

MIRAGE : Méso-Informatique Répartie pour des Applications en Géophysique et Environnement

Comprendre et prévoir les phénomènes naturels :

- ⇒ Disciplines concernées : météorologie, océanographie, hydrologie, glaciologie...**
- ⇒ Un impact sociétal important : prévention des risques naturels, gestion des ressources, aménagement du territoire, évolution du climat...**
- ⇒ Un outil privilégié : la modélisation**
- ⇒ Un travail de recherche très largement pluri-disciplinaire : mathématiciens, physiciens, numériciens, informaticiens...**

Evolution actuelle

- ➔ **Demande croissante de systèmes de prévision, dans tous les domaines de la géophysique**
- ➔ **Des systèmes de modélisation de plus en plus complexes : modèles à la physique plus complète, intégrant de plus en plus d'observations de types divers, et **couplés entre eux**.**

Atmosphere model

- 1- Rainfall
 - 2- Snowfall
 - 3- Incoming solar radia.
 - 4- Solar zenith angle
 - 5- Fraction of diffuse solar radiation
 - 6- Downward infrared radiation
 - 7- Sensitivity of atmos temp. & humidity to surf. fluxes
- ①

- 1- Sensible heat flux
 - 2- Latent heat flux
 - 3- Surf. emissivity
 - 4- Albedo, direct
 - 5- Albedo, diffuse
 - 6- Surf. radiative temp.
 - 7- Evaporation
 - 8- Wind stress
- ②

- 1- Surface pressure
 - 2-4 Air temperature, humidity and wind
 - 5- Wind module
 - 6- Height of these 4 variables
- ③

Surface layer turbulence

- 1- ρC_d
 - 2- ρC_e
 - 3- ρC_h
- ④
+③

- 1- Surf. Temp
 - 2- Surf. Roughness
 - 3- Displacement height
- ⑤

Ocean surface module

Sea ice model

wave model

- 1- Non solar heat flux
 - 2- Solar radiation
 - 3- Fresh water flux
 - 4- Salt flux
 - 5- Wind stress
 - 6- U^3
 - 7- Mass of snow and ice
- ⑥

- 1- Temp. at sea-ice base
 - 2- Sea surface temperature
 - 3- Surf. radiative temp.
 - 4- Surface ocean current
 - 5- Sea surface salinity
 - 6- Surface height
 - 7- Absorbed solar radiation (in first oceanic layer)
- ⑦

Land surface model

- 1- Continental runoff
- ⑧

Iceberg parameters

Ocean model

Couplages de modèles au sein de Mirage

⇒ Eaux de surface - Atmosphère

Projet CouMéHy (C. Messager [LTHE], H. Gallée [LGGE]) : Couplage des cycles hydrologiques atmosphériques et continentaux aux échelles régionales et climatiques - application à l'Afrique de l'Ouest. (ACI Grid)

⇒ Glace - Atmosphère

Modélisation des calottes glaciaires (C. Ritz, C. Dumas, P. Martinerie [LGGE])

⇒ Océan - Océan

Projet COMODO (E. Blayo et al. [LMC], B. Barnier, S. Cailleau [LEGI]) : Méthodologies de couplage de modèles - application au couplage d'un modèle côtier et d'un modèle de bassin.

⇒ Océan - Atmosphère

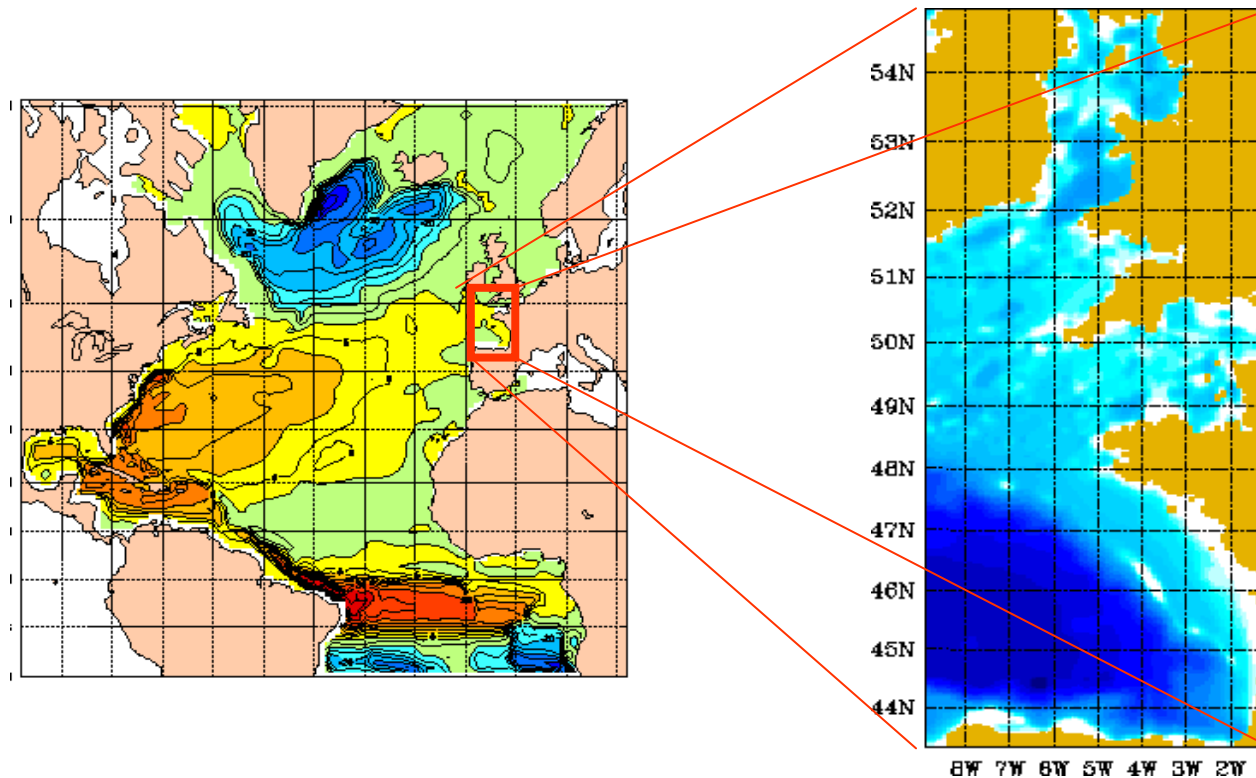
Méthodologies de couplage de modèles - application à un modèle couplé d'El Niño (E. Blayo, V. Chevallier [LMC])

Couplage de modèles (océaniques)

E. Blayo, L. Debreu, V. Fedorenko, L. Viry (LMC-IMAG Grenoble)

B. Barnier, S. Cailleau (LEGI Grenoble)

L. Halpern, C. Japhet, V. Martin (LAGA Paris 13)



Primitive equations

Momentum equations

$$\frac{\partial u}{\partial t} + \mathbf{U} \cdot \nabla u - \nu \Delta u - fv + \frac{1}{\rho_0} \frac{\partial p}{\partial x} = 0$$

$$\frac{\partial v}{\partial t} + \mathbf{U} \cdot \nabla v - \nu \Delta v + fu + \frac{1}{\rho_0} \frac{\partial p}{\partial y} = 0$$

$$\frac{\partial p}{\partial z} = -\rho g \quad (\text{hydrostatic approximation})$$

Conservation of mass

$$\text{div } \mathbf{U} = 0 \quad (\text{Boussinesq approximation})$$

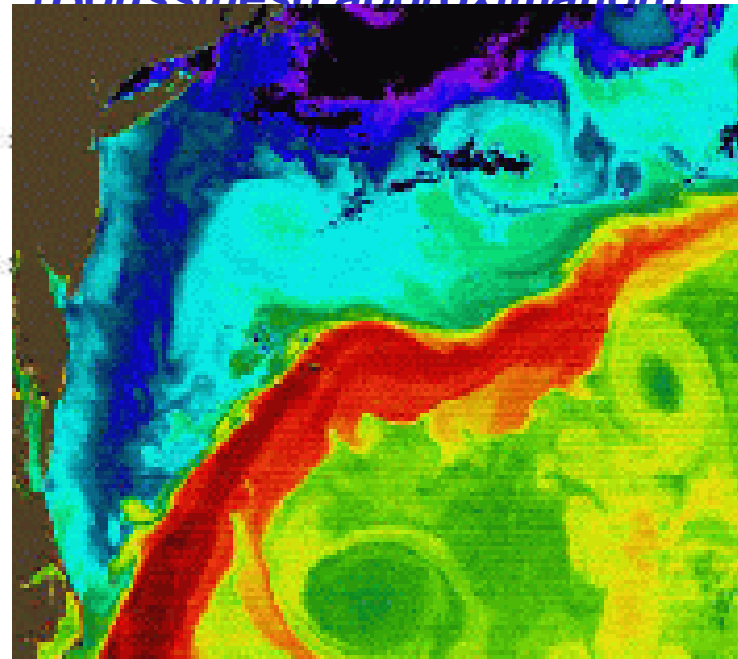
Conservation of tracers

$$\frac{\partial T}{\partial t} + \mathbf{U} \cdot \nabla T$$
$$\frac{\partial S}{\partial t} + \mathbf{U} \cdot \nabla S$$

