

DE LA RECHERCHE À L'INDUSTRIE

cea



**RUNTIME KAAPI DANS
EUROPLEXUS POUR LA
DYNAMIQUE RAPIDE EN INTERACTION
FLUIDE-STRUCTURE**



Journée *runtime* | Groupe Calcul | Vincent Faucher

www.cea.fr

20 JANVIER 2017

**PLEXUS** (1977-1999)

Simulation en dynamique rapide
pour les situations accidentelles du
nucléaire civil

O. Jamond, H. Bung, P. Galon, F. Bliard, A. Beccantini

**EURDYN** (1973-1988)

Simulation en dynamique rapide

PLEXIS-3C (1985-1999)

Structure de données dérivée de PLEXUS
Intégration des fonctionnalités d'EURDYN
(fluide-structure en particulier)

M. Larcher, G. Valsamos, F. Casadei



<http://www-epx.cea.fr>

Copropriété CEA/JRC

Développement ouvert à un **nombre limité de partenaires majeurs** (EDF, ONERA, SAFRAN TECH)

Distribution par le CEA à partir de 2013 : **Version de Production** sous licence, **Version Education & Recherche** gratuite pour l'académie

1	Contexte algorithmique	5
	<ul style="list-style-type: none">— Algorithme de résolution en dynamique rapide des fluides et des structures— Aspects logiciels et illustrations	
2	Parallélisme dominant et problématique	9
	<ul style="list-style-type: none">— Décomposition de domaine et gestion des frontières— Prise en compte générique des contraintes cinématiques— Optimisation et verrous	
3	Implémentation du runtime KAAPI	15
	<ul style="list-style-type: none">— Motivations et implémentation de base— Mise en œuvre à grande échelle (<i>PRACE Preparatory Acces</i> sur TGCC/Curie)	
4	R&D et collaboration CEA/INRIA	22
	<ul style="list-style-type: none">— Optimisation du placement mémoire— Parallélisme multi-niveaux— Perspectives	

■ Equations locales

$$\rho \ddot{\mathbf{q}} + \nabla \cdot \{ \boldsymbol{\sigma} [\boldsymbol{\varepsilon}(\mathbf{q})] \} = \mathbf{f}_{\text{vol}}^{\text{str}}$$

$$\rho \dot{\mathbf{u}} + \nabla P + \mathbf{f}_{\text{trans}}(\mathbf{u}) = \mathbf{f}_{\text{vol}}^{\text{flu}}$$

$$\dot{\rho} + \nabla \cdot (\rho \mathbf{u}) = 0$$

$$\dot{E} + \nabla \cdot [\mathbf{u}(E + P)] = 0$$

■ Contraintes cinématiques

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{u}, \dot{\mathbf{u}}) = \mathbf{S}$$

■ Intégration temporelle explicite

$$\dot{\mathbf{q}}^{n+1/2} = \dot{\mathbf{q}}^n + \frac{\Delta t}{2} \ddot{\mathbf{q}}^n$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \dot{\mathbf{q}}^{n+1/2}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \dot{\mathbf{u}}^n$$

■ Caractéristiques

Non-linéarité géométrique

$$\boldsymbol{\varepsilon}(\mathbf{q}) = \frac{1}{2} (\nabla \mathbf{q} + {}^t \nabla \mathbf{q} - \nabla \mathbf{q} {}^t \nabla \mathbf{q})$$

Stabilité conditionnelle

$$\Delta t \leq \frac{2}{\omega_{\text{max}}} \leftarrow \Delta t \leq \frac{l_c}{c}$$

■ Equations locales

$$\rho \ddot{\mathbf{q}} + \nabla \cdot \{ \boldsymbol{\sigma} [\boldsymbol{\varepsilon}(\mathbf{q})] \} = \mathbf{f}_{\text{vol}}^{\text{str}}$$

$$\rho \dot{\mathbf{u}} + \nabla \mathbf{P} + \mathbf{f}_{\text{trans}}(\mathbf{u}) = \mathbf{f}_{\text{vol}}^{\text{flu}}$$

$$\dot{\rho} + \nabla \cdot (\rho \mathbf{u}) = 0$$

$$\dot{\mathbf{E}} + \nabla \cdot [\mathbf{u}(\mathbf{E} + \mathbf{P})] = 0$$

■ Contraintes cinématiques

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{u}, \dot{\mathbf{u}}) = \mathbf{S}$$

■ Intégration temporelle explicite

$$\dot{\mathbf{q}}^{n+1/2} = \dot{\mathbf{q}}^n + \frac{\Delta t}{2} \ddot{\mathbf{q}}^n$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \dot{\mathbf{q}}^{n+1/2}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \dot{\mathbf{u}}^n$$

■ Caractéristiques

Non-linéarité géométrique

$$\boldsymbol{\varepsilon}(\mathbf{q}) = \frac{1}{2} (\nabla \mathbf{q} + {}^t \nabla \mathbf{q} - \nabla \mathbf{q} {}^t \nabla \mathbf{q})$$

Stabilité conditionnelle

$$\Delta t \leq \frac{2}{\omega_{\text{max}}} \leftarrow \Delta t \leq \frac{l_c}{c}$$

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + \mathbf{F}_{\text{link}}^{n+1} = \begin{bmatrix} \mathbf{F}_{\text{vol}}^{\text{str}} \\ \mathbf{F}_{\text{vol}}^{\text{flu}} \end{bmatrix}^{n+1} - \begin{bmatrix} \mathbf{F}_{\text{int}}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{\text{trans}}(\mathbf{U}^{n+1}) \end{bmatrix}$$

$$\mathbf{C}^{n+1} \begin{bmatrix} \dot{\mathbf{Q}}^{n+1} \\ \mathbf{U}^{n+1} \end{bmatrix} = \mathbf{S}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

■ Equations locales

$$\rho \ddot{\mathbf{q}} + \nabla \cdot \{ \boldsymbol{\sigma} [\boldsymbol{\varepsilon}(\mathbf{q})] \} = \mathbf{f}_{\text{vol}}^{\text{str}}$$

$$\rho \dot{\mathbf{u}} + \nabla P + \mathbf{f}_{\text{trans}}(\mathbf{u}) = \mathbf{f}_{\text{vol}}^{\text{flu}}$$

$$\dot{\rho} + \nabla \cdot (\rho \mathbf{u}) = 0$$

$$\dot{\mathbf{E}} + \nabla \cdot [\mathbf{u}(\mathbf{E} + P)] = 0$$

■ Contraintes cinématiques

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{u}, \dot{\mathbf{u}}) = \mathbf{S}$$

■ Intégration temporelle explicite

$$\dot{\mathbf{q}}^{n+1/2} = \dot{\mathbf{q}}^n + \frac{\Delta t}{2} \ddot{\mathbf{q}}^n$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \dot{\mathbf{q}}^{n+1/2}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \dot{\mathbf{u}}^n$$

■ Caractéristiques

Non-linéarité géométrique

$$\boldsymbol{\varepsilon}(\mathbf{q}) = \frac{1}{2} (\nabla \mathbf{q} + {}^t \nabla \mathbf{q} - \nabla \mathbf{q} {}^t \nabla \mathbf{q})$$

Stabilité conditionnelle

$$\Delta t \leq \frac{2}{\omega_{\max}} \leftarrow \Delta t \leq \frac{l_c}{c}$$

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + \mathbf{F}_{\text{link}}^{n+1} = \begin{bmatrix} \mathbf{F}_{\text{vol}}^{\text{str}} \\ \mathbf{F}_{\text{vol}}^{\text{flu}} \end{bmatrix}^{n+1} - \begin{bmatrix} \mathbf{F}_{\text{int}}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{\text{trans}}(\mathbf{U}^{n+1}) \end{bmatrix}$$

$$\mathbf{C}^{n+1} \begin{bmatrix} \dot{\mathbf{Q}}^{n+1} \\ \mathbf{U}^{n+1} \end{bmatrix} = \mathbf{S}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

■ Calcul des forces de liaison

— Approche par pénalisation : $\mathbf{F}_{\text{link}}^{n+1} = \mathbf{K} \left(\mathbf{C}^{n+1} \begin{bmatrix} \dot{\mathbf{Q}}^{n+1/2} \\ \mathbf{U}^{n+1} \end{bmatrix} - \mathbf{S}^{n+1} \right)$

✓ Forces de liaisons explicites

✗ Choix des coefficients de pénalité

✗ Impact sur la stabilité de l'intégration temporelle

— Approche par dualisation : $\mathbf{F}_{\text{link}}^{n+1} = {}^t \tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda}$

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t \tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}_{\text{vol}}^{\text{str}} \\ \mathbf{F}_{\text{vol}}^{\text{flu}} \end{bmatrix}^{n+1} - \begin{bmatrix} \mathbf{F}_{\text{int}}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{\text{trans}}(\mathbf{U}^{n+1}) \end{bmatrix}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \dot{\mathbf{Q}}^{n+1} \\ \mathbf{U}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

✓ Vérification exacte des liaisons

✓ Pas d'impact sur la stabilité de l'intégration

✗ Système matriciel à construire et résoudre

■ Equations locales

$$\rho \ddot{\mathbf{q}} + \nabla \cdot \{ \boldsymbol{\sigma} [\boldsymbol{\varepsilon}(\mathbf{q})] \} = \mathbf{f}_{\text{vol}}^{\text{str}}$$

$$\rho \dot{\mathbf{u}} + \nabla P + \mathbf{f}_{\text{trans}}(\mathbf{u}) = \mathbf{f}_{\text{vol}}^{\text{flu}}$$

$$\dot{\rho} + \nabla \cdot (\rho \mathbf{u}) = 0$$

$$\dot{\mathbf{E}} + \nabla \cdot [\mathbf{u}(\mathbf{E} + P)] = 0$$

■ Contraintes cinématiques

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{u}, \dot{\mathbf{u}}) = \mathbf{S}$$

■ Intégration temporelle explicite

$$\dot{\mathbf{q}}^{n+1/2} = \dot{\mathbf{q}}^n + \frac{\Delta t}{2} \ddot{\mathbf{q}}^n$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \dot{\mathbf{q}}^{n+1/2}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \dot{\mathbf{u}}^n$$

■ Caractéristiques

Non-linéarité géométrique

$$\boldsymbol{\varepsilon}(\mathbf{q}) = \frac{1}{2} (\nabla \mathbf{q} + {}^t \nabla \mathbf{q} - \nabla \mathbf{q} {}^t \nabla \mathbf{q})$$

Stabilité conditionnelle

$$\Delta t \leq \frac{2}{\omega_{\text{max}}} \leftarrow \Delta t \leq \frac{l_c}{c}$$

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + \mathbf{F}_{\text{link}}^{n+1} = \begin{bmatrix} \mathbf{F}_{\text{vol}}^{\text{str}} \\ \mathbf{F}_{\text{vol}}^{\text{flu}} \end{bmatrix}^{n+1} - \begin{bmatrix} \mathbf{F}_{\text{int}}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{\text{trans}}(\mathbf{U}^{n+1}) \end{bmatrix}$$

$$\mathbf{C}^{n+1} \begin{bmatrix} \dot{\mathbf{Q}}^{n+1} \\ \mathbf{U}^{n+1} \end{bmatrix} = \mathbf{S}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

■ Calcul des forces de liaison

Approche par pénalisation : $\mathbf{F}_{\text{link}}^{n+1} = \mathbf{K} \left(\mathbf{C}^{n+1} \begin{bmatrix} \dot{\mathbf{Q}}^{n+1/2} \\ \mathbf{U}^{n+1} \end{bmatrix} - \mathbf{S}^{n+1} \right)$

✓ Forces de liaisons explicites

✗ Choix des coefficients de pénalité

✗ Impact sur la stabilité de l'intégration temporelle

Approche par dualisation : $\mathbf{F}_{\text{link}}^{n+1} = {}^t \tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda}$

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t \tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}_{\text{vol}}^{\text{str}} \\ \mathbf{F}_{\text{vol}}^{\text{flu}} \end{bmatrix}^{n+1} - \begin{bmatrix} \mathbf{F}_{\text{int}}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{\text{trans}}(\mathbf{U}^{n+1}) \end{bmatrix}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \dot{\mathbf{Q}}^{n+1} \\ \mathbf{U}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

✓ Vérification exacte des liaisons

✓ Pas d'impact sur la stabilité de l'intégration

✗ **Système matriciel à construire et résoudre**

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t \tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \underbrace{\begin{bmatrix} \mathbf{F}_{vol}^{str} \\ \mathbf{F}_{vol}^{flu} \end{bmatrix}^{n+1} - \begin{bmatrix} \mathbf{F}_{int}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{trans}(\mathbf{U}^{n+1}) \end{bmatrix}}_{\mathbf{F}^{n+1}}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t \tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}_{vol}^{str} \\ \mathbf{F}_{vol}^{flu} \end{bmatrix}^{n+1} - \underbrace{\begin{bmatrix} \mathbf{F}_{int}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{trans}(\mathbf{U}^{n+1}) \end{bmatrix}}_{\mathbf{F}^{n+1}}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

Centre de coût n°1 = Boucle élémentaire

- Calcul des contraintes (relation de comportement, loi d'état)
- Calcul des flux
- *Hétérogénéité des coûts élémentaires*

Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t\tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}_{vol}^{str} \\ \mathbf{F}_{vol}^{flu} \end{bmatrix}^{n+1} - \underbrace{\begin{bmatrix} \mathbf{F}_{int}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{trans}(\mathbf{U}^{n+1}) \end{bmatrix}}_{\mathbf{F}^{n+1}}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_P(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

Centre de coût n°1 = Boucle élémentaire

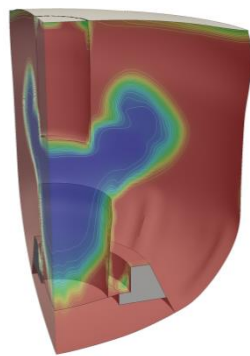
- Calcul des contraintes (relation de comportement, loi d'état)
- Calcul des flux
- *Hétérogénéité des coûts élémentaires*

Centre de coût n°2 = Ecriture des liaisons

- Détection des contacts
- Ecriture des relations d'interaction fluide-structure
- Tris spatiaux, calculs d'intersection et d'inclusion



Crash avec auto-contact



Explosion en cuve avec structures immergées

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t\tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}_{vol}^{str} \\ \mathbf{F}_{vol}^{flu} \end{bmatrix}^{n+1} - \underbrace{\begin{bmatrix} \mathbf{F}_{int}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{trans}(\mathbf{U}^{n+1}) \end{bmatrix}}_{\mathbf{F}^{n+1}}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

Centre de coût n°1 = Boucle élémentaire

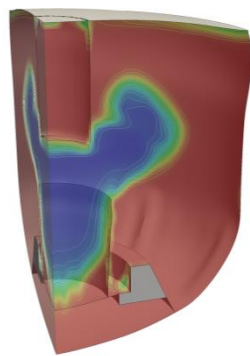
- Calcul des contraintes (relation de comportement, loi d'état)
- Calcul des flux
- *Hétérogénéité des coûts élémentaires*

Centre de coût n°2 = Ecriture des liaisons

- Détection des contacts
- Ecriture des relations d'interaction fluide-structure
- Tris spatiaux, calculs d'intersection et d'inclusion



