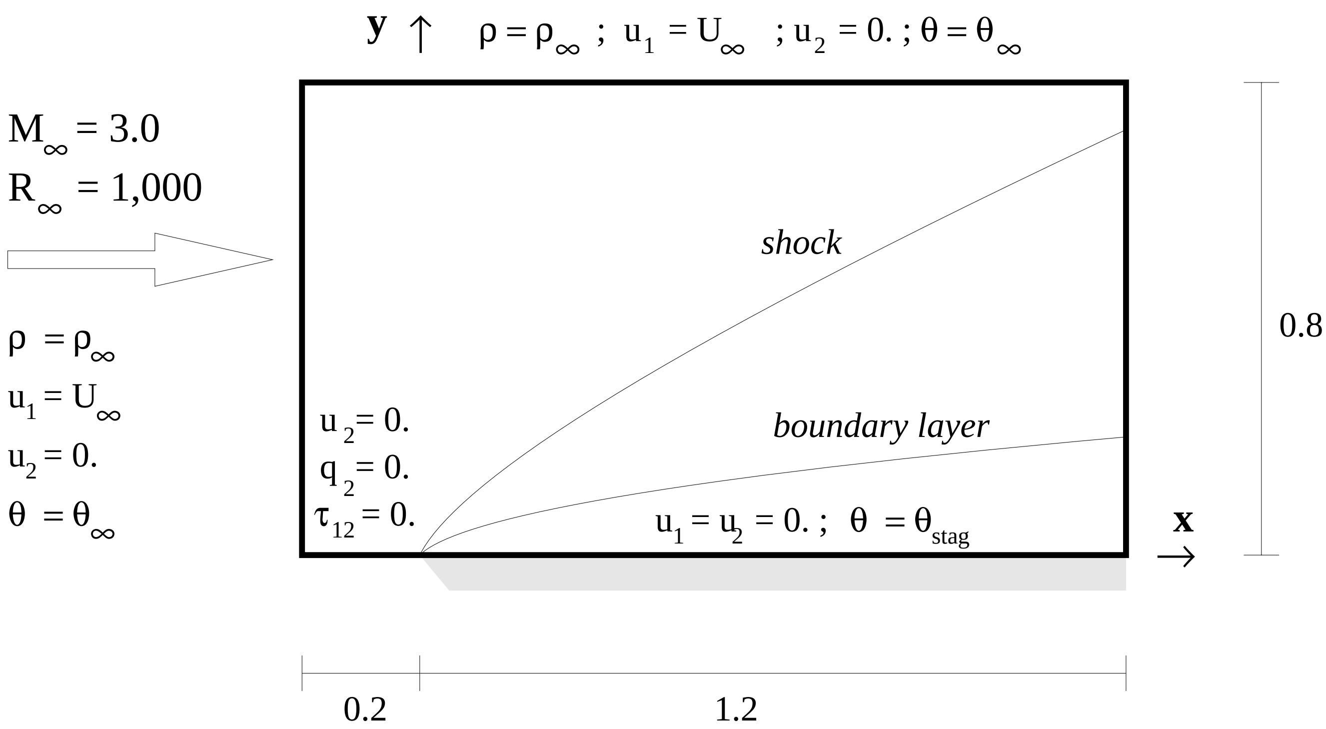
heat flux

$$F^{h}_i = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ -q_i \end{Bmatrix}$$

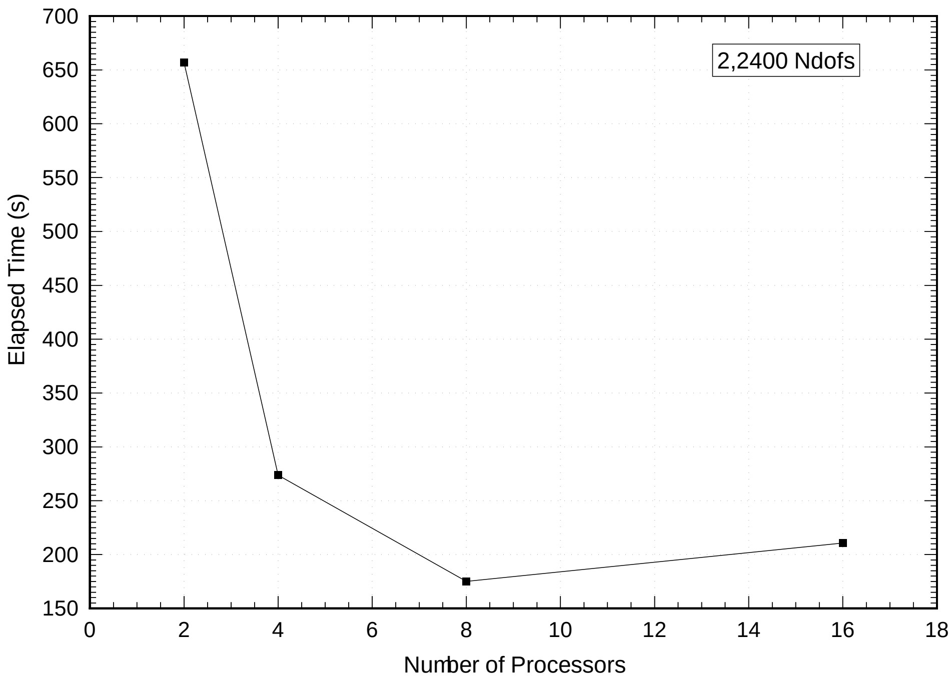
source term

$$F = \begin{Bmatrix} 0 \\ \rho b_1 \\ \rho b_2 \\ \rho b_i u_i + \rho r \end{Bmatrix}$$

Example 3: Navier-Stokes Equations



Example 3: Navier-Stokes Equations



5,580 Ndofs	
Serial	Distributed (p = 16)
Frontal	ASM - GMRES(k)
39.3 min	8.1 min

Conclusions

- Algorithm:

  - good performance also for *Large* problems
  - partition and communication pattern are appropriate for distributed systems

- Additive Schwarz Method:

  - effective for distributed systems
    - save iterations using few communications

Distributed Finite Element Solver

$\Downarrow$

new possibilities for solving CFD problems