Equation of state

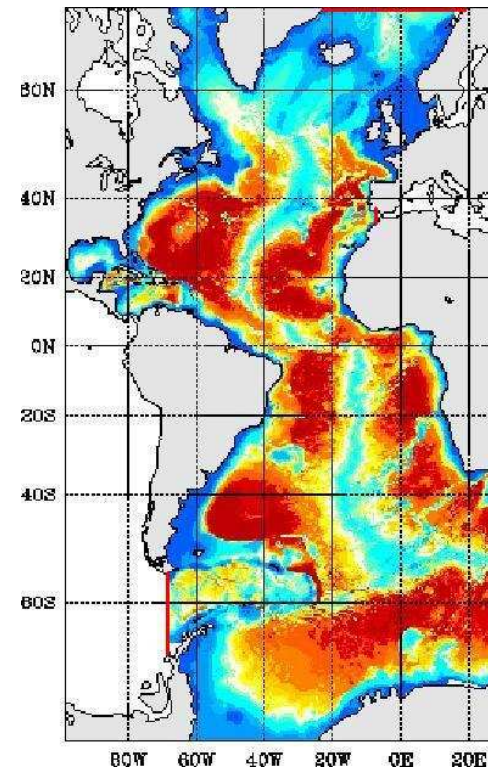
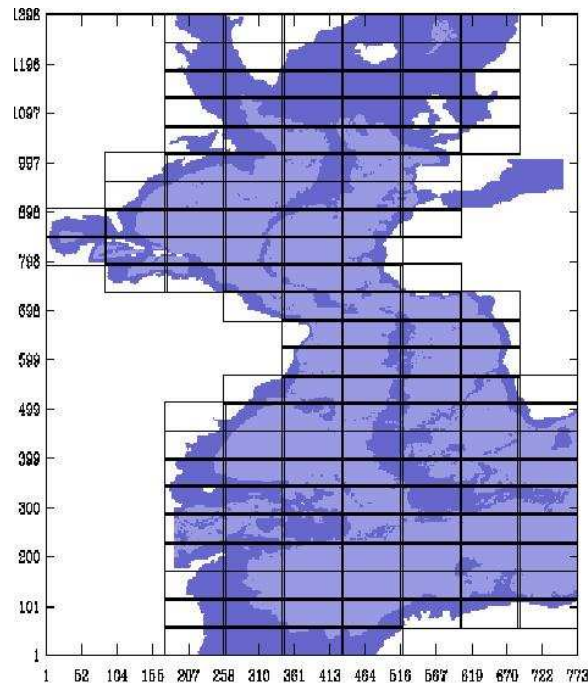
$$\rho = \rho(T, S, p)$$

+ boundary conditions...



CLIPPER Project (Brest, Grenoble, Paris) : Atlantic ocean, $1/6^\circ$

**Grid size : 773 x 1296 x 42 (x 5)
= $2.1 \cdot 10^8$ variables**



140 processors Cray T3E

A one-year integration :

34 cpu h (x 140 proc. T3E)

Storing : each model state = 1.26 Go

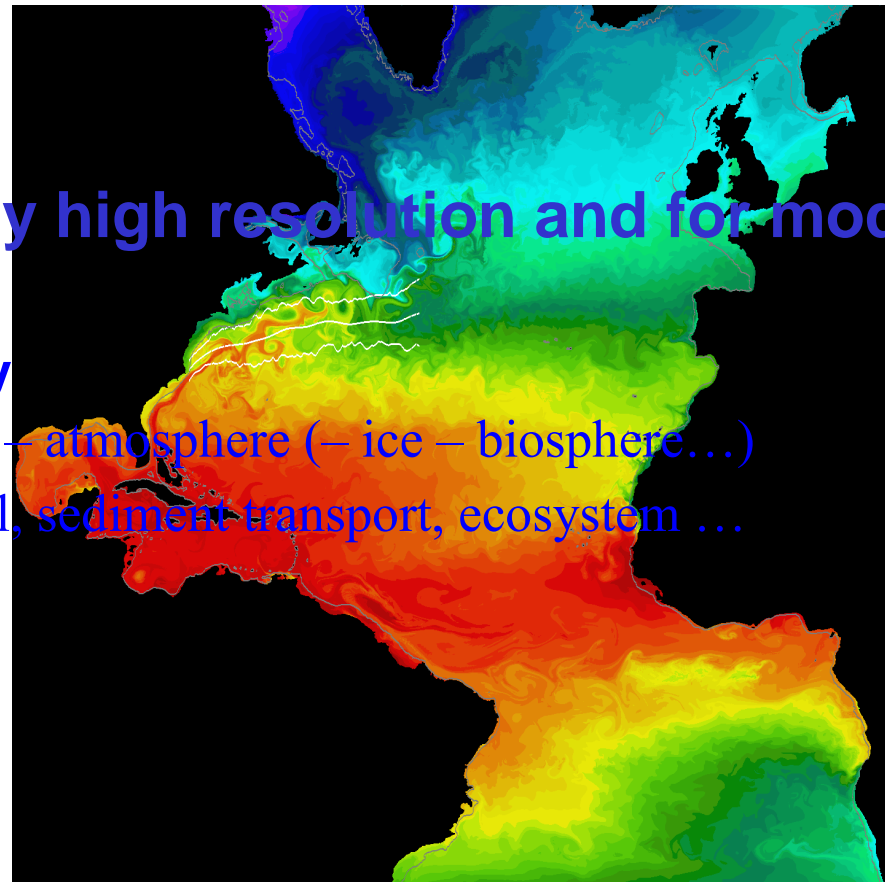
MOTIVATIONS

Ocean circulation is strongly heterogeneous

- ❑ Multiscale aspect of the ocean circulation
- ❑ Some key areas

Present needs for locally high resolution and for model coupling ...

- ❑ Coastal oceanography
- ❑ Model coupling : ocean – atmosphere (– ice – biosphere...)
large scale / coastal, sediment transport, ecosystem ...

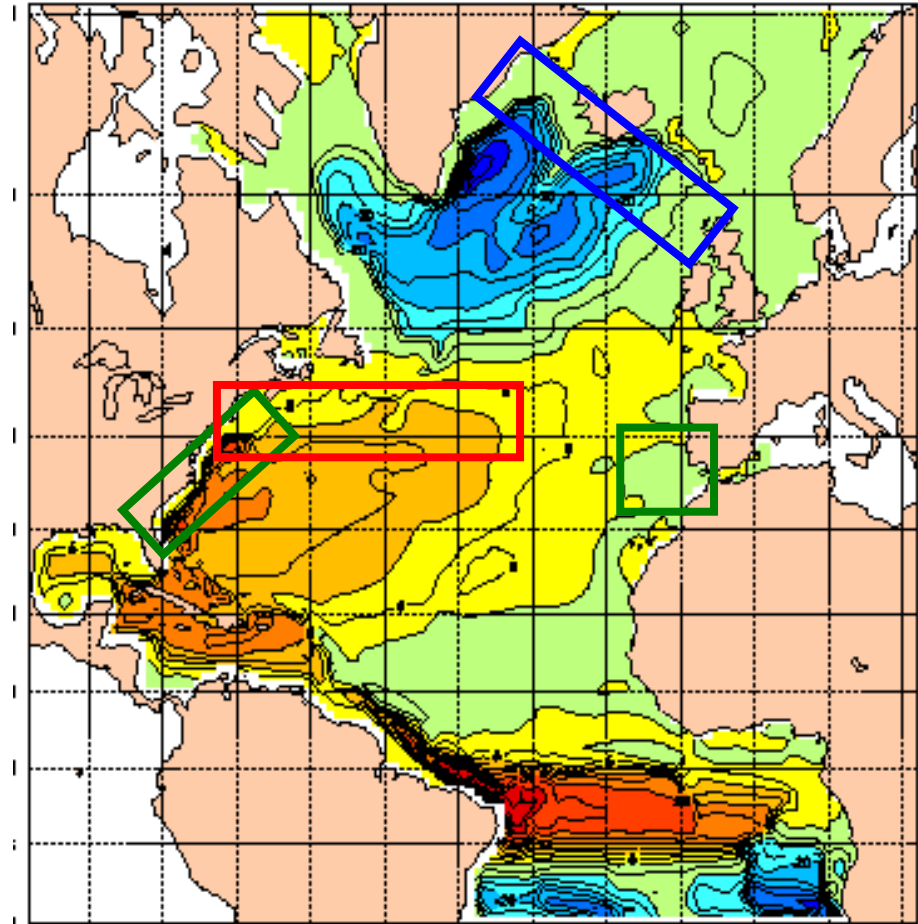


Ocean model : a **modelling system**

(i.e. different components in interaction)

Component :

- geographical domain
- physics
- numerics
- resolution



Pb : make those components work together !

- ⇒ **Physical aspects : *it must make sense***
 - ⇒ **Mathematical aspects : *consistency***
 - ⇒ **Numerical aspects : *efficiency***
 - ⇒ **Programming aspects : “ *black box* ” approach, *parallel implementation***
-
- ⇒ **Step 1 : mesh refinement (same physics and numerics)**
 - ⇒ **Step 2 : model nesting and coupling (on going project)**

Step 1 : a general tool for mesh refinement

→ to allow **locally** (very) high resolution

→ refinement can be

Constraints :

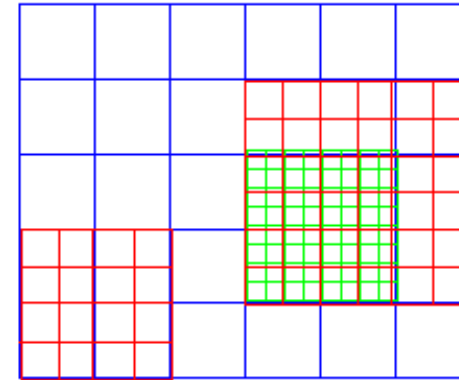
- finite differences
- model independence
- easy to use and p