Crash avec auto-contact



Explosion en cuve avec structures immergées

Centre de coût n°3 = Calcul des forces de liaison

- Condensation sur les multiplicateurs

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} {}^t\tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} \mathbf{F}^{n+1} - \tilde{\mathbf{S}}^{n+1}$$

$$\Leftrightarrow \mathbf{H}^{n+1} \boldsymbol{\Lambda} = \mathbf{B}^{n+1}$$

- Résolution et calcul des forces de liaison

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t\tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}_{vol}^{str} \\ \mathbf{F}_{vol}^{flu} \end{bmatrix}^{n+1} - \underbrace{\begin{bmatrix} \mathbf{F}_{int}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{trans}(\mathbf{U}^{n+1}) \end{bmatrix}}_{\mathbf{F}^{n+1}}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

Centre de coût n°1 = Boucle élémentaire

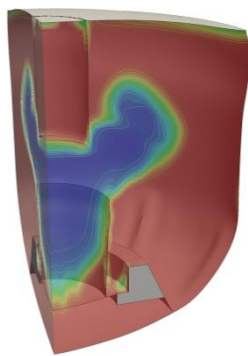
- Calcul des contraintes (relation de comportement, loi d'état)
- Calcul des flux
- *Hétérogénéité des coûts élémentaires*

Centre de coût n°2 = Ecriture des liaisons

- Détection des contacts
- Ecriture des relations d'interaction fluide-structure
- Tris spatiaux, calculs d'intersection et d'inclusion



Crash avec auto-contact



Explosion en cuve avec structures immergées

Centre de coût n°3 = Calcul des forces de liaison

- Condensation sur les multiplicateurs

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} {}^t\tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} \mathbf{F}^{n+1} - \tilde{\mathbf{S}}^{n+1}$$

$$\Leftrightarrow \mathbf{H}^{n+1} \boldsymbol{\Lambda} = \mathbf{B}^{n+1}$$

- Résolution et calcul des forces de liaison

■ Système discret

$$\begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} + {}^t\tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}_{vol}^{str} \\ \mathbf{F}_{vol}^{flu} \end{bmatrix}^{n+1} - \underbrace{\begin{bmatrix} \mathbf{F}_{int}(\mathbf{Q}^{n+1}) \\ \mathbf{F}_P(\mathbf{U}^{n+1}) + \mathbf{F}_{trans}(\mathbf{U}^{n+1}) \end{bmatrix}}_{\mathbf{F}^{n+1}}$$

$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \ddot{\mathbf{Q}}^{n+1} \\ \dot{\mathbf{U}}^{n+1} \end{bmatrix} = \tilde{\mathbf{S}}^{n+1}$$

$$[\rho]^{n+1} = [\rho]^n + \mathbf{F}_\rho(\mathbf{U})$$

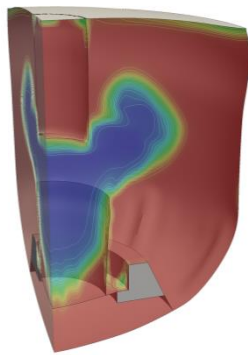
$$[\mathbf{E}]^{n+1} = [\mathbf{E}]^n + \mathbf{F}_E(\mathbf{U})$$

Centre de coût n°2 = Écriture des liaisons

- Détection des contacts
- Écriture des relations d'interaction fluide-structure
- Tris spatiaux, calculs d'intersection et d'inclusion



Crash avec auto-contact



Explosion en cuve avec structures immergées

Centre de coût n°1 = Boucle élémentaire

- Calcul des contraintes (relation de comportement, loi d'état)
- Calcul des flux
- Hétérogénéité des coûts élémentaires

Centre de coût n°3 = Calcul des forces de liaison

- Condensation sur les multiplicateurs

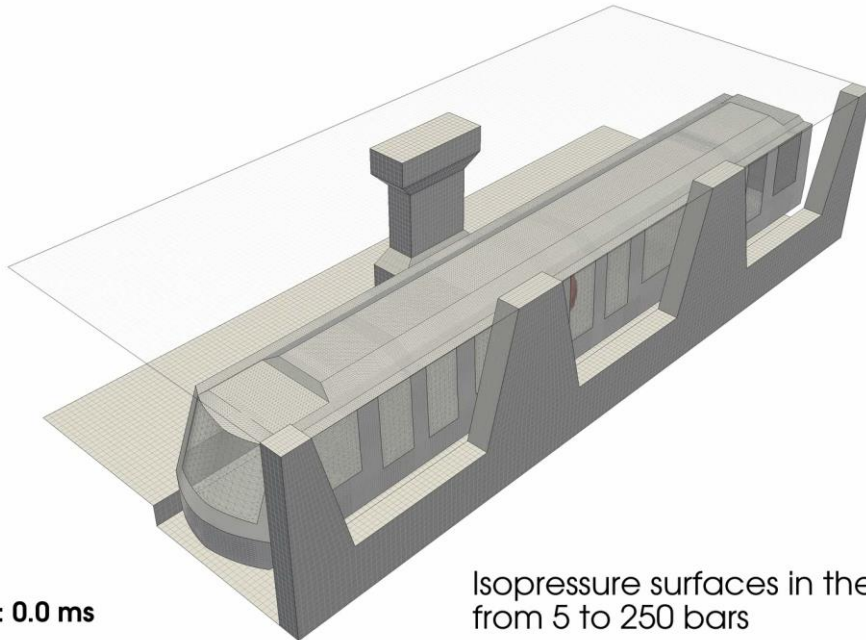
$$\tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} {}^t\tilde{\mathbf{C}}^{n+1} \boldsymbol{\Lambda} = \tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} \mathbf{F}^{n+1} - \tilde{\mathbf{S}}^{n+1}$$

$$\Leftrightarrow \mathbf{H}^{n+1} \boldsymbol{\Lambda} = \mathbf{B}^{n+1}$$

- Résolution et calcul des forces de liaison

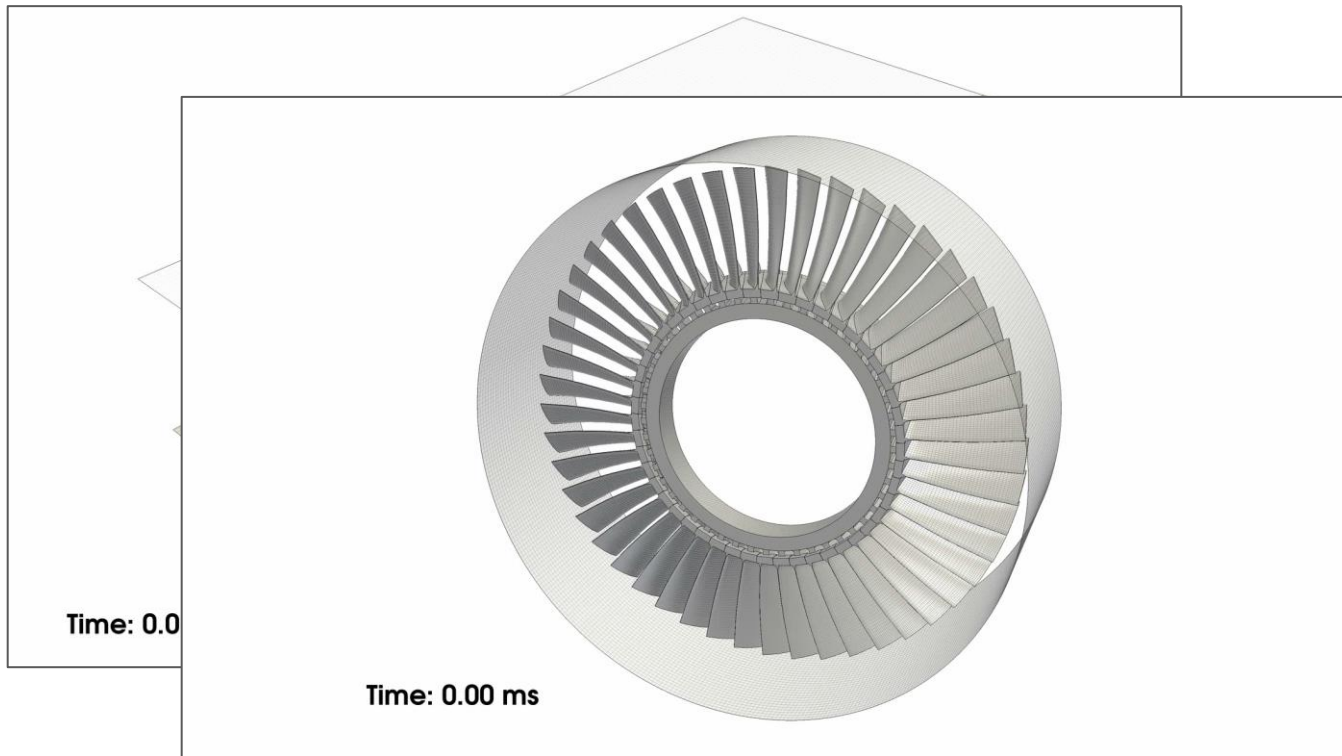
■ Conclusions sur l'algorithme

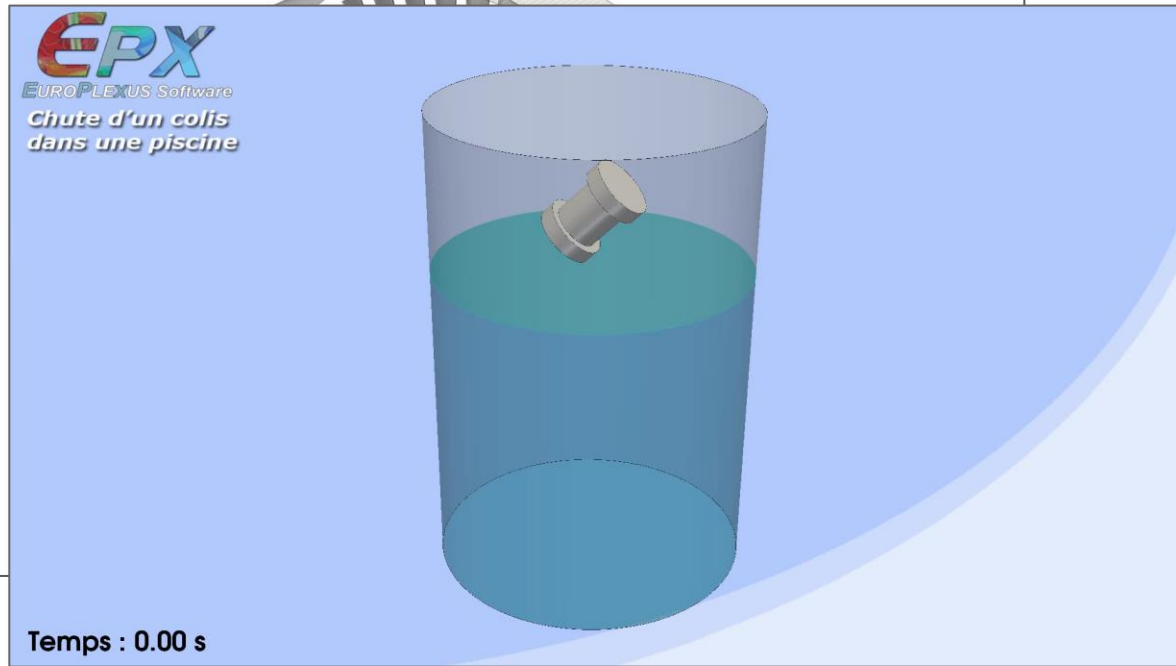
- EPX = du classique : intégration explicite, calculs élémentaires, écriture des liaisons cinématiques
- ...et du moins classique : calcul dual des forces de liaison
- Stratégie parallèle (très) spécifique



Time: 0.0 ms

Isopressure surfaces in the fluid
from 5 to 250 bars





EPX
EUROPLEXUS Software

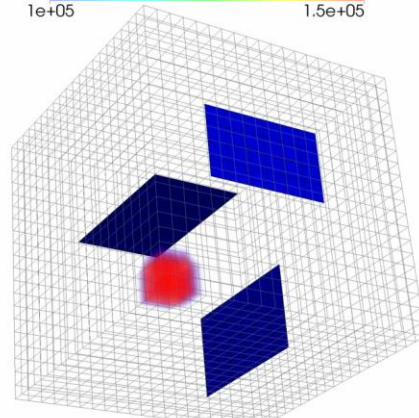
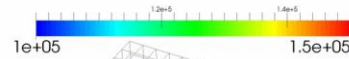
Chute d'un
dans une p

Time: 0.0

Temps : 0.0

Volume rendering

Pressure (Pa)

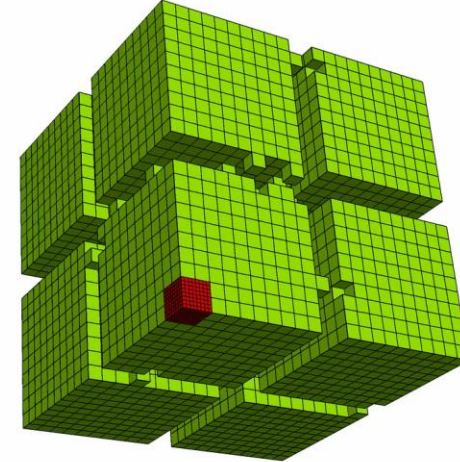


Displacement (m)

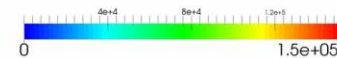


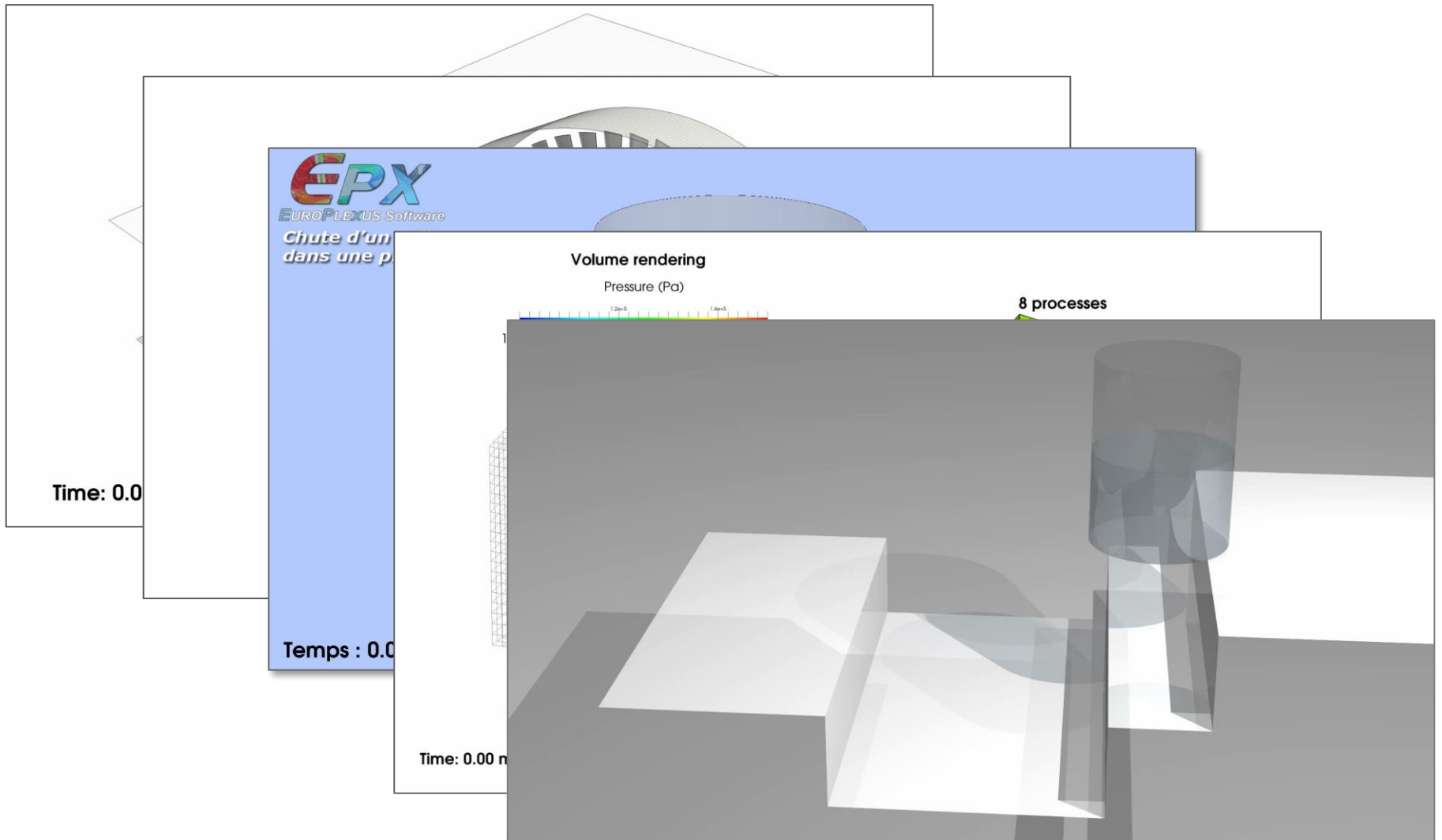
Time: 0.00 ms

8 processes



Pressure (Pa)





■ Pourquoi la mémoire distribuée ?

- *Hardware* dominant en 2007 : clusters de nœuds mono ou bi-processeurs
- De l'ordre de 4 cœurs par nœud
- Approche distribuée largement dominante
- Seule approche permettant dans tous les cas d'adresser de grands nombres de cœurs
- Parallélisme massif au prix de l'écriture des communications MPI

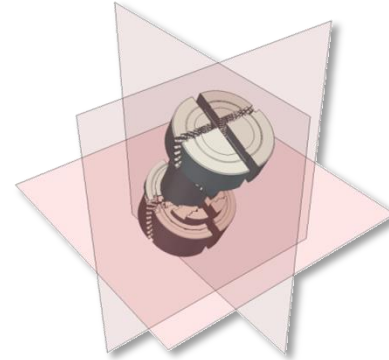
PARALLÉLISME DOMINANT À MÉMOIRE DISTRIBUÉE

■ Pourquoi la mémoire distribuée ?

- *Hardware* dominant en 2007 : clusters de nœuds mono ou bi-processeurs
- De l'ordre de 4 cœurs par nœud
- Approche distribuée largement dominante
- Seule approche permettant dans tous les cas d'adresser de grands nombres de cœurs
- Parallélisme massif au prix de l'écriture des communications MPI

■ Mise en œuvre dans EPX

- Décomposition de domaine pour distribuer le centre de coût n°1 (*boucle élémentaire*)
- Logique SPMD et réutilisation de la structure des sous-domaines existante (*passage global-local*)



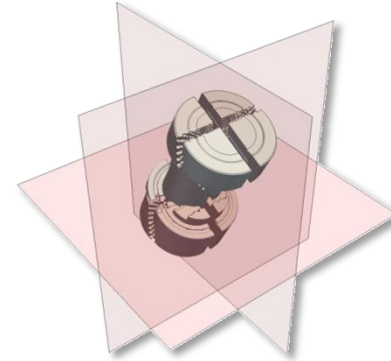
Décomposition automatique par Recursive Orthogonal Bisection

Pourquoi la mémoire distribuée ?

- Hardware dominant en 2007 : clusters de nœuds mono ou bi-processeurs
- De l'ordre de 4 cœurs par nœud
- Approche distribuée largement dominante
- Seule approche permettant dans tous les cas d'adresser de grands nombres de cœurs
- Parallélisme massif au prix de l'écriture des communications MPI

Mise en œuvre dans EPX

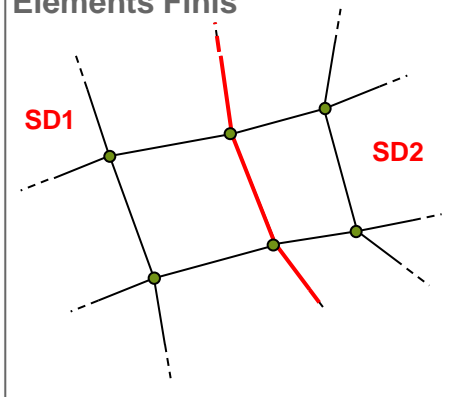
- Décomposition de domaine pour distribuer le centre de coût n°1 (boucle élémentaire)
- Logique SPMD et réutilisation de la structure des sous-domaines existante (passage global-local)



Décomposition automatique par Recursive Orthogonal Bisection

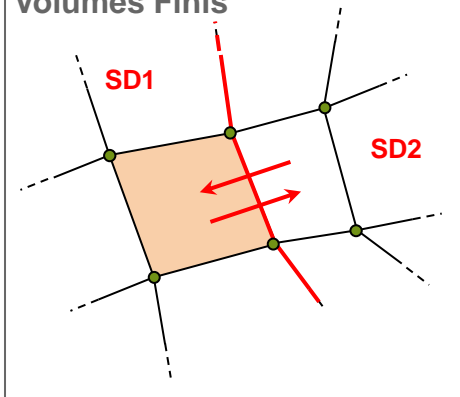
Formalisation et construction des frontières pour toutes les formulations

Eléments Finis

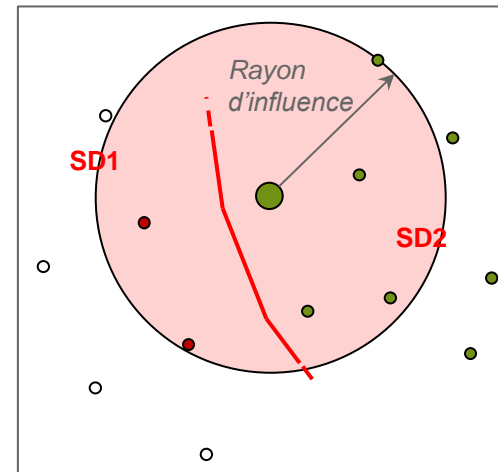


Sommation des forces nodales

Volumes Finis



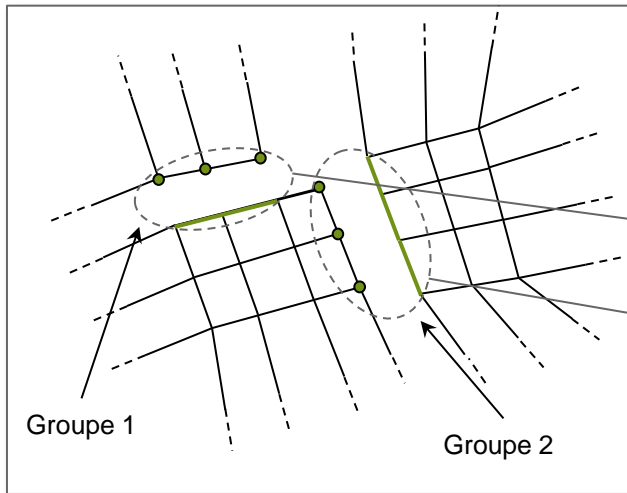
Cellules fantômes pour les flux Volumes Finis



Particules remote pour le voisinage des particules locales

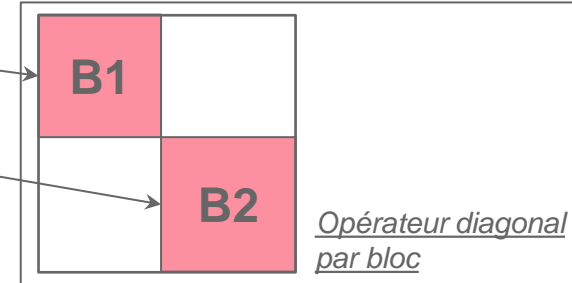
■ Calcul générique des forces de liaison

■ Structure de l'opérateur de condensation



Expression de l'opérateur

$$\mathbf{H}^{n+1} = \tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} {}^t\tilde{\mathbf{C}}^{n+1} + \text{Matrice de masse diagonale}$$

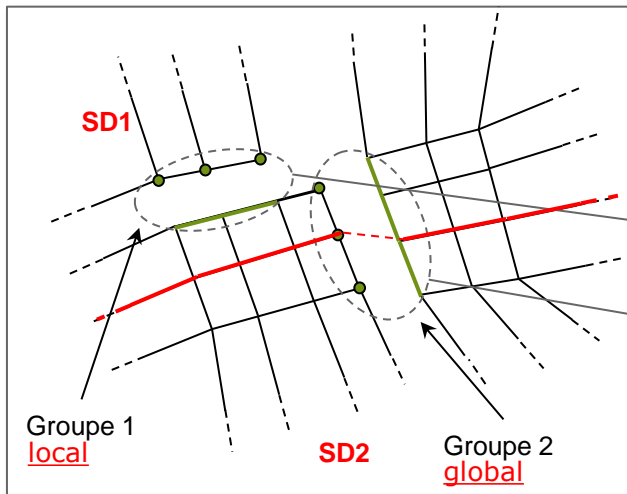


■ Structure de données pour les liaisons

- Liste chaînée à deux niveaux = liste de groupes de liaisons couplées
- Ajout/suppression dynamiques de liaisons en cours de calcul, avec identification des couplages
 - ✓ Identification a priori des blocs dans l'opérateur de condensation
 - ✗ Coût de parcours de la structure des liaisons (non-locale en cache)

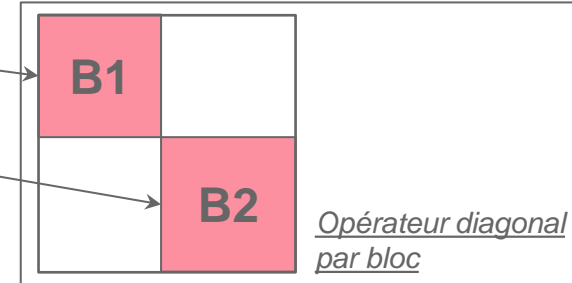
■ Calcul générique des forces de liaison

■ Structure de l'opérateur de condensation



Expression de l'opérateur

$$\mathbf{H}^{n+1} = \tilde{\mathbf{C}}^{n+1} \begin{bmatrix} \mathbf{M}_S \\ \mathbf{M}_F^{n+1} \end{bmatrix}^{-1} {}^t \tilde{\mathbf{C}}^{n+1} + \text{Matrice de masse diagonale}$$



■ Structure de données pour les liaisons

- Liste chaînée à deux niveaux = liste de groupes de liaisons couplées
- Ajout/suppression dynamiques de liaisons en cours de calcul, avec identification des couplages
 - ☑ *Identification a priori des blocs dans l'opérateur de condensation*
 - ☒ *Coût de parcours de la structure des liaisons (non-locale en cache)*

■ Avec décomposition de domaines

- Groupes locaux traités localement : parallélisme (non-équilibré...)
- Problème(s) global(aux) à construire et résoudre
 - ☒ *Surcoût en communication pour la construction*
 - ☒ *Tâche de résolution majoritairement séquentielle*

1^e contrainte : ne pas assembler d'opérateur global

Approche itérative : $\tilde{\Lambda}_k \rightarrow \mathbf{R}_k = \mathbf{H}^{n+1} \tilde{\Lambda}_k - \mathbf{B}^{n+1}$

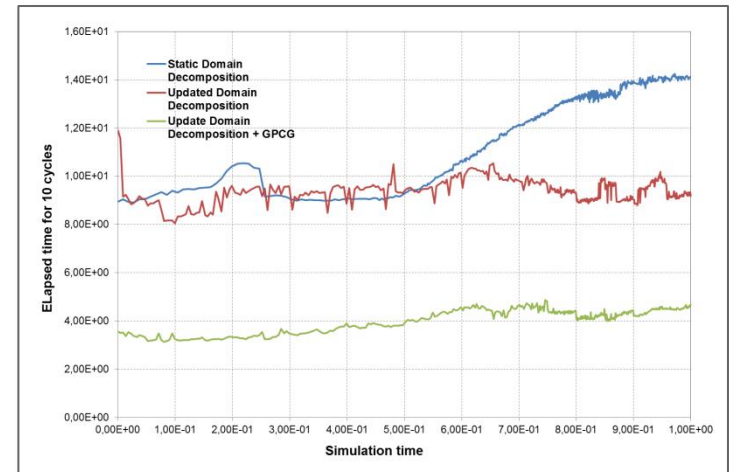
Evaluation du résidu :

① $\mathbf{F}_k = {}^t \tilde{\mathbf{C}}_g^{n+1} \tilde{\Lambda}_k$ *Calcul des forces sur les ddls locaux et remote sur le SD où chaque liaison est écrite*

② Transfert et sommation des forces sur les ddls remote \mathbf{F}_k^S

③ $\mathbf{R}_k = \tilde{\mathbf{C}}_g^{n+1} \mathbf{M}^{-1} \mathbf{F}_k^S - \mathbf{B}^{n+1}$ *Résidu partiel sur chaque SD*