→ **Hiera**



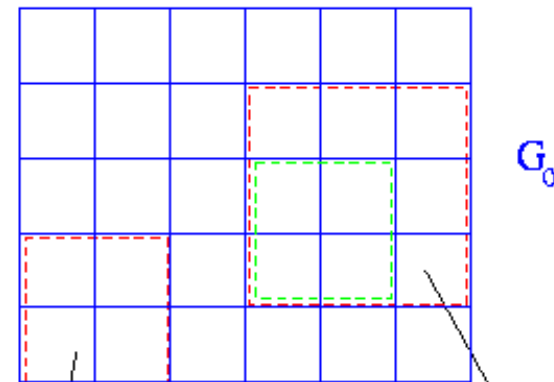
Berger and Olinger algorithm (1984)

Refined model

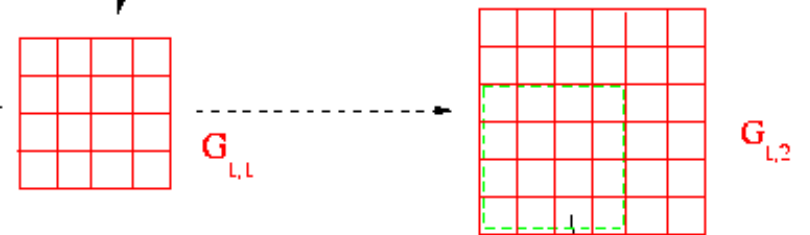


Equivalent grid hierarchy

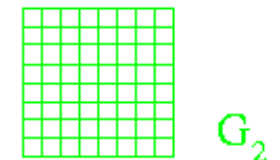
Level 0



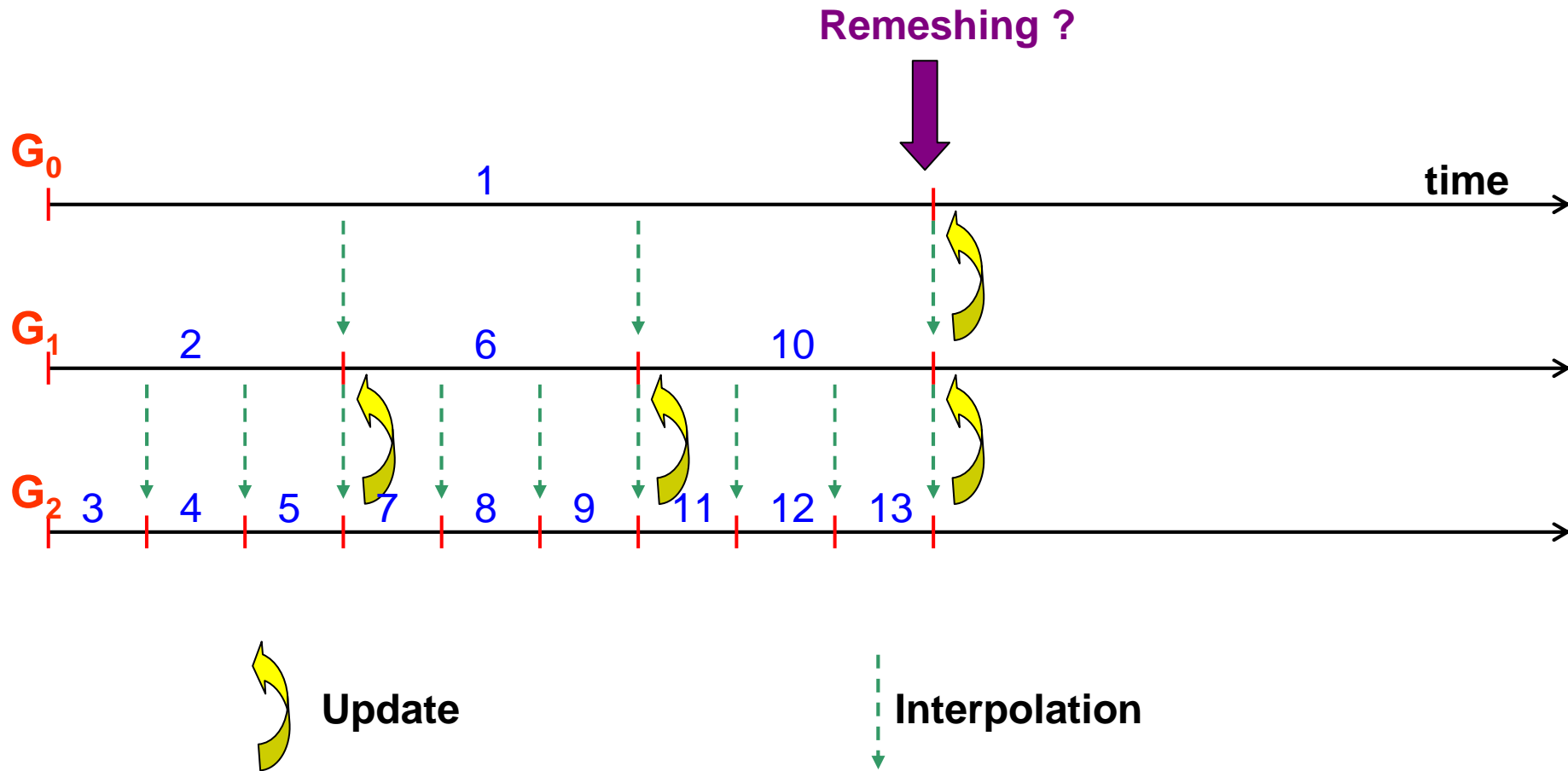
Level 1



Level 2



Time stepping procedure



Convergence and stability results : $E_h = \text{Ker } I_h^H \oplus \text{Im } (I_h^H)^T$

Specificities w.r. to multiresolution

- $u, v \gg w$: horizontal refinement (+ eventually vertical)
- Artificial horizontal diffusion to simulate the effect of unresolved scales : $A_h \Delta V$, $A_h \Delta^2 V$, $A_h(V) \Delta V$... A_h must decrease with δx
- Bathymetry varying with the resolution : shape of the domain, mass conservation...
- Eventually : high resolution forcing fields (wind, heat fluxes...)

The AGRIF package

AGRIF - Adaptive Grid Refinement In Fortran (L. Debreu, E. Blayo, C. Vouland) : a package for easily transforming an existing numerical model into a multi-resolution code (model independent, arbitrary number of grids, flux correction...)

Available for the scientific community

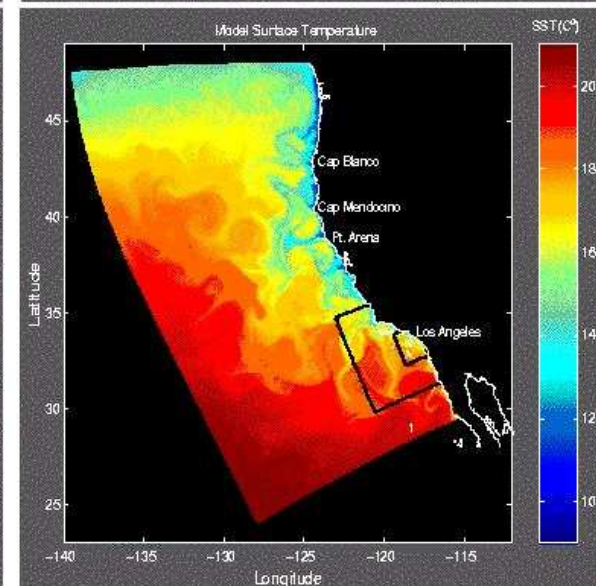
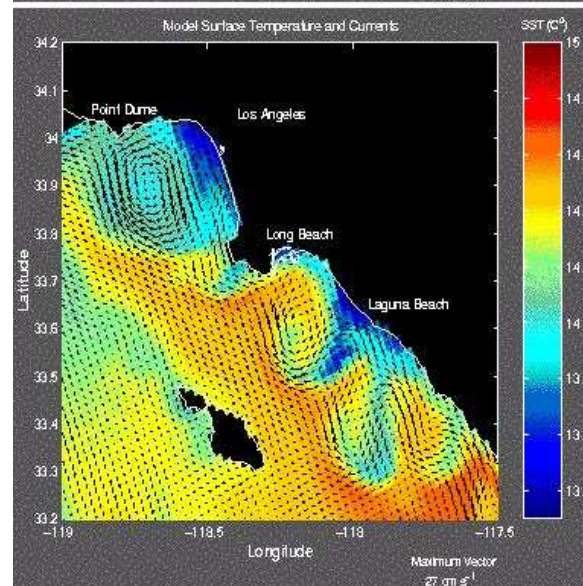
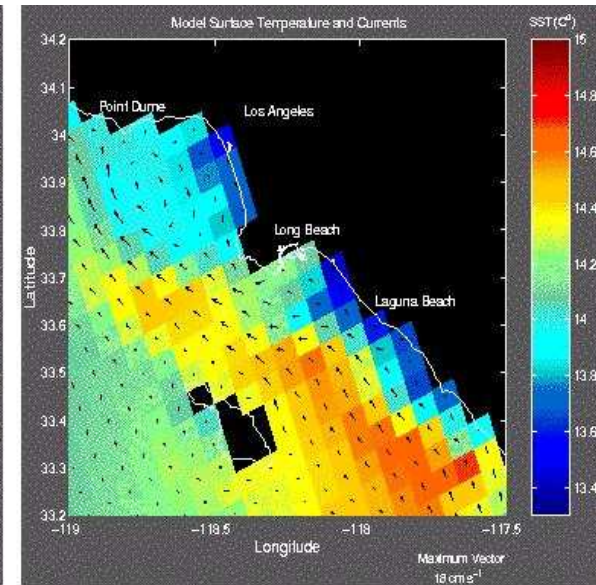
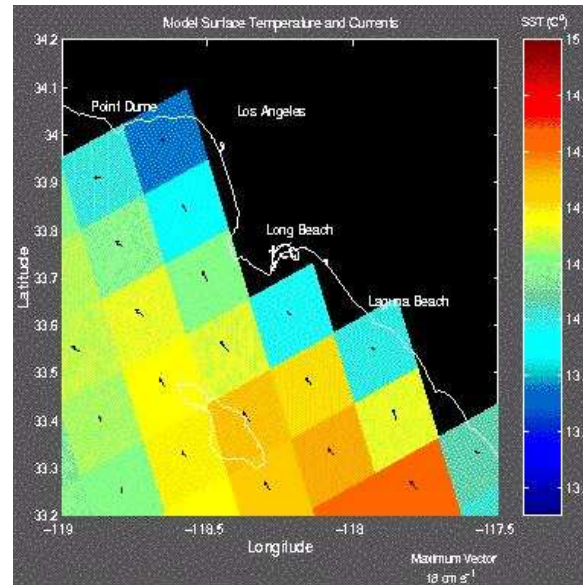
Cooperation: LEGI, SHOM, IFREMER, UCLA + several other users