Algorithme de résolution : Gradient Conjugué



2^e contrainte : converger rapidement

Préconditionnement indispensable

Résolution de problèmes locaux avec remote

3^e contrainte : optimiser les communications

Communications supplémentaires durant les itérations du solveur global

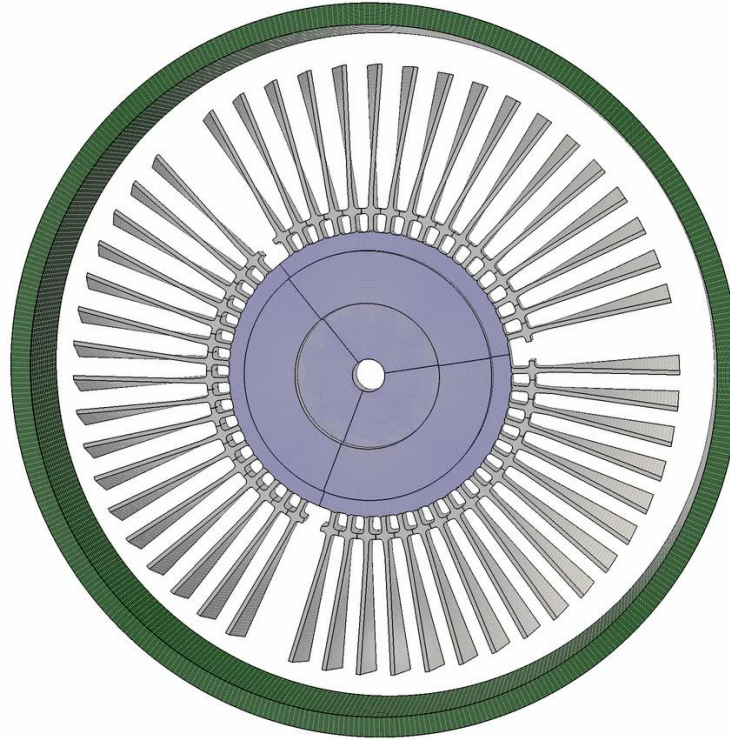


Front view

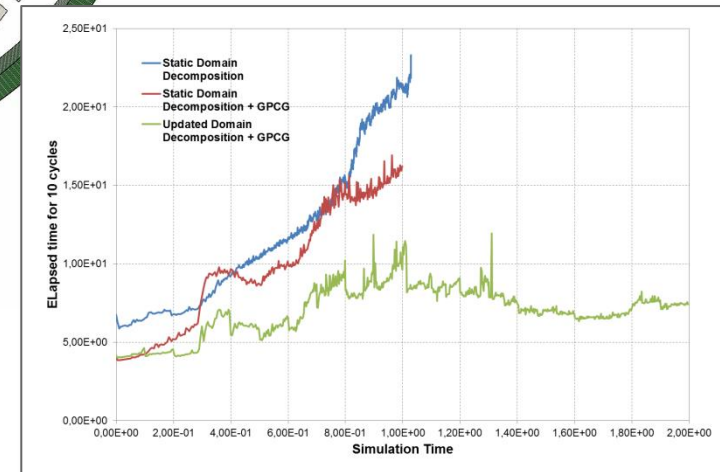
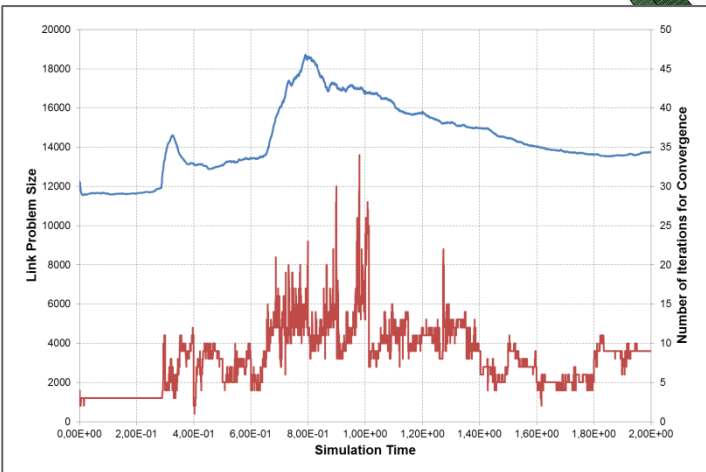
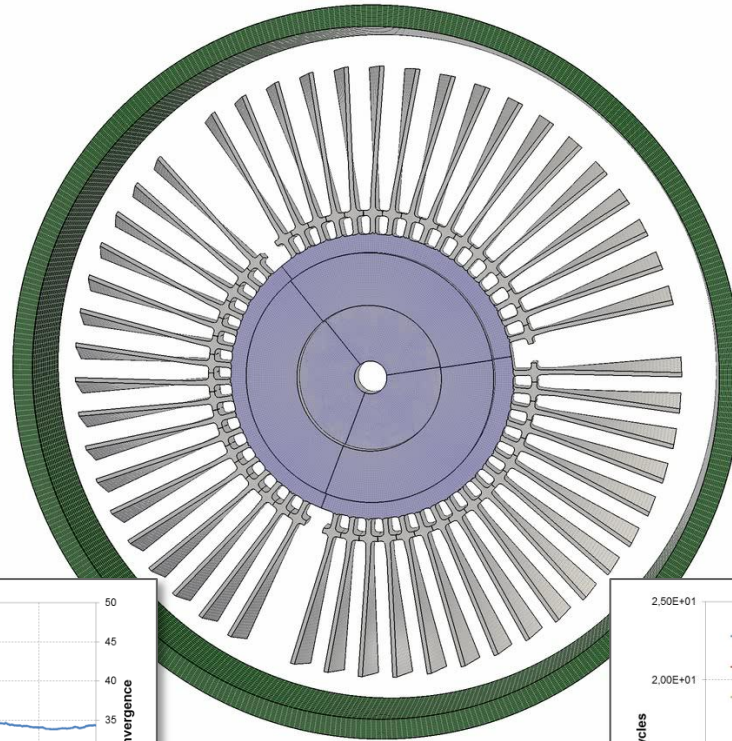


Rear view

PARALLÉLISME À MÉMOIRE DISTRIBUTÉE OPTIMISATIONS ET VERROUS



PARALLÉLISME À MÉMOIRE DISTRIBUÉE OPTIMISATIONS ET VERROUS



IMPLÉMENTATION DU RUNTIME À MÉMOIRE PARTAGÉE KAAPI (INRIA/LIG - MOAIS)

■ Problématique principale

- Les centres de coût 1 et 2 ont des coûts par itération hétérogènes
- Mise en défaut des approches statiques de distribution du travail sur les *threads*

■ Avantages de la mémoire partagée (à l'intérieur d'un sous-domaine)

- Pas de redondance des données
- Exploitation des ressources en limitant la tension sur les liaisons globales et les communications en général

■ Difficultés principales

- Loi d'Amdahl
- Boucles *memory bound* sur architectures NUMA

■ Approche par vol de travail avec KAAPI

- Programmation DFG
- Implémentation FORTRAN aisée (boucles parallèles en particulier)
- Performance équivalente à Intel TBB ou Cilk avec un environnement plus simple

■ Investissement CEA/INRIA en complément du projet RePDyn

2011 : Stage de Master 2 de D. Leone
Interface FORTRAN pour KAAPI dans EPX

2011/2012 : Contrat F. Lementec (suivi T. Gautier)
Consolidation de l'implémentation et définition du cadre générique d'intégration de KAAPI dans EPX

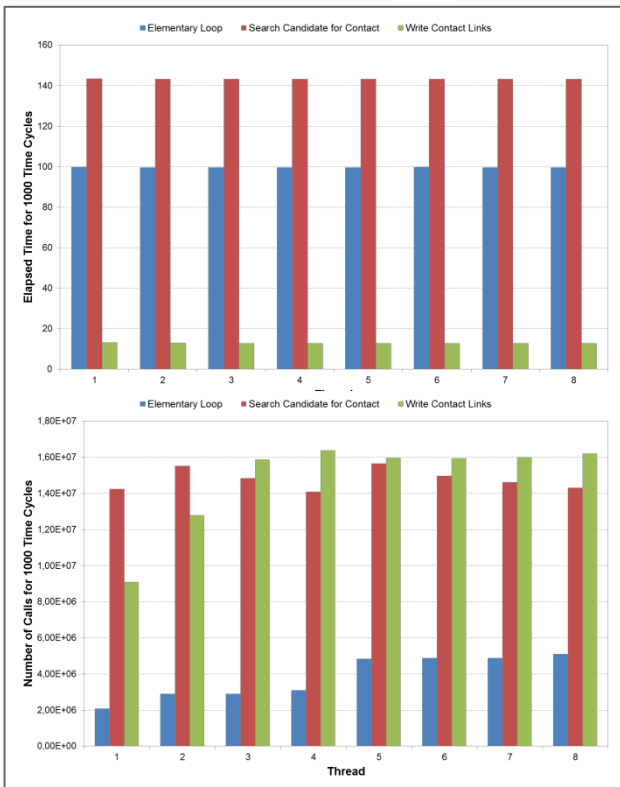
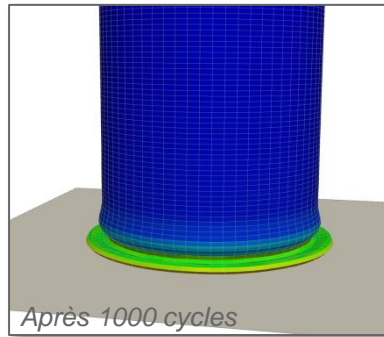
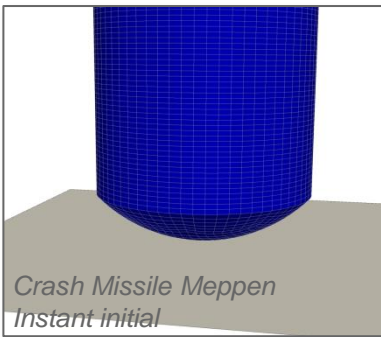
■ Projet RepDyn (ANR-COSI-09-011)

AGENCE NATIONALE DE LA RECHERCHE

ANR

Calcul parallèle en dynamique
rapide des fluides et des structure

IMPLÉMENTATION DU RUNTIME À MÉMOIRE PARTAGÉE KAAPI (INRIA/LIG - MOAIS)



83% du travail accéléré par KAAPI
Speed-up 6/8 (memory bound)

■ Approche par vol de travail avec KAAPI

- Programmation DFG
- Implémentation FORTRAN aisée (boucles parallèles en particulier)
- Performance équivalente à Intel TBB ou Cilk avec un environnement plus simple

■ Investissement CEA/INRIA en complément du projet RePDyn

2011 : Stage de Master 2 de D. Leone
Interface FORTRAN pour KAAPI dans EPX

2011/2012 : Contrat F. Lementec (suivi T. Gautier)
Consolidation de l'implémentation et définition du cadre générique d'intégration de KAAPI dans EPX

■ Projet RepDyn (ANR-COSI-09-011)

AGENCE NATIONALE DE LA RECHERCHE
ANR Calcul parallèle en dynamique rapide des fluides et des structure

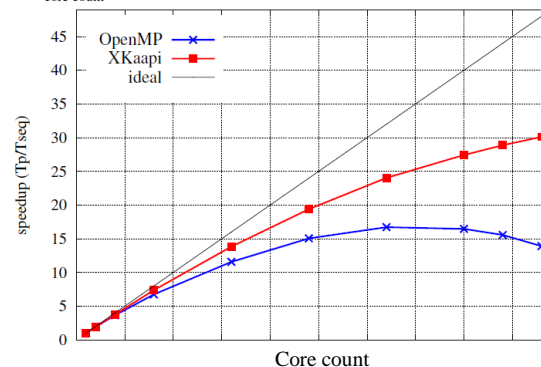
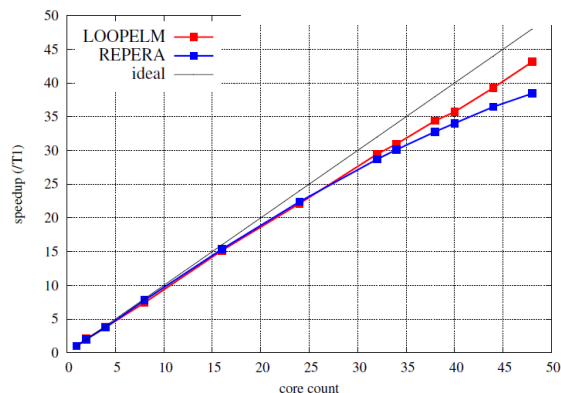
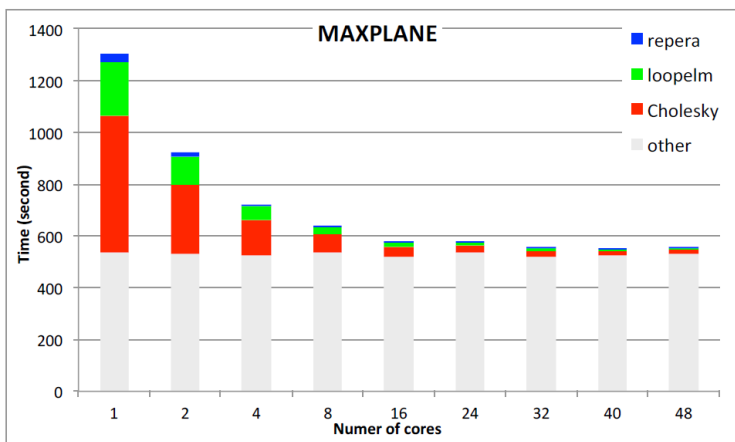
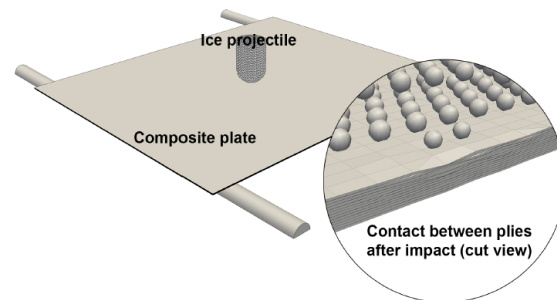
■ Implémentation prototype en programmation DFG

■ Cas de démonstration multi-contacts

Main characteristics

- ◆ Most of the time in sparse Cholesky factorization

AMD Many Cours, 2.2GHz, 48 cores, 256GB main



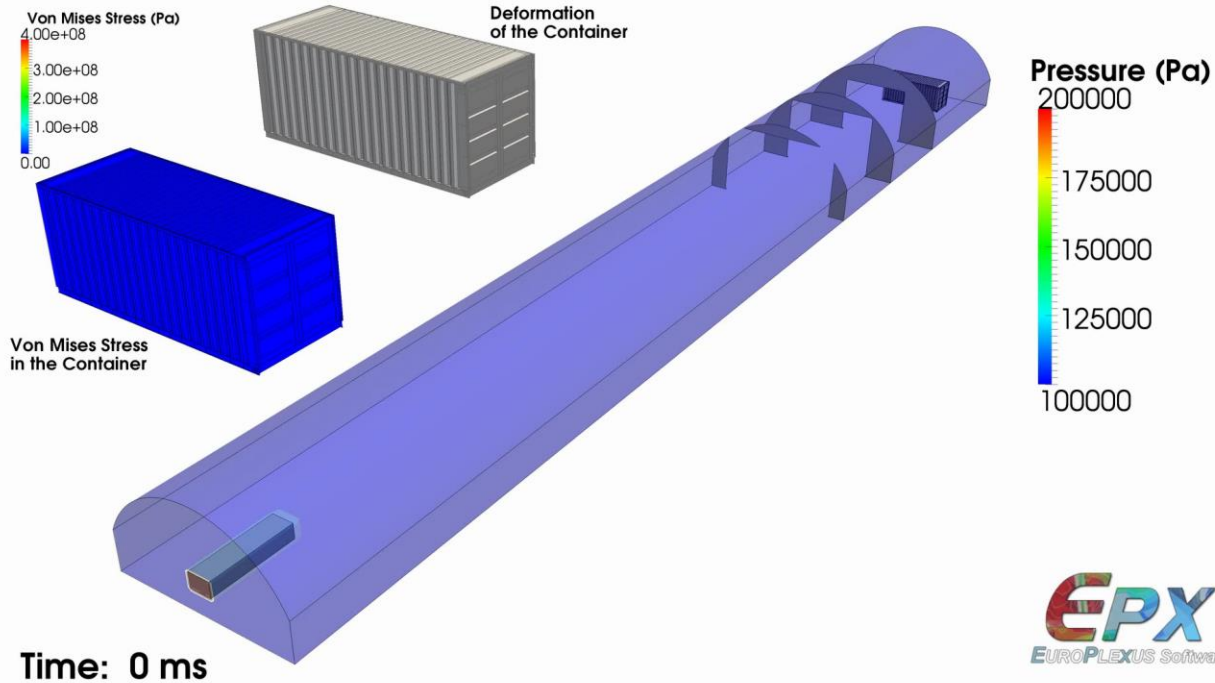
* Thierry Gautier, Fabien Le Mentec, Vincent Faucher, Bruno Raffin.
 X-kaapi: A Multi Paradigm Runtime for Multicore Architectures. Workshop P2PS
 ICPP 2013: 728-735
 * V. Faucher, P. Galon, A. Beccantini, F.Crouzet, F. Debaud, T. Gautier,
 Hybrid parallel strategy for the simulation of fast transient accidental situations at reactor scale,
 Annals of Nuclear Energy, 2014

MISE EN ŒUVRE À GRANDE ÉCHELLE DE L'APPROCHE HYBRIDE MPI/KAAPI

PRACE Preparatory Access 0624 (2012)

- Machine cible : TGCC/Curie *Fat Nodes* (32 cœurs, 128 Go RAM, Intel Westmere)
- Objectif : calcul parallèle sur 1024 pour des simulations fortement couplées en Interaction Fluide-Structure

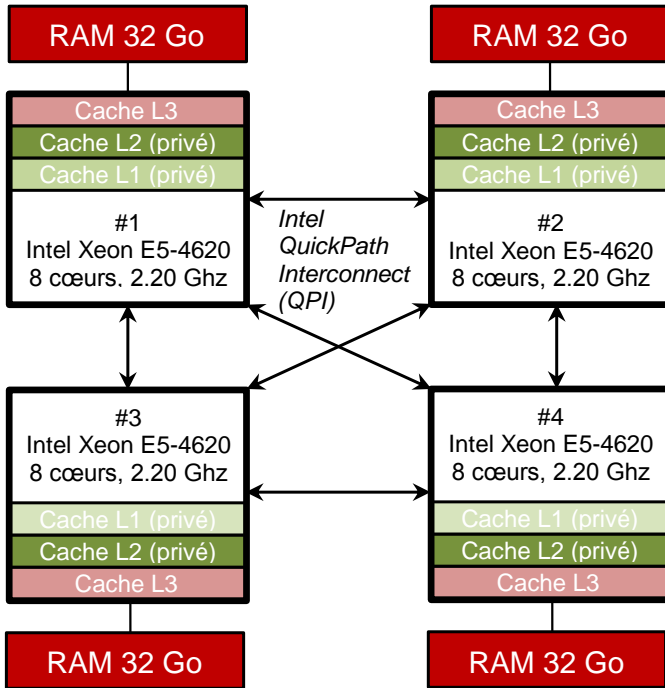
Onde de Choc sur un Container Métallique (modèle NTNU/JRC Ispra)



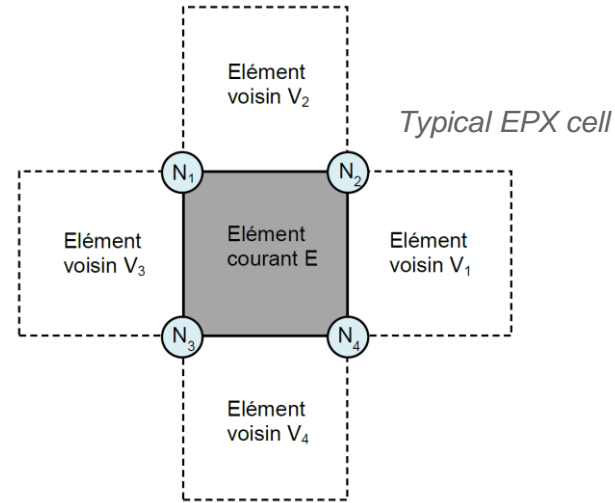
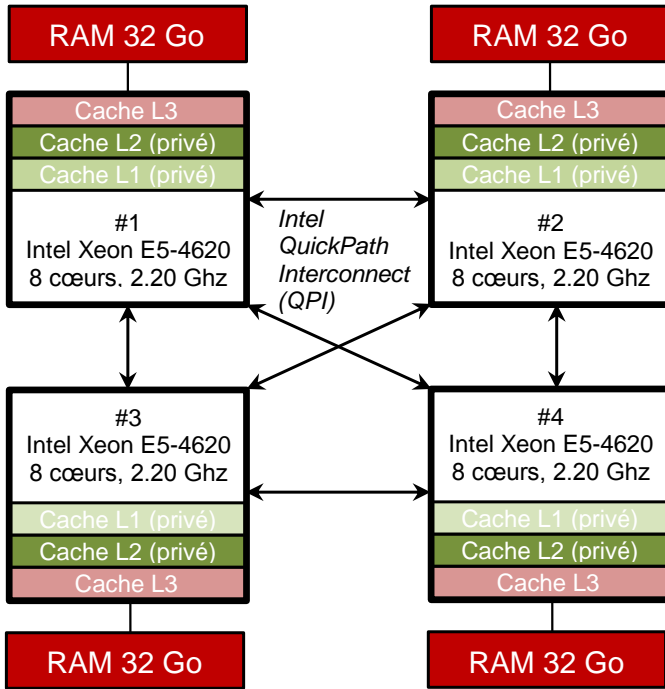
Retour d'expérience

- Un exécutable mal placé et un mois de débogage inutile, un code modérément au point
- Des modèles de taille relativement petite (~ 2M de mailles) : gestion reportée des modèles de grande taille
- Des boucles *memory bound* et des facteurs NUMA importants : utilisation de KAAPI limitée à 8 threads

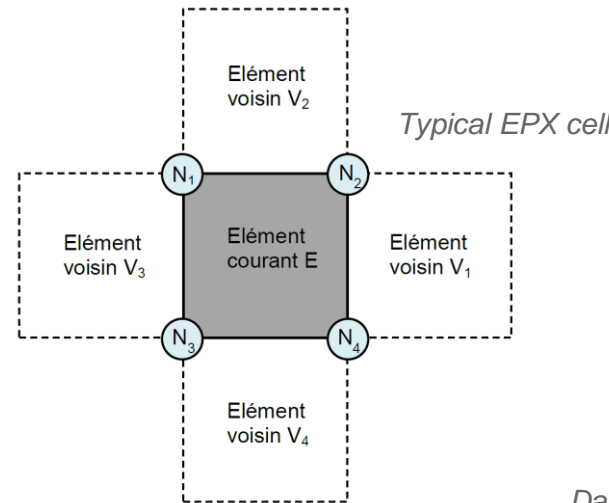
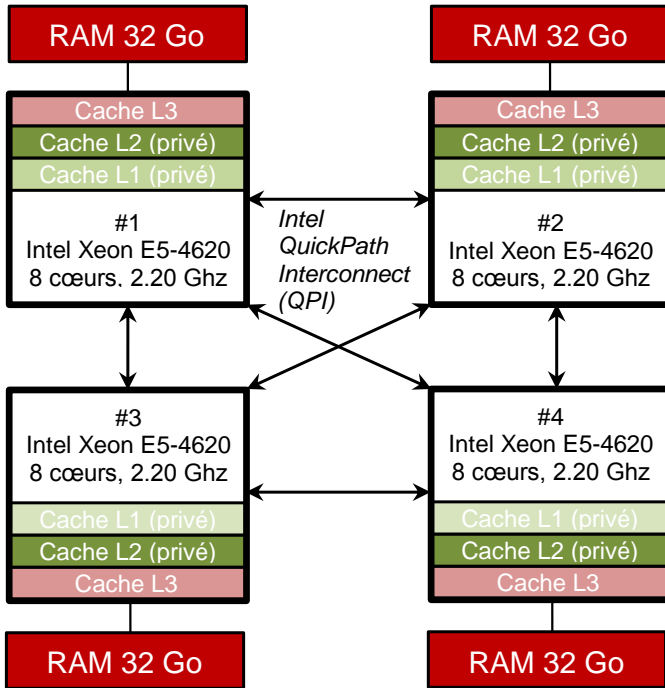
Target architecture



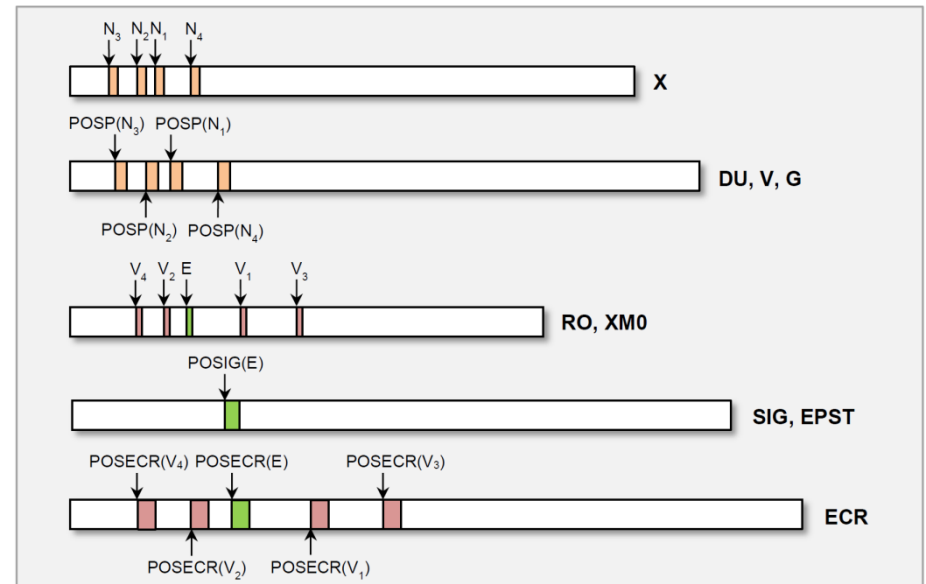
Target architecture



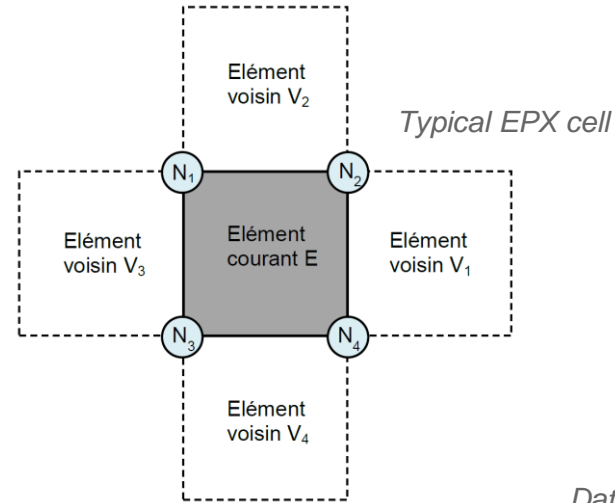
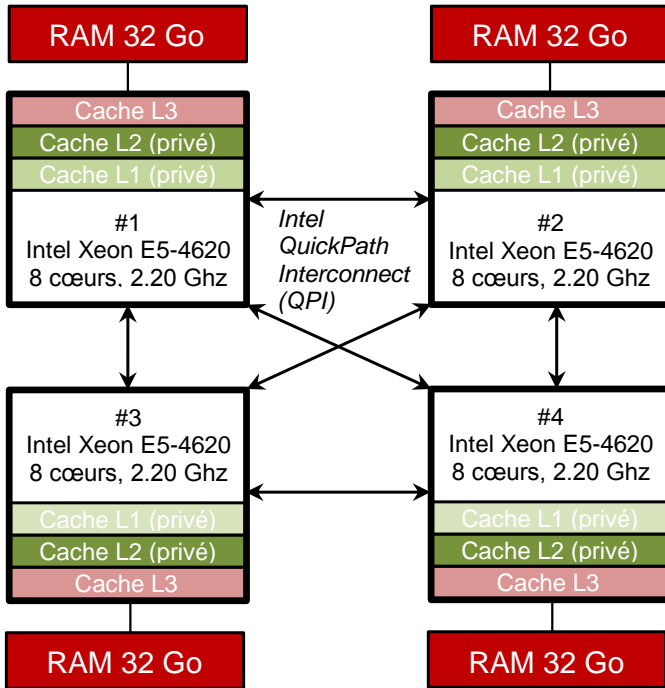
Target architecture