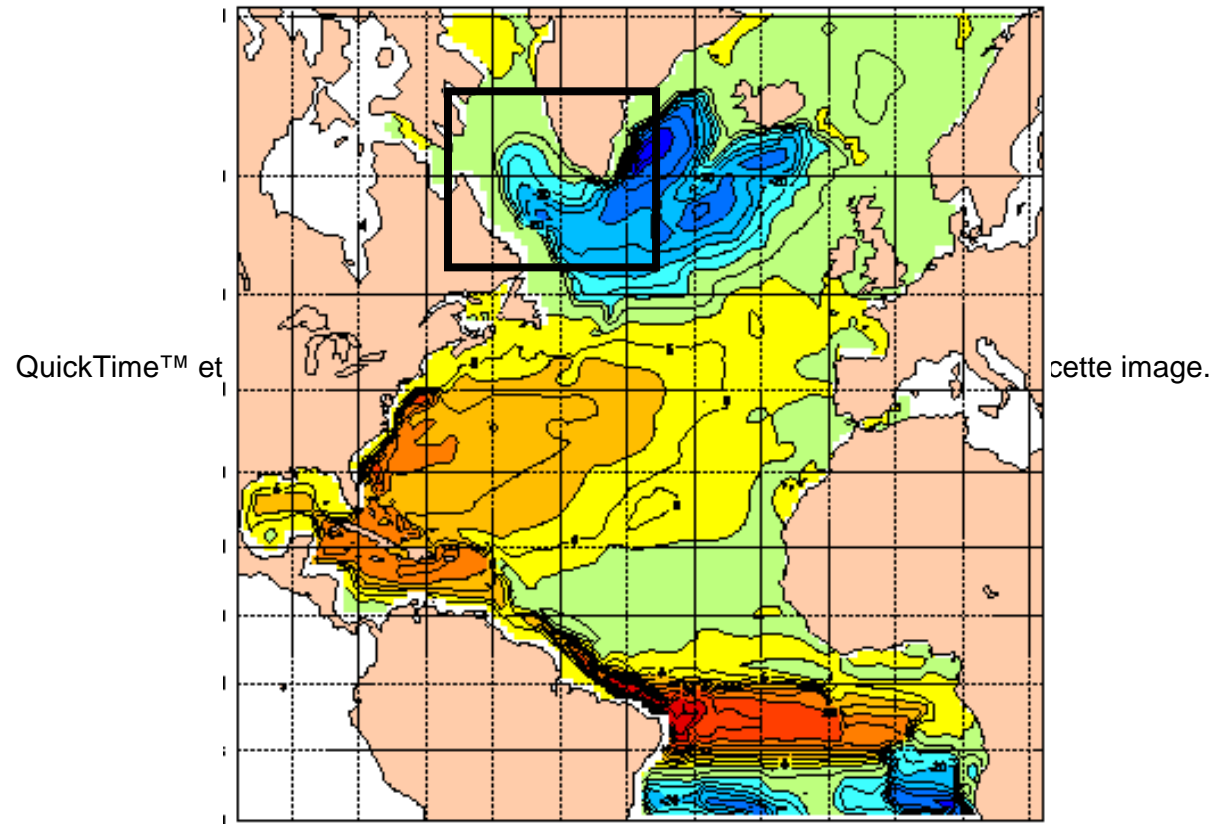
Current developments : vertical refinement, improved updates, parallelization of AGRIF (load balancing with G. Mounié and D. Trystram)

Application to a model of the Southern California bight (L. Debreu, P. Marchesiello, J. McWilliams)

ROMS model (UCLA),
+ AGRIF package

3-levels of resolution
(18 - 6 - 2 km)





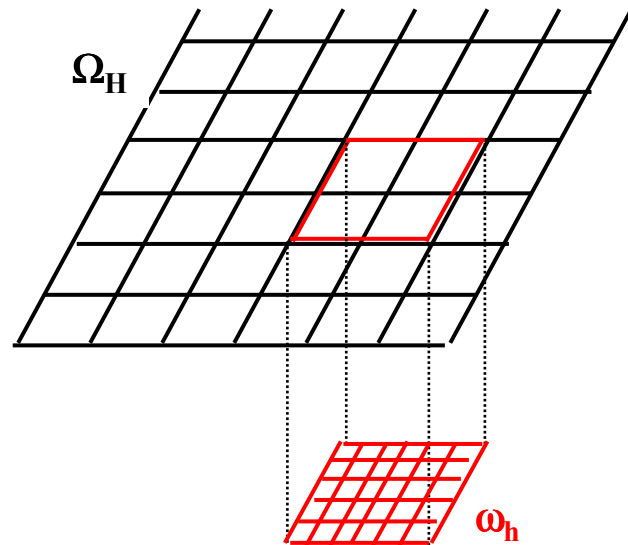
Raffinement de maillage
L. Debreu, E. Blayo, C. Vouland, J. Chanut

Step 2 : model coupling

Goals :

- ❑ development of efficient coupling algorithms, adapted to ocean models
- ❑ performing validation experiments on a realistic test-case
- ❑ meta-computing approach

Méthodes usuelles : emboîtement

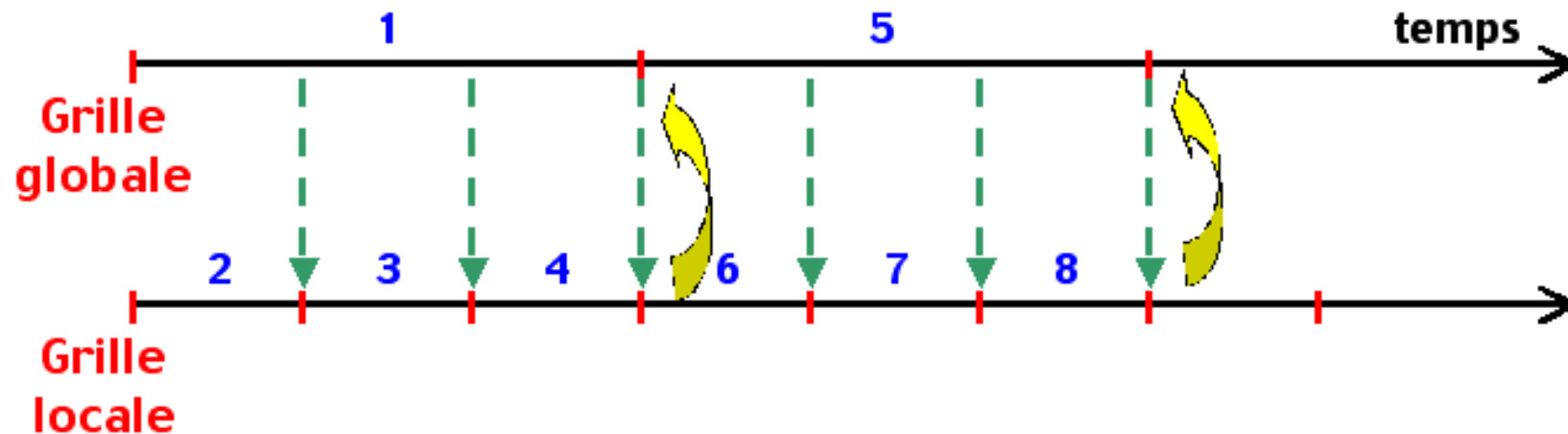


$$L_H u_H = f_H \quad \text{dans } \Omega_H$$

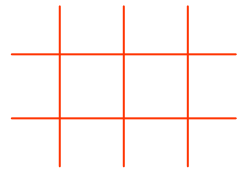
puis

$$\begin{cases} L_h u_h = f_h & \text{dans } \omega_h \\ u_h = u_H & \text{sur } \partial\omega_h \end{cases}$$

Eventuellement $u_H = \text{lissage}(u_h)$ dans $\Omega_H \cap \omega_h$



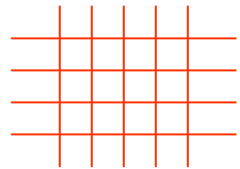
Méthodes usuelles : couplage océan/atmosphère



Ω_{air}

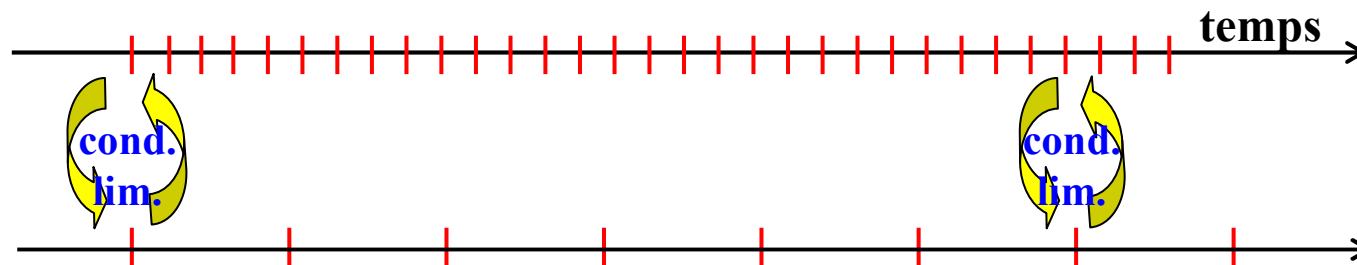
$$\begin{cases} L_{\text{air}} u_{\text{air}} = f_{\text{air}} & \text{dans } \Omega_{\text{air}} \times [t_i, t_{i+1}] \\ u_{\text{air}}(z = 0, t) = u_{\text{océan}}(z = 0, t_i) & t \in [t_i, t_{i+1}] \end{cases}$$

et



$\Omega_{\text{océan}}$

$$\begin{cases} L_{\text{océan}} u_{\text{océan}} = f_{\text{océan}} & \text{dans } \Omega_{\text{océan}} \times [t_i, t_{i+1}] \\ \frac{\partial u_{\text{océan}}}{\partial z}(z = 0, t) = \frac{\partial u_{\text{air}}}{\partial z}(z = 0, t_i) & t \in [t_i, t_{i+1}] \end{cases}$$



Méthodes usuelles : avantages

- ✎ Algorithmes synchrones, faciles à mettre en œuvre
- ✎ On impose en permanence une certaine consistance entre les solutions

→ Cette consistance est-elle suffisante ?