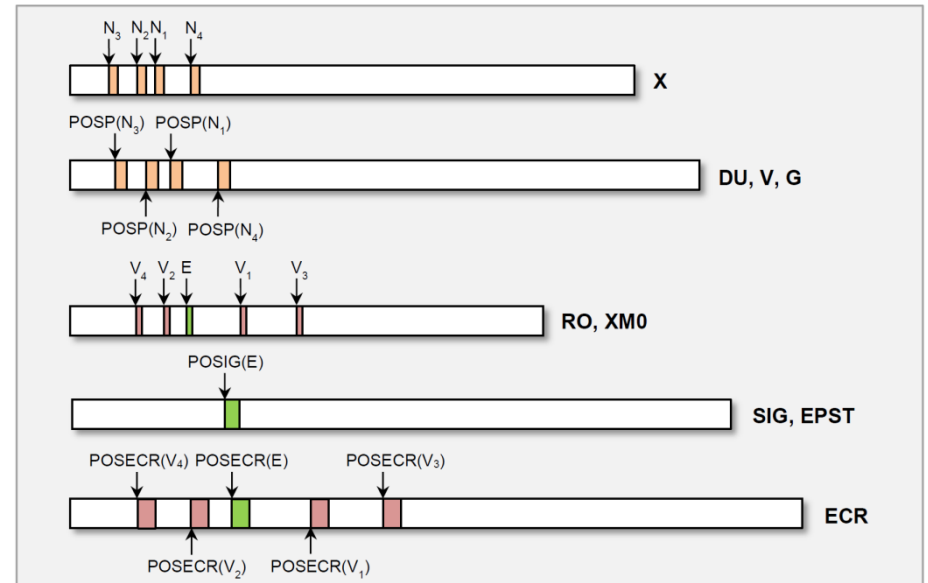
Data placement in memory



Target architecture



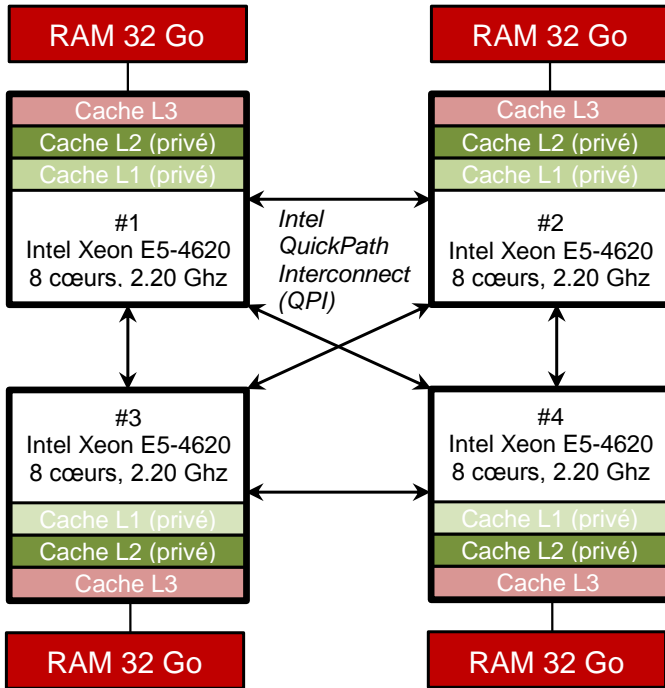
Data placement in memory



Problematics

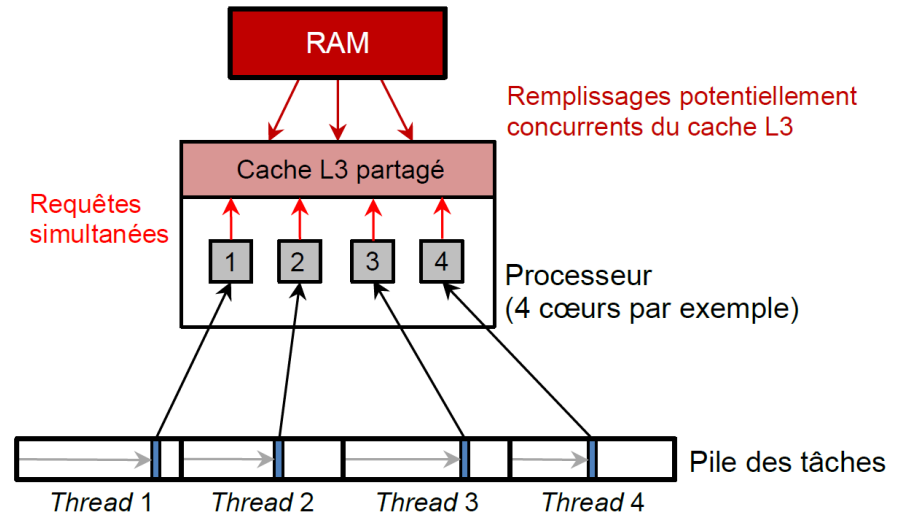
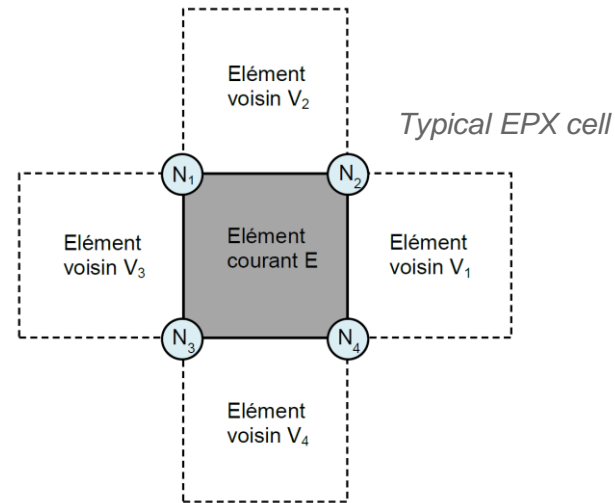
- Global data structure inherited from past choices
- Lack of scalability for the elementary loop using threads on several sockets

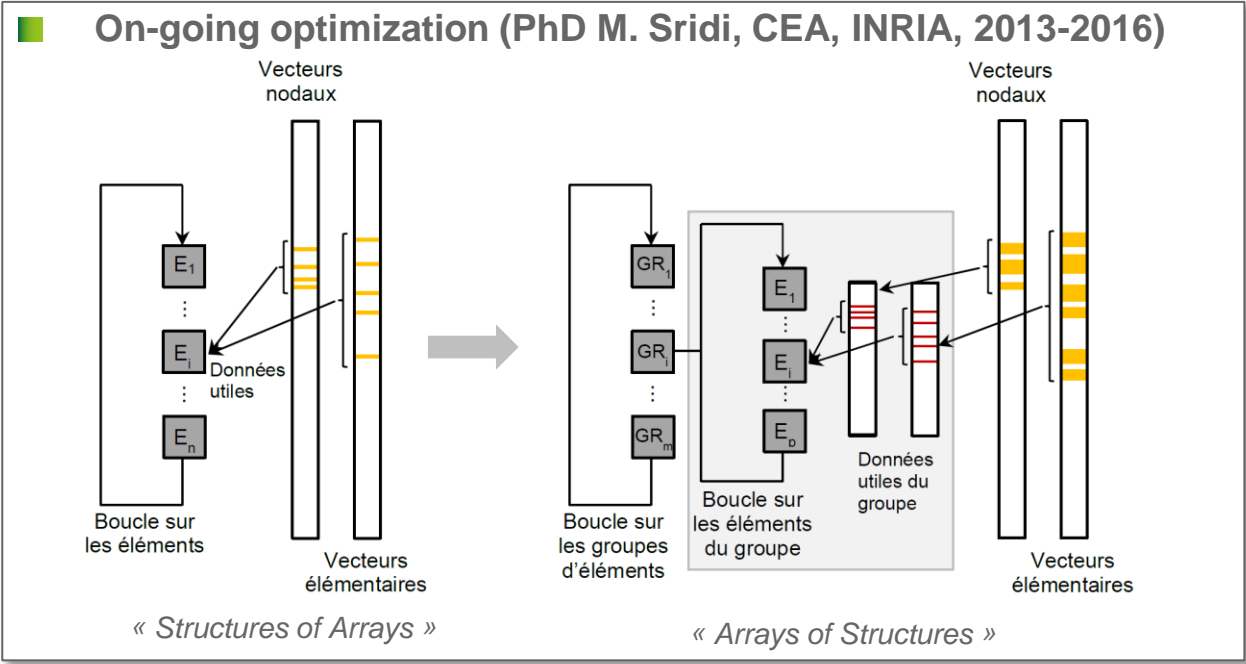
Target architecture



Problematics

- Global data structure inherited from past choices
- Lack of scalability for the elementary loop using threads on several sockets

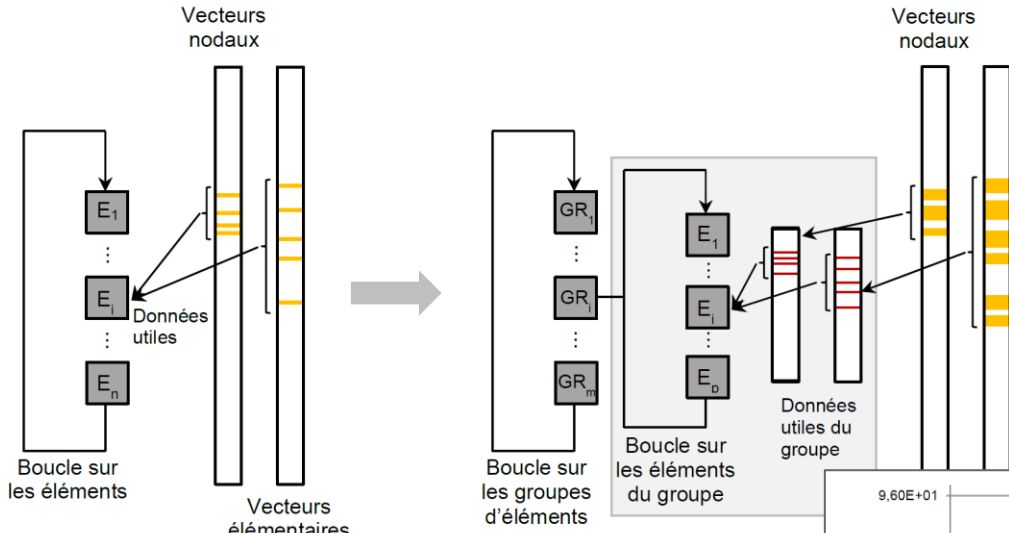




R&D COLLABORATIVE CEA/INRIA

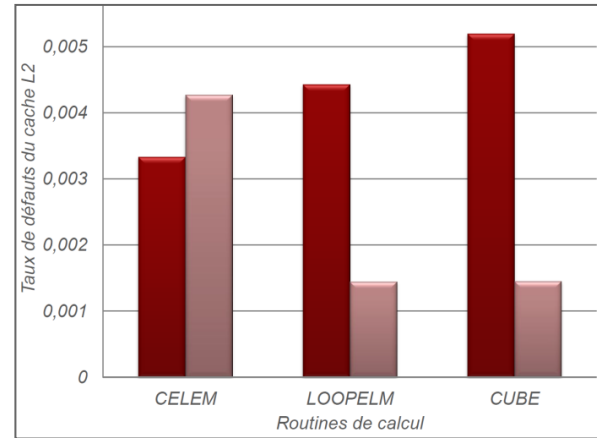
PLACEMENT MÉMOIRE ET PARALLÉLISME À 2 NIVEAUX

On-going optimization (PhD M. Sridi, CEA, INRIA, 2013-2016)

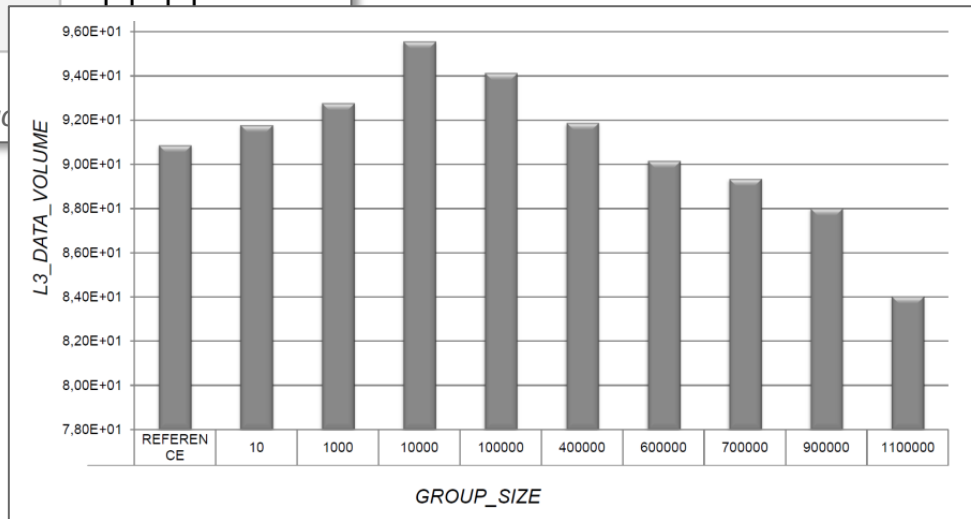


« Structures of Arrays »

« Arrays of Structures »



L2-Cache misses reduction

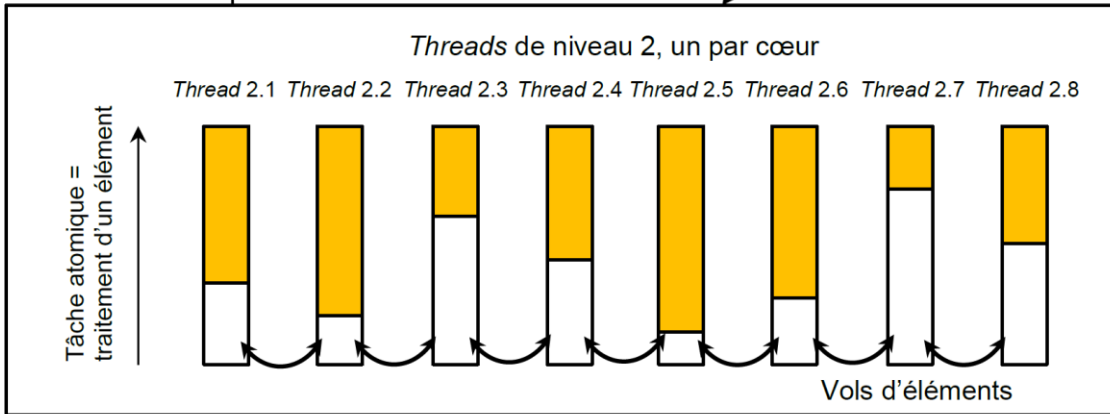
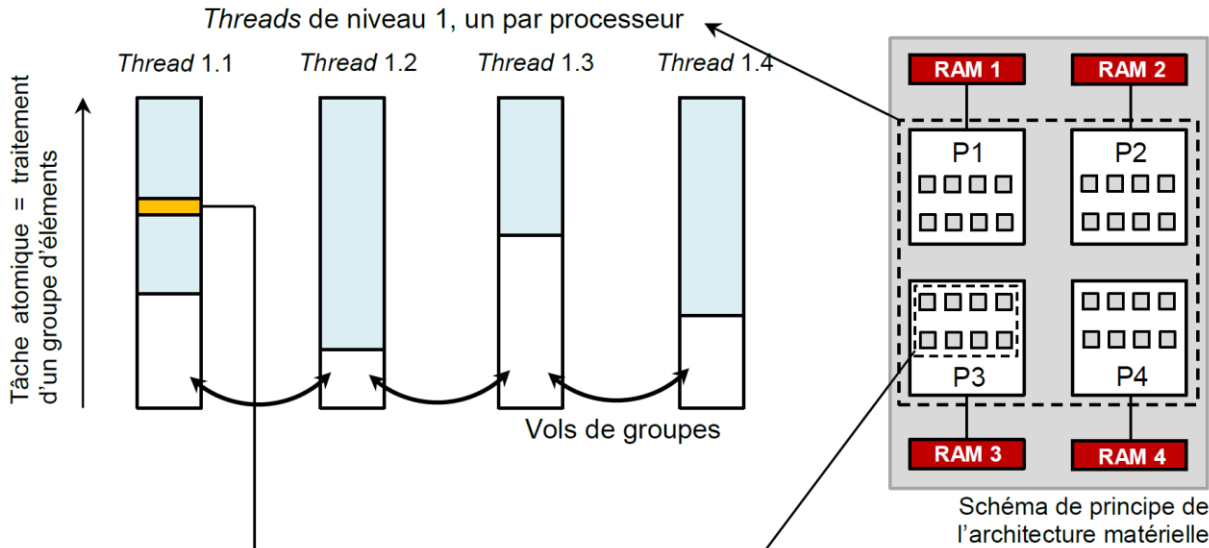