Exemple 1-D

$$\begin{cases} -\nu u''(x) + u(x) = \sin nx & x \in]0, 1[\\ u(0) = u(1) = 0 \end{cases}$$

Couplage :



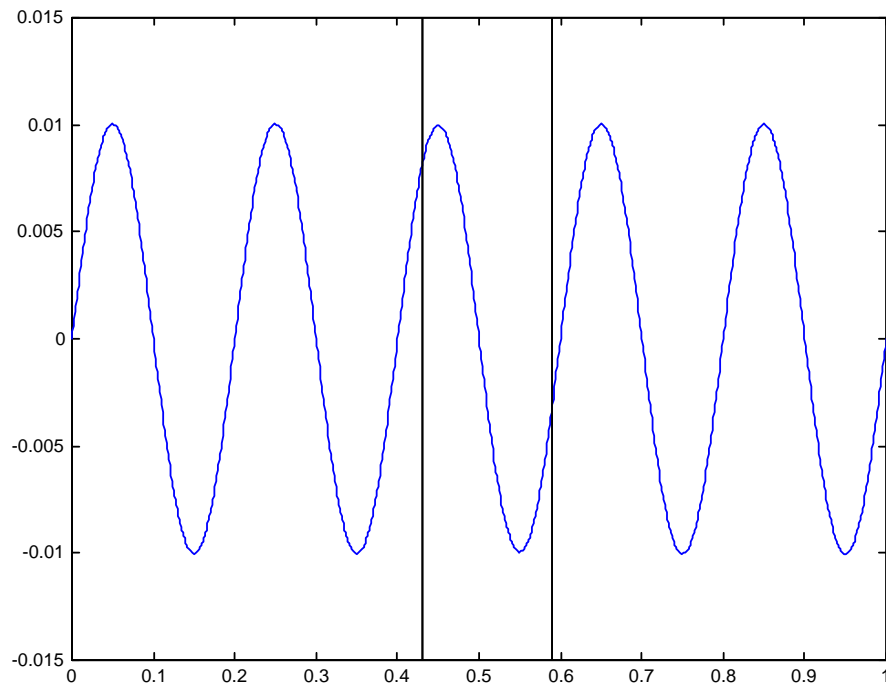
Emboîtement :



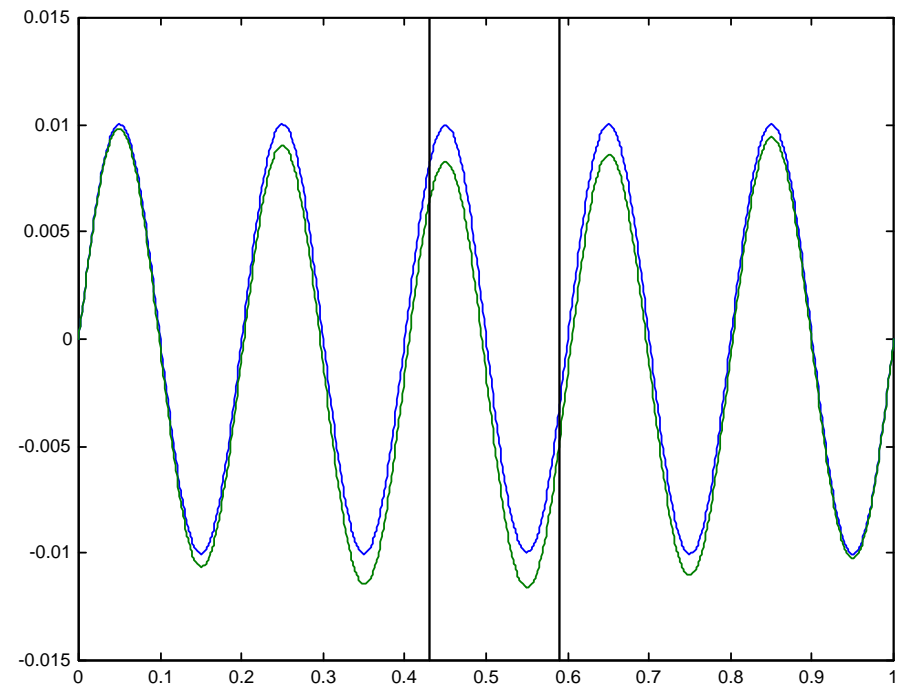
Exemple 1-D (2)

$$\begin{cases} -\nu u''(x) + u(x) = \sin nx & x \in]0, a[\cup]b, 1[\\ -\nu u''(x) + u(x) = \sin nx + 0.05 \sin 3nx & x \in]a, b[\\ u(0) = u(1) = 0 \end{cases}$$

Emboîtement



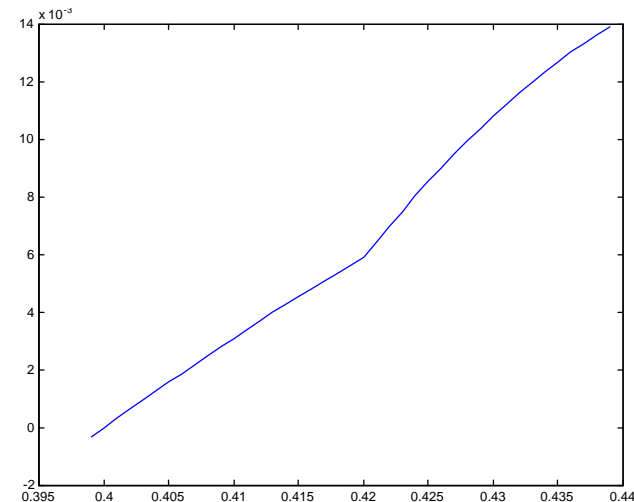
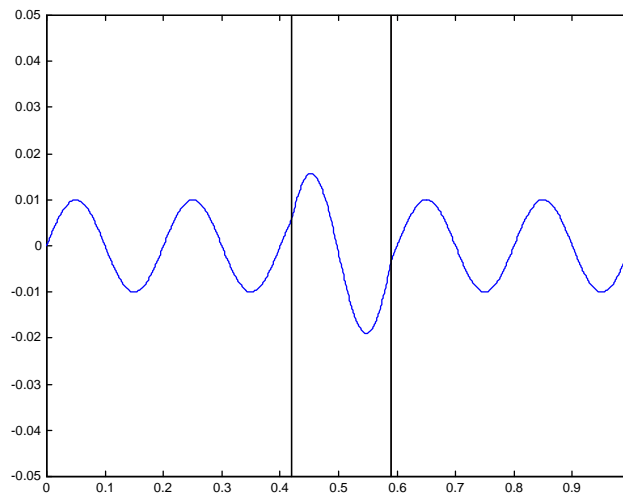
Couplage



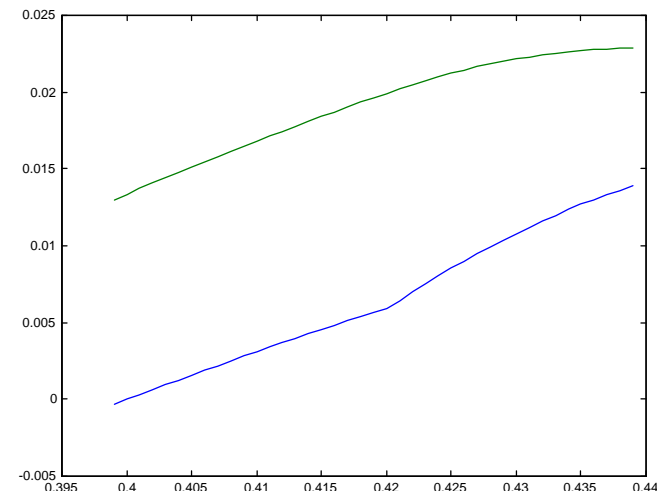
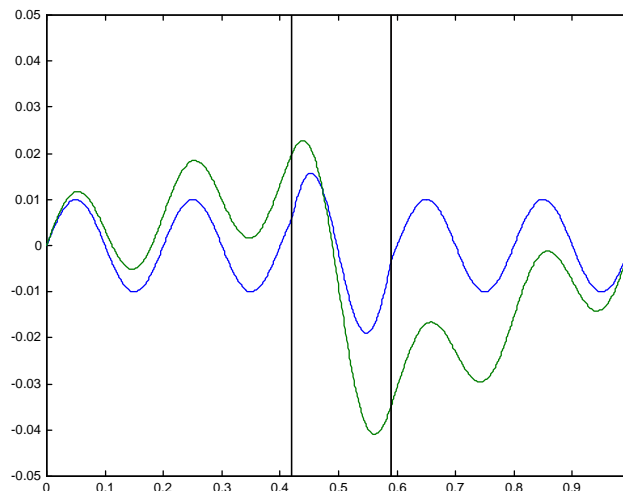
Exemple 1-D (3)

$$\begin{cases} -\nu u''(x) + u(x) = \sin nx & x \in]0, a[\cup]b, 1[\\ -\frac{\nu}{2} u''(x) + u(x) = \sin nx & x \in]a, b[\\ u(0) = u(1) = 0 \end{cases}$$

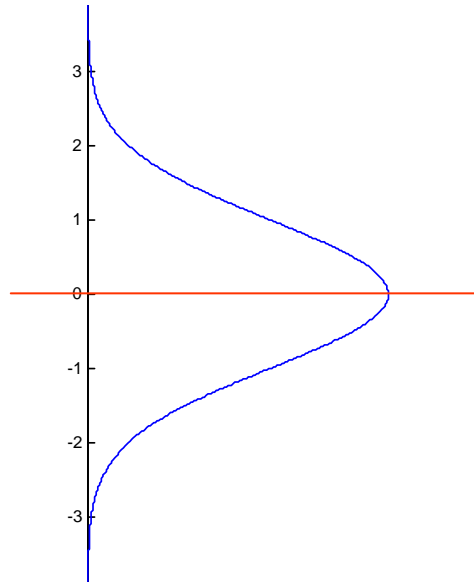
Emboîtement



Couplage

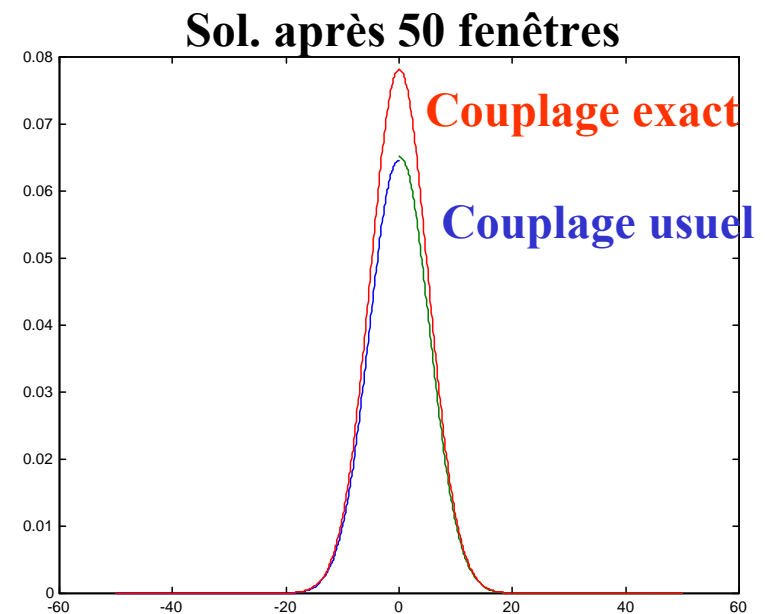
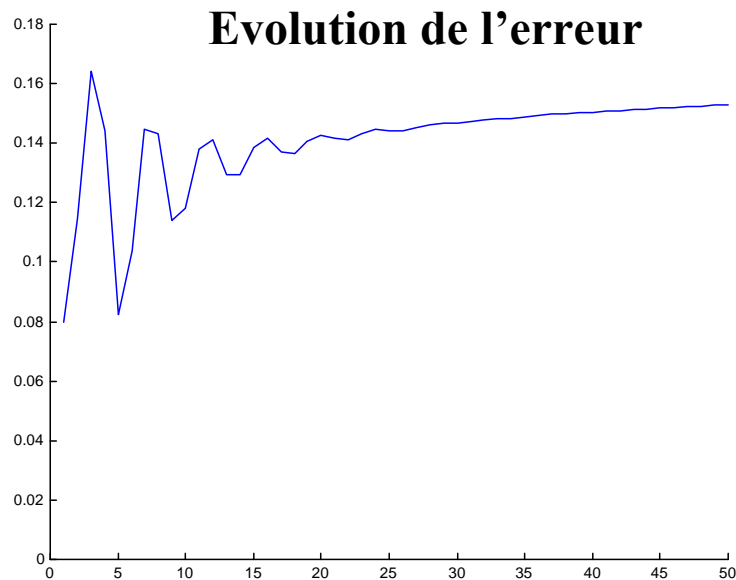


Exemple 1-D (4)



$$\begin{cases} \frac{\partial u_1}{\partial t}(z, t) - \nu \frac{\partial^2 u_1}{\partial z^2}(z, t) = 0 & t \in]t_i, t_{i+1}] \\ u_1(0, t) = u_2(0, t_i) & t \in]t_i, t_{i+1}] \end{cases}$$

$$\begin{cases} \frac{\partial u_2}{\partial t}(z, t) - \nu \frac{\partial^2 u_2}{\partial z^2}(z, t) = 0 & t \in]t_i, t_{i+1}] \\ \frac{\partial u_2}{\partial z}(0, t) = \frac{\partial u_1}{\partial z}(0, t_i) & t \in]t_i, t_{i+1}] \end{cases}$$



Couplage de modèles

D'un point de vue mathématique :

↻ Problème de modélisation

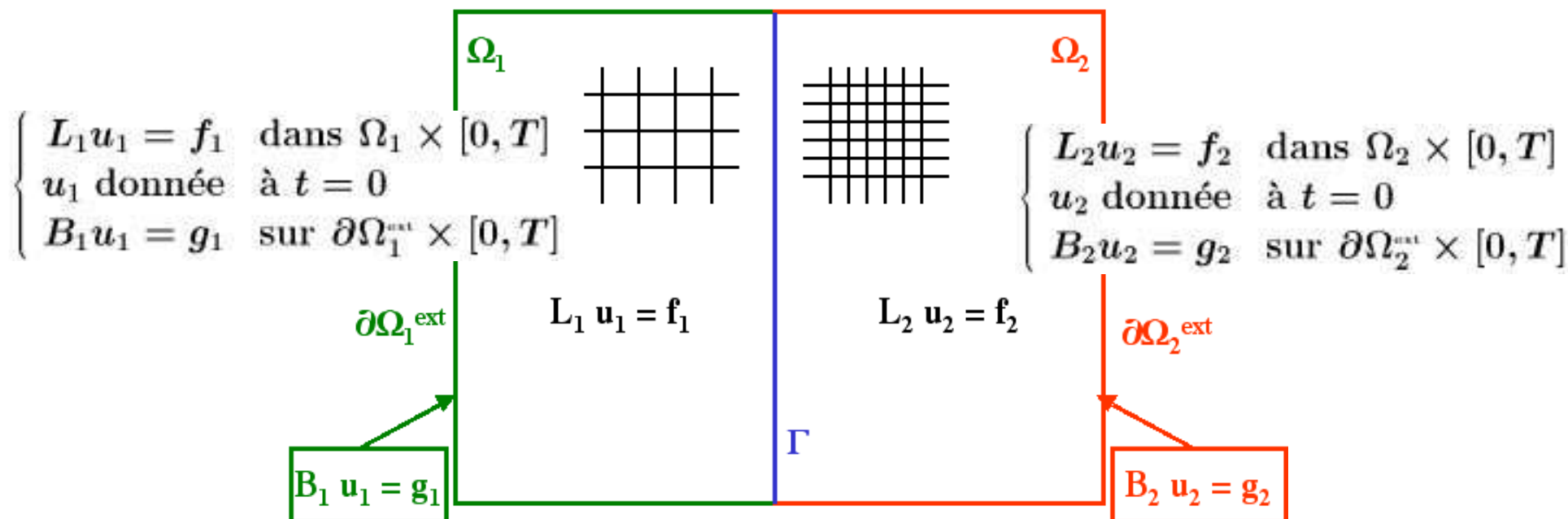
transition “lisse” des équations entre les deux modèles

↻ Comment résoudre correctement ces équations couplées ?

analyse mathématique, méthodes numériques

D'un point de vue physique : cela est-il utile dans le contexte applicatif, i.e. y a-t-il amélioration de la solution physique ?

Les méthodes de Schwarz



- u_2^0 donnée sur Γ

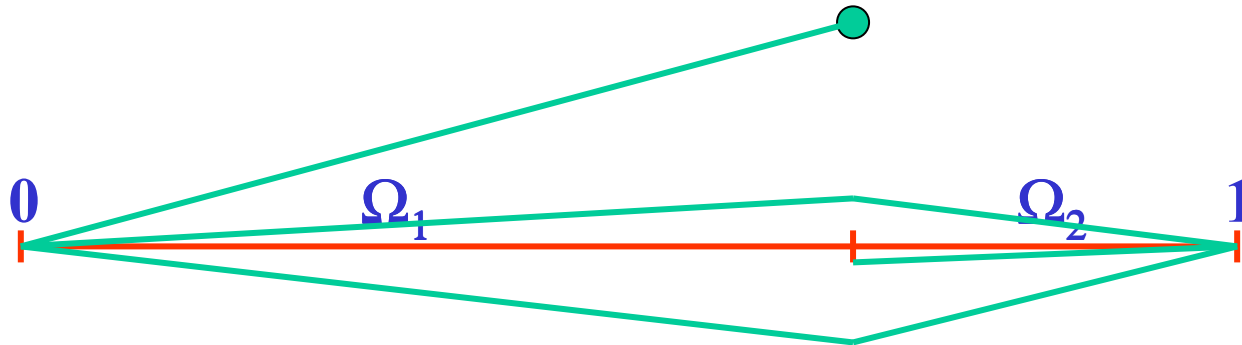
● Iterate

$$\begin{cases} L_1 u_1^{n+1} = f_1 & \text{dans } \Omega_1 \times [0, T] \\ u_1^{n+1} \text{ donnée} & \text{à } t = 0 \\ B_1 u_1^{n+1} = g_1 & \text{sur } \partial\Omega_1^{\text{ext}} \times [0, T] \\ C_1 u_1^{n+1} = C_1 u_2^n & \text{sur } \Gamma \end{cases}$$

$$\begin{cases} L_2 u_2^{n+1} = f_2 & \text{dans } \Omega_2 \times [0, T] \\ u_2^{n+1} \text{ donnée} & \text{à } t = 0 \\ B_2 u_2^{n+1} = g_2 & \text{sur } \partial\Omega_2^{\text{ext}} \times [0, T] \\ C_2 u_2^{n+1} = C_2 u_1^{n+1} & \text{sur } \Gamma \end{cases}$$