L3-Cache filling improvement

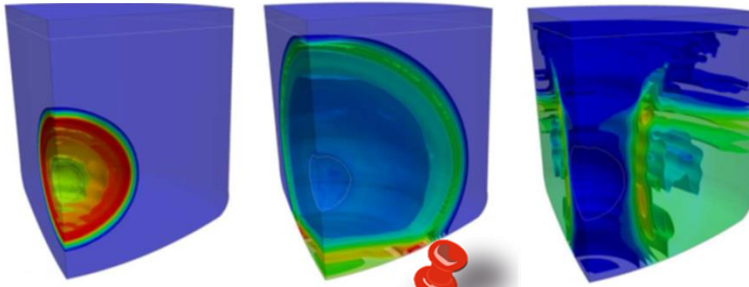
R&D COLLABORATIVE CEA/INRIA

PLACEMENT MÉMOIRE ET PARALLÉLISME À 2 NIVEAUX

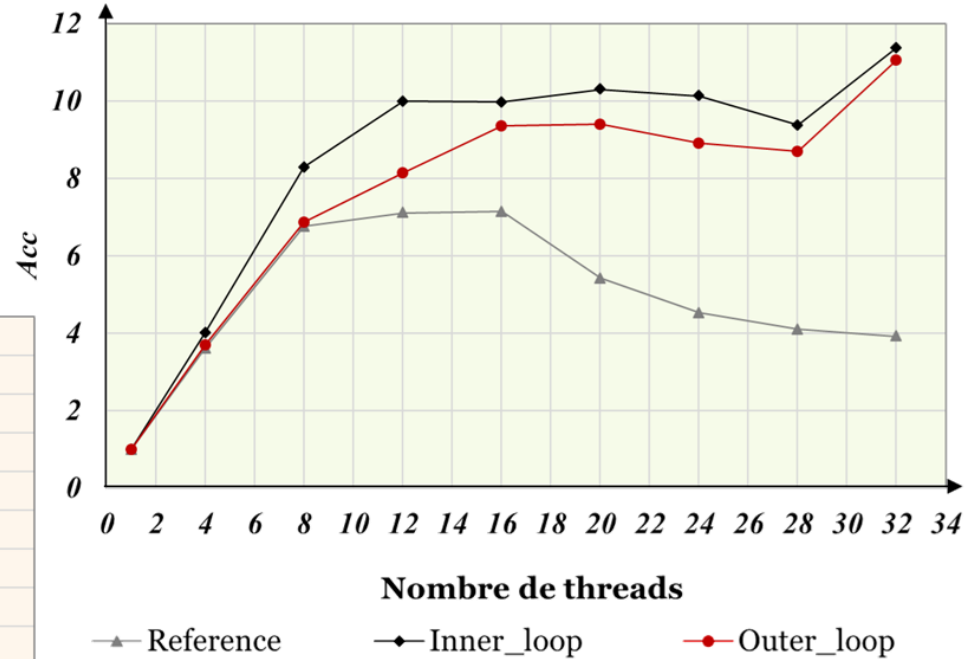
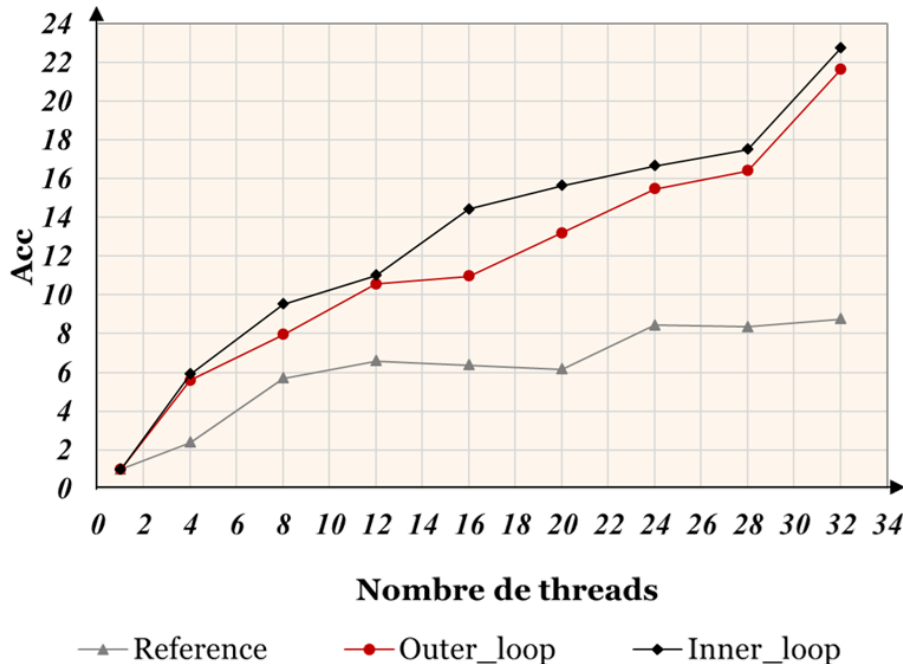
On-going optimization (PhD M. Sridi, CEA, INRIA, 2013-2016)



XKAAPI – v3, 2015, January
T. Gautier, INRIA



Simulation MARA_big



Simulation MARA_mdm

Integration of KAAPI solutions to libOMP

Goal: make research developments available in production quality runtime

- ◆ Intel Runtime selected for several reasons
 - target of icc and ifort compiler
 - good algorithms and data structures basis
 - distributed work queues + (naive) work stealing algorithm
 - scalable barrier algorithm
 - small overhead in task creation
 - integration of a lot of loop schedulers (~ 15)
 - initial support for libGOMP / GCC application
 - initial support for OMPT (interface for integrating tracing tools with OpenMP runtime)

Kaapi solutions

- ◆ task representation and task scheduling algorithm
 - adaptive task
 - affinity scheduler
- ◆ tracing tools

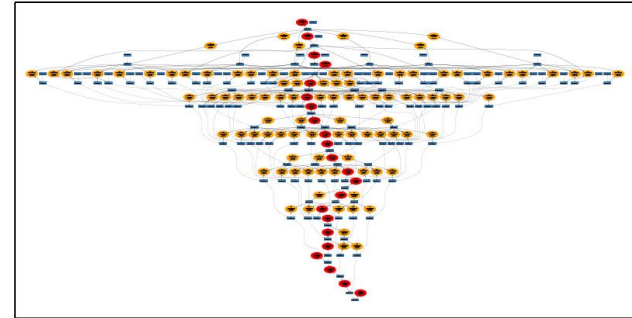
libKOMP: <https://gitlab.inria.fr/openmp/libkomp>

Intel RUNTIME with Kaapi extensions

- ◆ same functionalities with better performances (!)

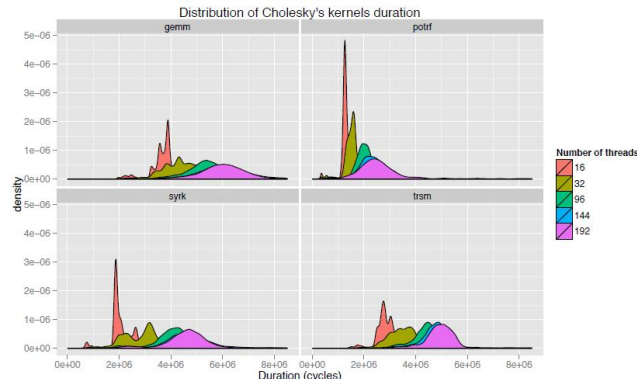
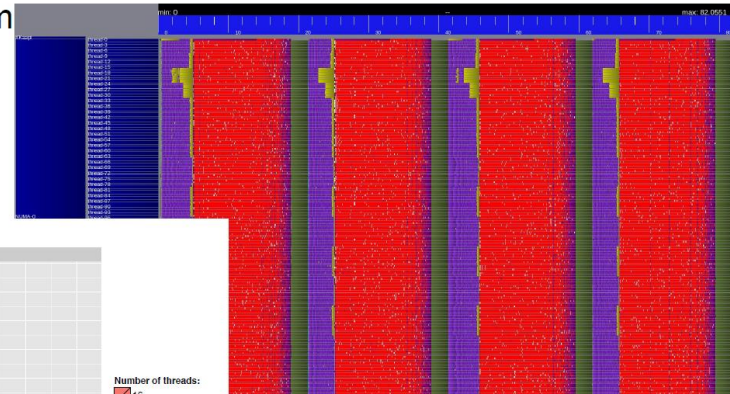
Cooperation with INRIA Team Corse

- ◆ Coupling compilation and runtime information



Version 0.1

- ◆ Include affinity scheduler from Kaapi
- ◆ Support for dependent task in the libGOMP component
- ◆ Tracing tools integrated in libOMP
 - graph
 - gantt
 - statistics about execution



PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

■ Petascale = (big) clusters of PCs + accelerators



- Performance obtained from precisely mastering data exchanges and memory accesses
- Relevant due to relative architectural uniformity
- Proximity between large scale supercomputers and local development hardware

PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

■ Petascale = (big) clusters of PCs + accelerators

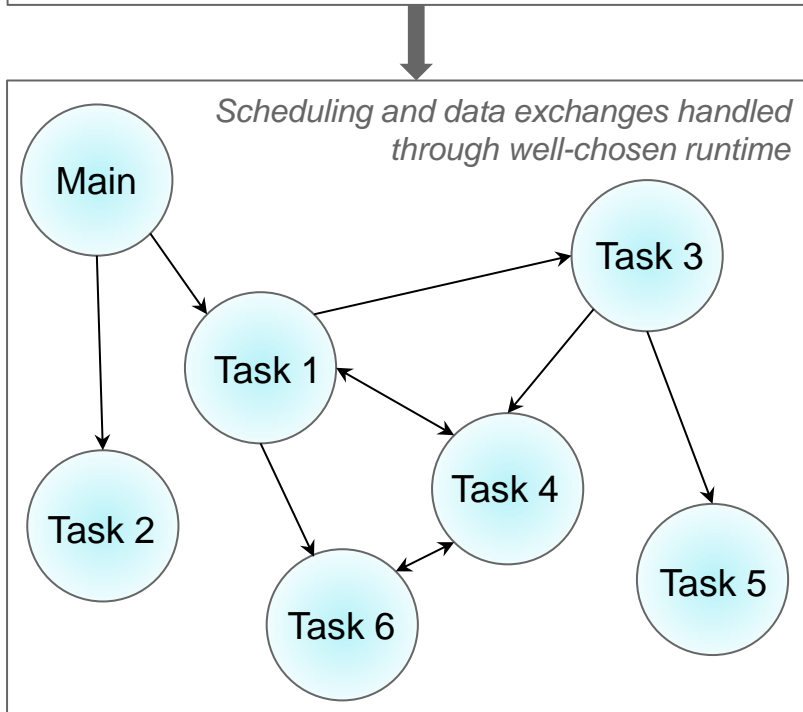
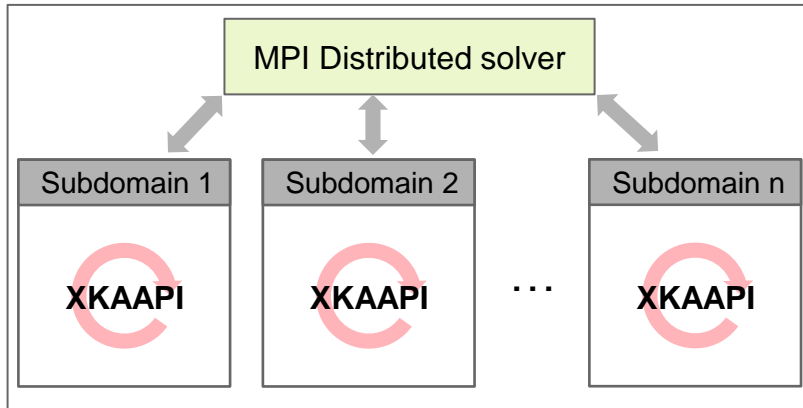


- Performance obtained from precisely mastering data exchanges and memory accesses
- Relevant due to relative architectural uniformity
- Proximity between large scale supercomputers and local development hardware

■ Exascale = *[undefined variable]*



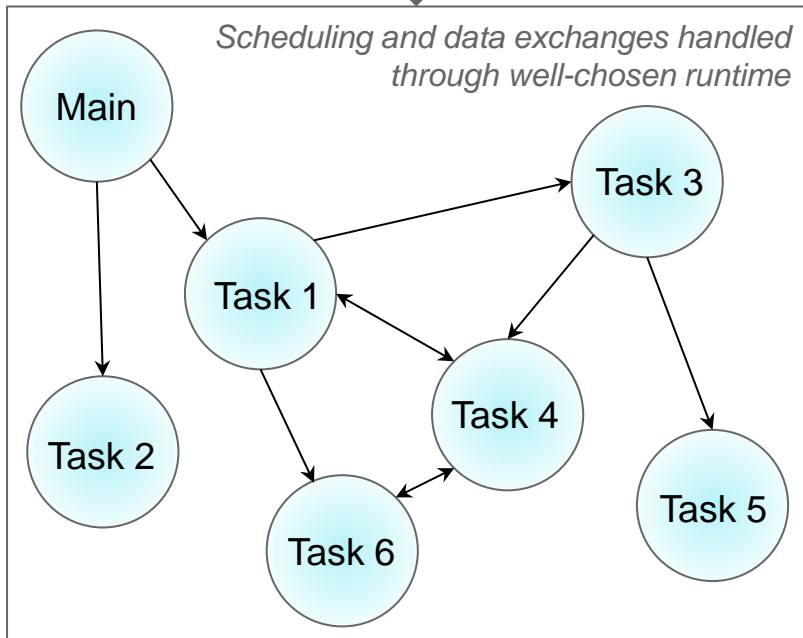
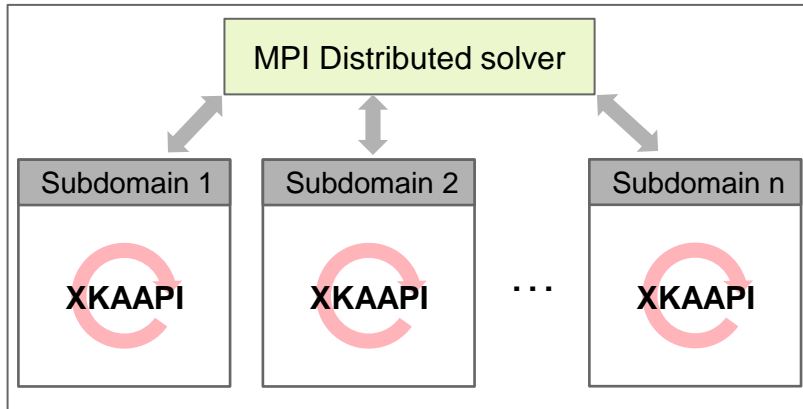
- Local performance constraints due to power management policies
- No guaranty on where the tasks are executed
- Probable strong heterogeneity
 - ❖ From one supercomputer to another
 - ❖ Inside one supercomputer



■ Exascale = [undefined variable]



- Local performance constraints due to power management policies
- No guaranty on where the tasks are executed
- Probable strong heterogeneity
 - ❖ From one supercomputer to another
 - ❖ Inside one supercomputer



■ Exascale = [undefined variable]



- Local performance constraints due to power management policies
- No guaranty on where the tasks are executed
- Probable strong heterogeneity
 - ❖ From one supercomputer to another
 - ❖ Inside one supercomputer

■ Shortcomings

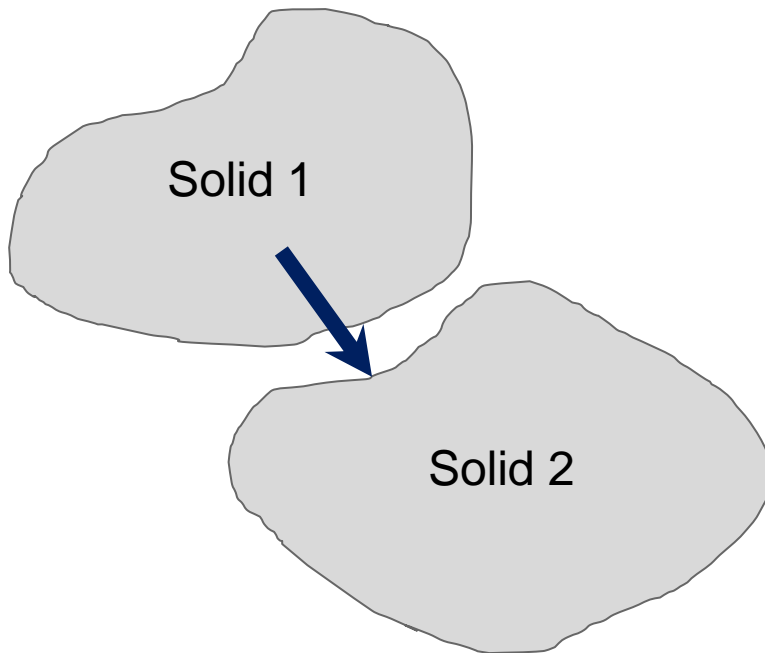
- Runtime must handle efficiently shared and distributed exchanges
- Deep evolution of the data structure to expect...