Méthodes de Schwarz : exemple 1-D

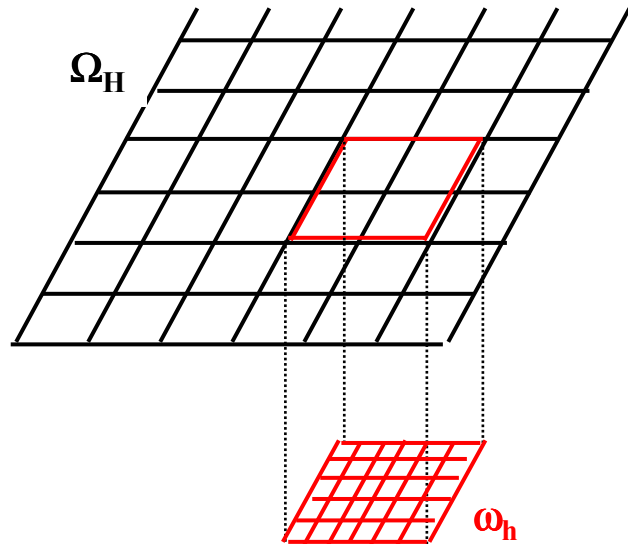
$$\begin{cases} u''(x) = 0 & x \in]0, 1[\\ u(0) = u(1) = 0 \end{cases} \quad \text{solution exacte : } u \equiv 0$$



$$\begin{cases} (u_1^{n+1})''(x) = 0 & x \in]0, a[\\ u_1^{n+1}(0) = 0 \\ u_1^{n+1}(a) = u_2^n(a) \end{cases}$$

$$\begin{cases} (u_2^{n+1})''(x) = 0 & x \in]a, 1[\\ u_2^{n+1}(1) = 0 \\ \frac{\partial u_2^{n+1}}{\partial x}(a) = \frac{\partial u_1^{n+1}}{\partial x}(a) \end{cases}$$

Méthodes de Schwarz et couplages usuels



$$L_H u_H = f_H \quad \text{dans } \Omega_H$$

puis

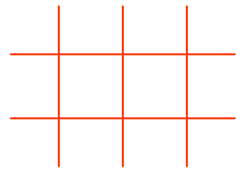
$$\begin{cases} L_h u_h = f_h & \text{dans } \omega_h \\ u_h = u_H & \text{sur } \partial\omega_h \end{cases}$$

Eventuellement $u_H = \text{lissage}(u_h)$ dans $\Omega_H \cap \omega_h$



**Une itération d'un algorithme de Schwarz
(multiplicatif) avec recouvrement total**

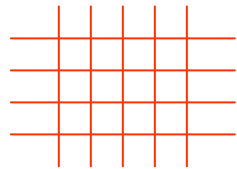
Méthodes de Schwarz et couplages usuels (2)



Ω_{air}

$$\begin{cases} L_{\text{air}} u_{\text{air}} = f_{\text{air}} & \text{dans } \Omega_{\text{air}} \times [t_i, t_{i+1}] \\ u_{\text{air}}(z = 0, t) = u_{\text{ocean}}(z = 0, t_i) & t \in [t_i, t_{i+1}] \end{cases}$$

et



Ω_{ocean}

$$\begin{cases} L_{\text{ocean}} u_{\text{ocean}} = f_{\text{ocean}} & \text{dans } \Omega_{\text{ocean}} \times [t_i, t_{i+1}] \\ \frac{\partial u_{\text{ocean}}}{\partial z}(z = 0, t) = \frac{\partial u_{\text{air}}}{\partial z}(z = 0, t_i) & t \in [t_i, t_{i+1}] \end{cases}$$



Une itération d'un algorithme de Schwarz (additif) sans recouvrement

Méthodes de Schwarz : coût

Coût de l'algo de Schwarz = coût des modèles **x nb d'itérations**

→ Trouver des conditions d'interface qui assurent une convergence rapide

$$\left\{ \begin{array}{l} L_1 u_1^{n+1} = f_1 \quad \text{in } \Omega_1 \times [0, T] \\ u_1^{n+1} \text{ given} \quad \text{at } t = 0 \\ B_1 u_1^{n+1} = g_1 \quad \text{on } \partial\Omega_1^{\text{ext}} \times [0, T] \\ C_1 u_1^{n+1} = C_1 u_2^n \quad \text{on } \Gamma \end{array} \right. \quad \left\{ \begin{array}{l} L_2 u_2^{n+1} = f_2 \quad \text{in } \Omega_2 \times [0, T] \\ u_2^{n+1} \text{ given} \quad \text{at } t = 0 \\ B_2 u_2^{n+1} = g_2 \quad \text{on } \partial\Omega_2^{\text{ext}} \times [0, T] \\ C_2 u_2^{n+1} = C_2 u_1^n \quad \text{on } \Gamma \end{array} \right.$$

Erreurs : $e_i^n = u_i - u_i^n \quad i = 1, 2$

$$\left\{ \begin{array}{l} L_1 e_1^{n+1} = 0 \quad \text{in } \Omega_1 \times [0, T] \\ e_1^{n+1} = 0 \quad \text{at } t = 0 \\ B_1 e_1^{n+1} = 0 \quad \text{on } \partial\Omega_1^{\text{ext}} \times [0, T] \\ C_1 e_1^{n+1} = C_1 e_2^n \quad \text{on } \Gamma \end{array} \right. \quad \left\{ \begin{array}{l} L_2 e_2^{n+1} = 0 \quad \text{in } \Omega_2 \times [0, T] \\ e_2^{n+1} = 0 \quad \text{at } t = 0 \\ B_2 e_2^{n+1} = 0 \quad \text{on } \partial\Omega_2^{\text{ext}} \times [0, T] \\ C_2 e_2^{n+1} = C_2 e_1^n \quad \text{on } \Gamma \end{array} \right.$$

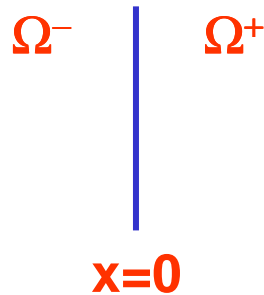
→ Si l'on trouve C_1 et C_2 tels que $C_1 e_2^n = 0$ et $C_2 e_1^n = 0$ alors l'algorithme converge exactement en deux itérations.

Méthodes de Schwarz : conditions d'interface

Conditions aux limites artificielles exactes et approchées
(Engquist & Majda, 1977; Halpern, 1986; Nataf et al., 1995; Lie, 2001)

Exemple : advection-diffusion (V. Martin)

$$Lu = \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y} - \nu \Delta u + cu = f \quad \text{in } \mathbb{R}^2 \times [0, T]$$



$$\hat{u}(x, k, \omega) = \frac{1}{2\pi} \int_t \int_y u(x, y, t) e^{i(ky + \omega t)} dy dt$$

$$\hat{L} = -\nu \frac{\partial^2}{\partial x^2} + a \frac{\partial}{\partial x} + (i(\omega + bk) + c + \nu k^2) Id = -\nu \left(\frac{\partial}{\partial x} - \lambda^+ Id \right) \left(\frac{\partial}{\partial x} - \lambda^- Id \right)$$

$$\text{with } \lambda^\pm = \frac{1}{2\nu} \left(a \pm \sqrt{a^2 + 4\nu c + 4\nu i(\omega + bk) + 4\nu^2 k^2} \right)$$

Méthodes de Schwarz : conditions d'interface (2)

$$\begin{cases} Le_1^n = 0 \implies \hat{L}\hat{e}_1^n = 0 \implies \hat{e}_1^n = \alpha^n e^{\lambda^+ x} \\ Le_2^n = 0 \implies \hat{L}\hat{e}_2^n = 0 \implies \hat{e}_2^n = \beta^n e^{\lambda^- x} \end{cases}$$

$$\implies \begin{cases} \frac{\partial \hat{e}_1^n}{\partial x} - \lambda^+ \hat{e}_1^n = 0 \implies \frac{\partial e_1^n}{\partial x} - \Lambda^+ e_1^n = 0 \\ \frac{\partial \hat{e}_2^n}{\partial x} - \lambda^- \hat{e}_2^n = 0 \implies \frac{\partial e_2^n}{\partial x} - \Lambda^- e_2^n = 0 \end{cases} \quad \text{with } \Lambda^\pm(u) = TF^{-1}(\lambda^\pm \hat{u})$$

Idéalement : $C_1 = \frac{\partial}{\partial x} - \Lambda^- Id$ and $C_2 = \frac{\partial}{\partial x} - \Lambda^+ Id$

Mais Λ^+ et Λ^- sont non-locaux \longrightarrow opérateurs approchés

Méthodes de Schwarz : conditions d'interface (3)

Approximations de Taylor :

$$\text{Ordre 0 : } \lambda_0^\pm = \frac{a \pm \sqrt{a^2 + 4\nu c}}{2\nu} \implies \Lambda_0^\pm = \frac{a \pm \sqrt{a^2 + 4\nu c}}{2\nu} Id$$