- **Need for asynchronicity**
 - Minimum constraints on data flow
 - Fill-in the computing cores at their maximum

PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

■ Need for asynchronicity

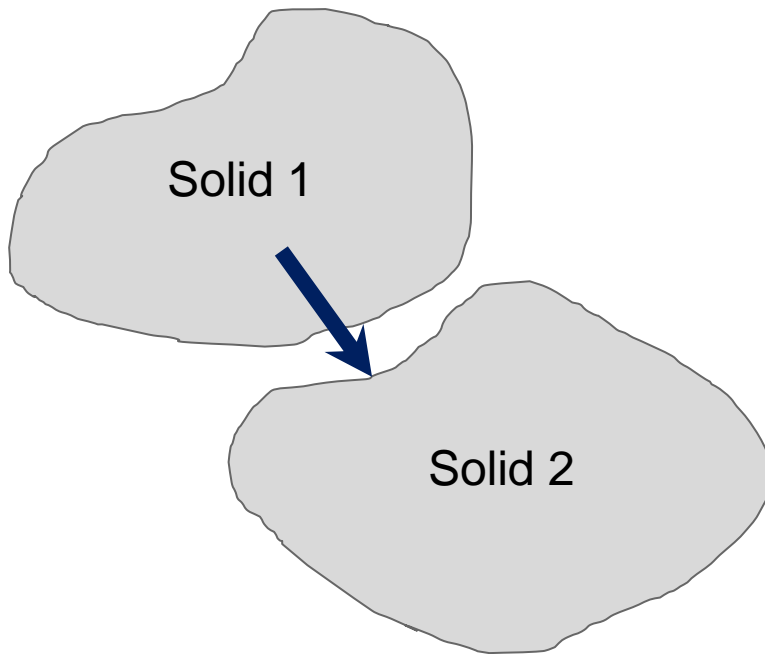
- Minimum constraints on data flow
- Fill-in the computing cores at their maximum



PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

■ Need for asynchronicity

- Minimum constraints on data flow
- Fill-in the computing cores at their maximum



Task 1

Elementary forces with solids 1 & 2

Task 2

Contact detection between solids 1 & 2

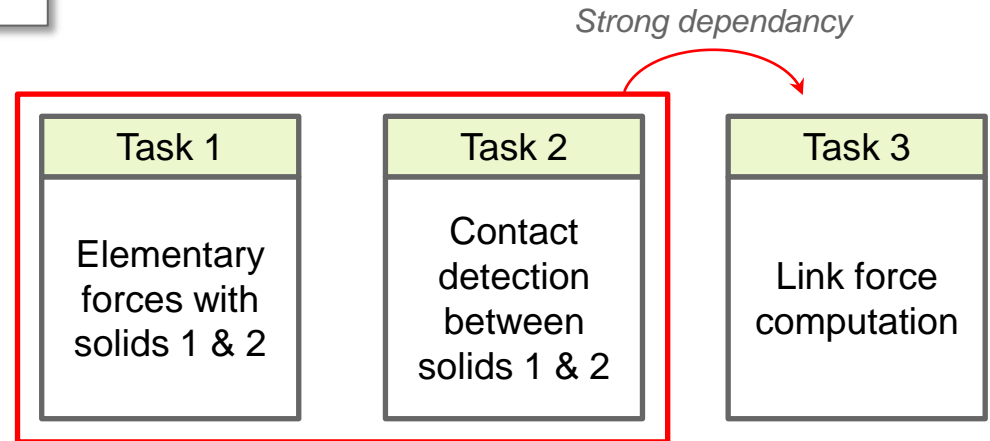
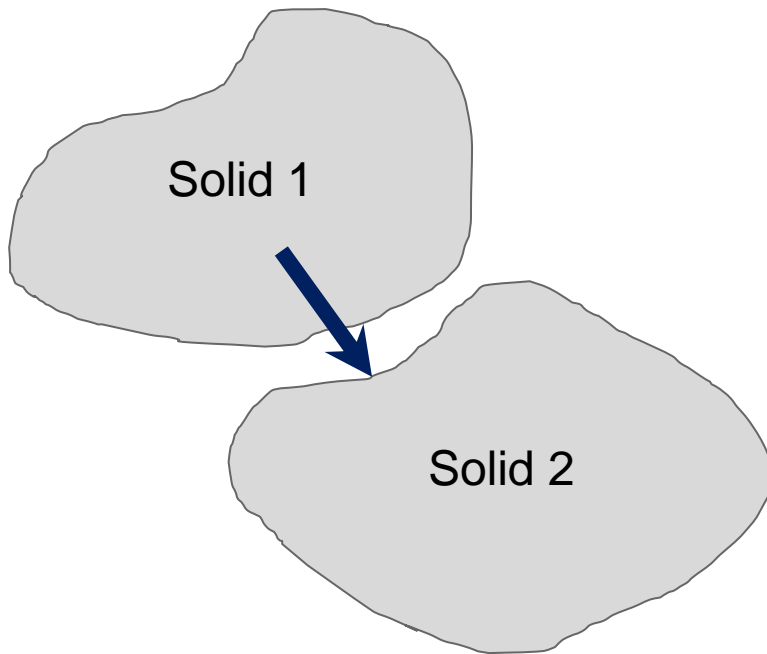
Task 3

Link force computation

PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

■ Need for asynchronicity

- Minimum constraints on data flow
- Fill-in the computing cores at their maximum

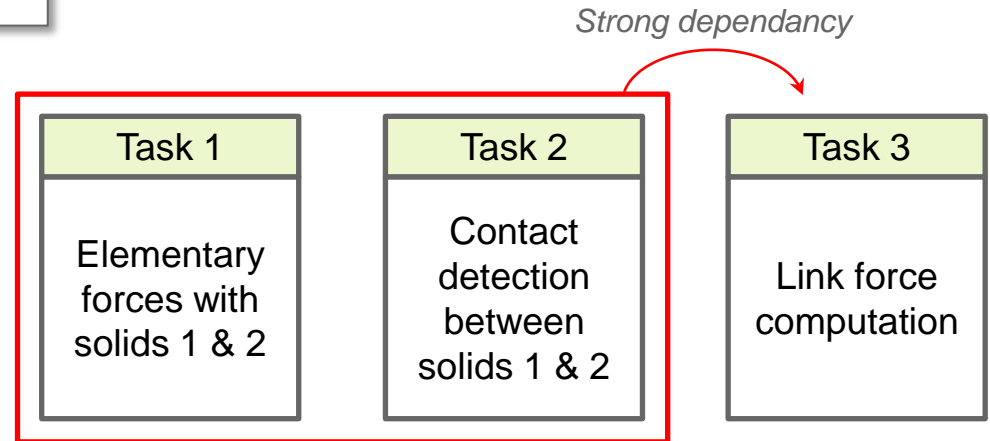
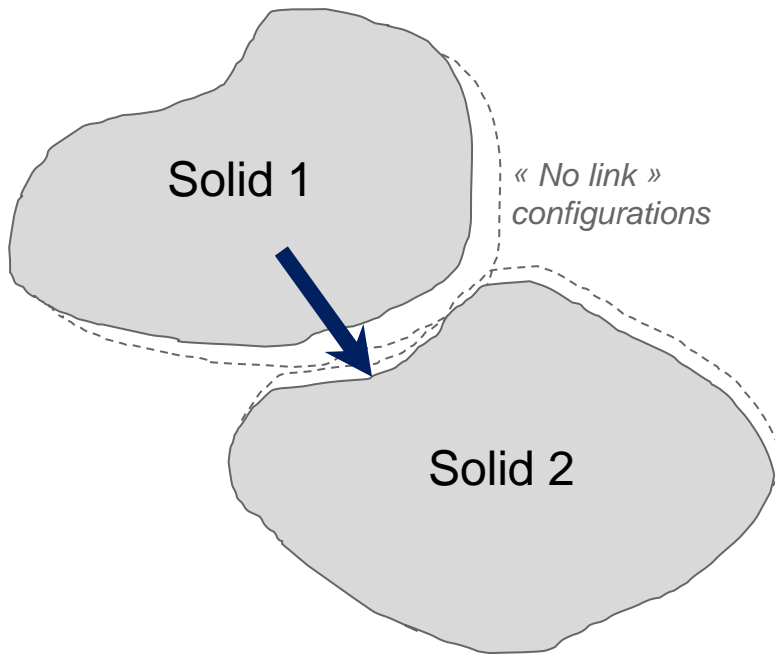


Are these tasks independants ?

PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

Need for asynchronicity

- Minimum constraints on data flow
- Fill-in the computing cores at their maximum

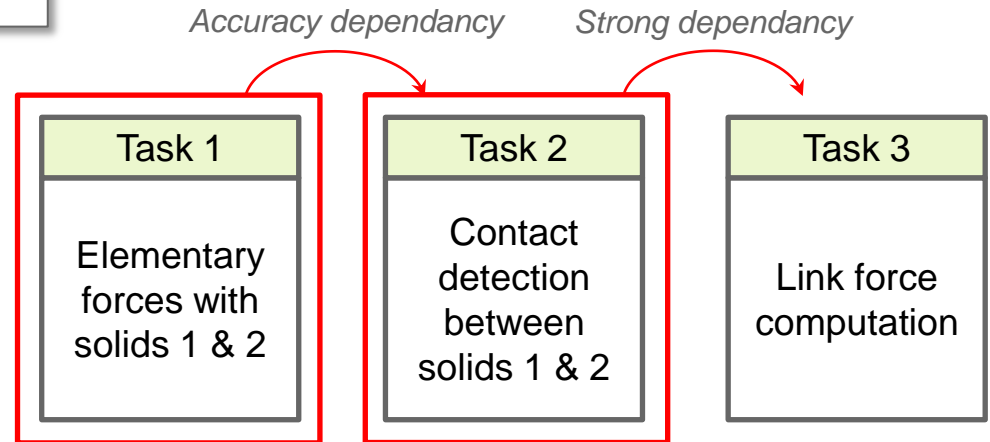
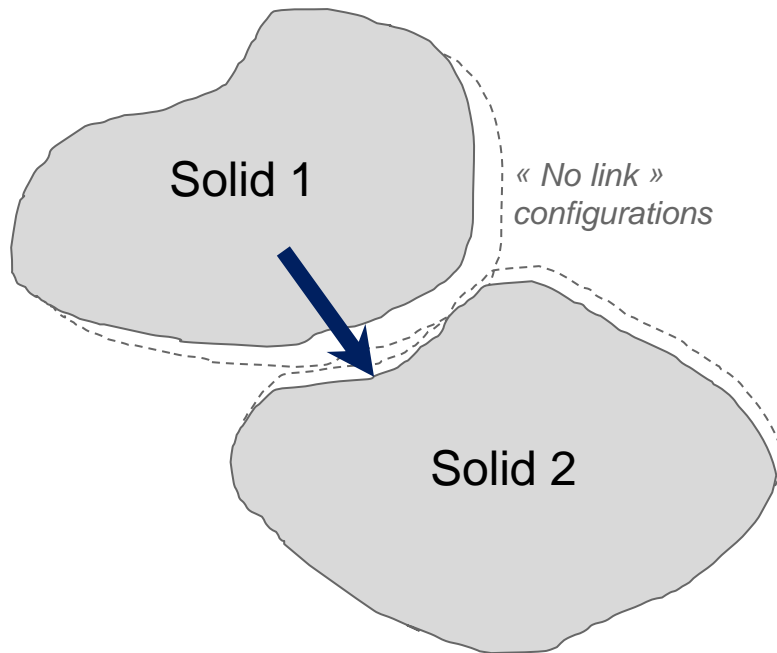


Are these tasks independants ?

PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

Need for asynchronicity

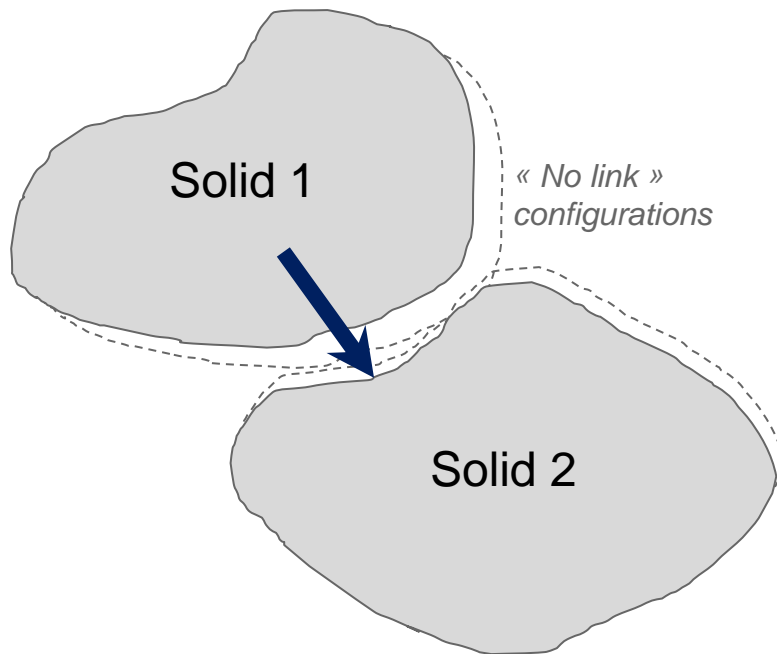
- Minimum constraints on data flow
- Fill-in the computing cores at their maximum



PERSPECTIVES GENERALES POUR EPX ET LES RUNTIMES POUR ANTICIPER L'EXASCALE

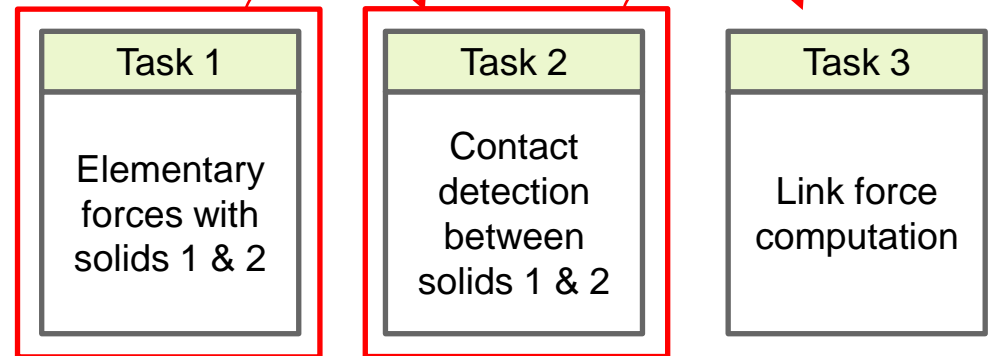
Need for asynchronicity

- Minimum constraints on data flow
- Fill-in the computing cores at their maximum



Accuracy dependancy

Strong dependancy



Beware the priorities

- Final cut in the hands of physics and applied mathematics
- Asynchronicity implies strong algorithmic innovation



<http://www-epx.cea.fr>

**MERCI POUR VOTRE ATTENTION
DES QUESTIONS ?**

Commissariat à l'énergie atomique et aux énergies alternatives
Centre de Cadarache | 13108 Saint-Paul les Durance
T. +33 (0)4 42 25 43 73 | F. **+33 (0)4 42 25 20 01**

Etablissement public à caractère industriel et commercial | RCS Paris B 775 685 019

Nuclear Energy Division
Nuclear Technology Department