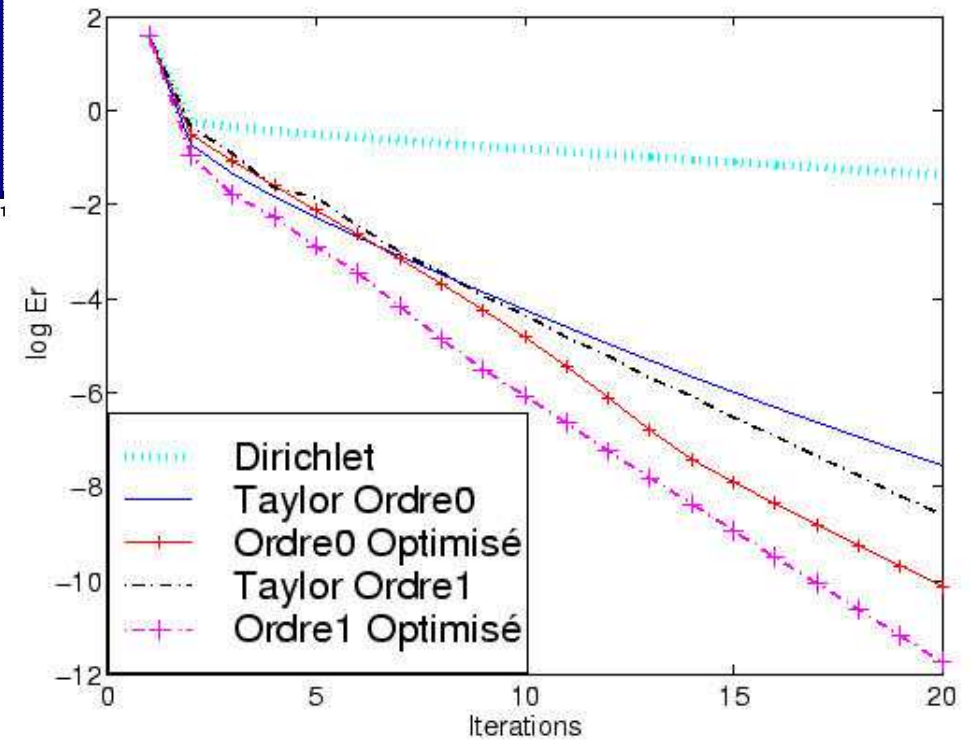
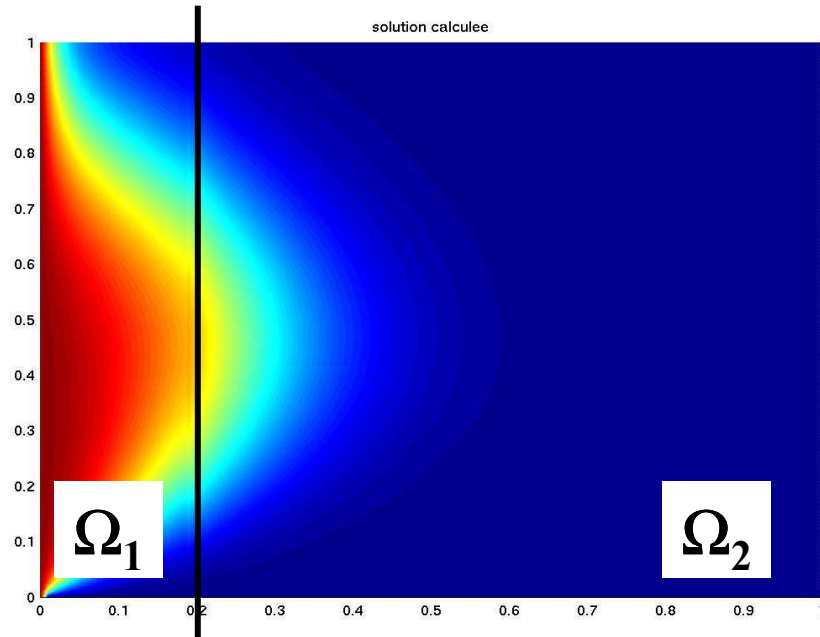
$$\text{Ordre 1 : } \lambda_1^\pm = \frac{a \pm \sqrt{a^2 + 4\nu c}}{2\nu} \pm i \frac{\omega + bk}{\sqrt{a^2 + 4\nu c}}$$
$$\implies \Lambda_1^\pm = \frac{a \pm \sqrt{a^2 + 4\nu c}}{2\nu} Id \pm \frac{1}{\sqrt{a^2 + 4\nu c}} \frac{\partial}{\partial t} \pm \frac{b}{\sqrt{a^2 + 4\nu c}} \frac{\partial}{\partial y}$$

Conditions optimisées (Japhet, 1998; Gander et al., 1999) :

Trouver $\tilde{\lambda}^+, \tilde{\lambda}^-$ qui minimisent le facteur de convergence pour tout k, ω

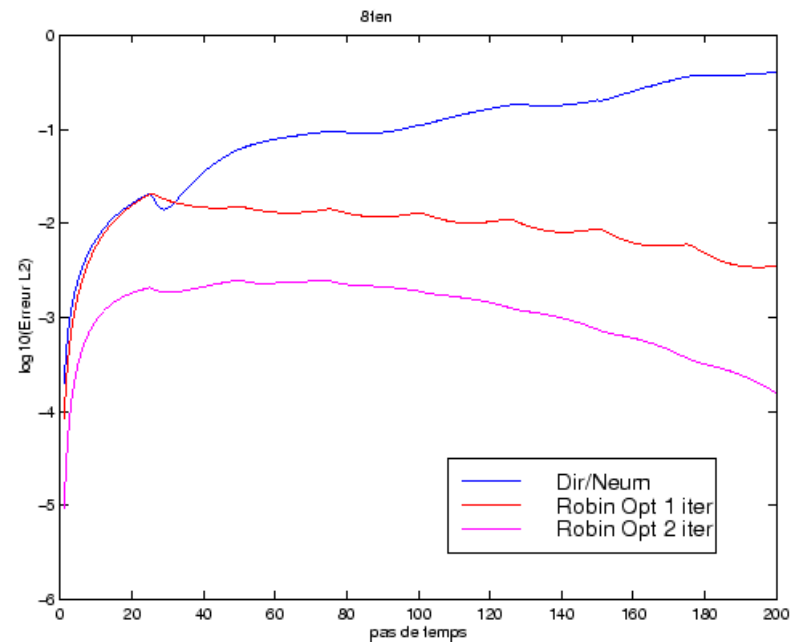
$$\rho(k, \omega) = \left(\frac{\lambda^- - \tilde{\lambda}^-}{\lambda^+ - \tilde{\lambda}^-} \right) \left(\frac{\lambda^+ - \tilde{\lambda}^+}{\lambda^- - \tilde{\lambda}^+} \right) \longrightarrow \min_{\tilde{\lambda}^+, \tilde{\lambda}^-} \max_{k, \omega} \rho(k, \omega)$$

Méthodes de Schwarz : conditions d'interface (4)



Aspects mathématiques et numériques : situation actuelle

- 🔗 recherche en cours sur plusieurs systèmes d'équations
- 🔗 tests de stratégies intermédiaires à 1-2 itérations

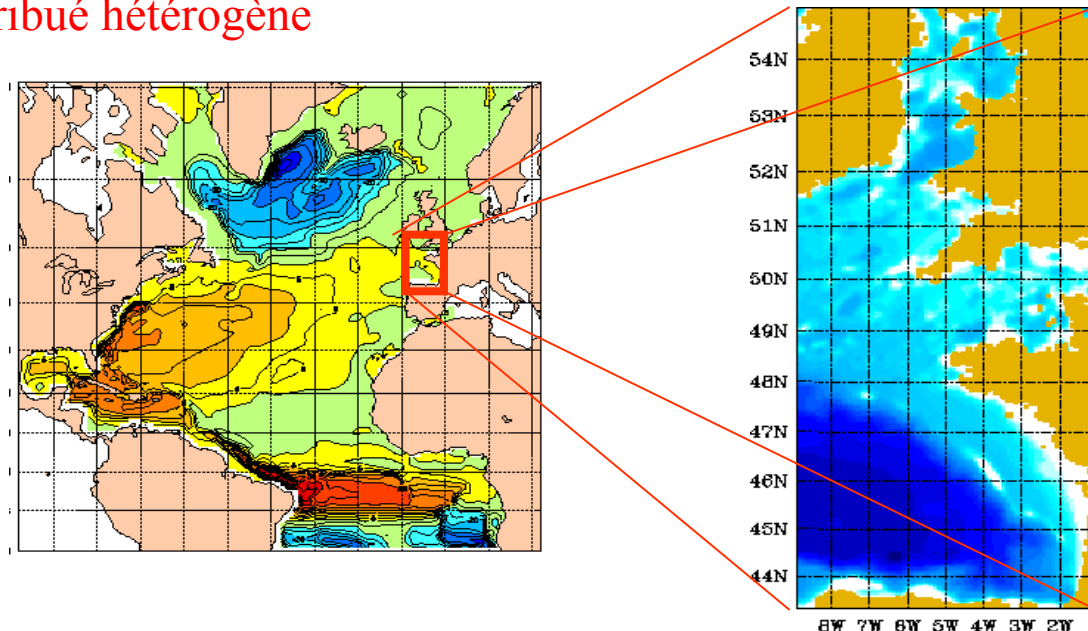


Aspects informatiques

But : Autoriser une **modélisation multi-physiques** ou **multi-échelles** par le **couplage de plusieurs applications**, sans bouleverser celles qui continuent à évoluer par les développeurs respectifs.

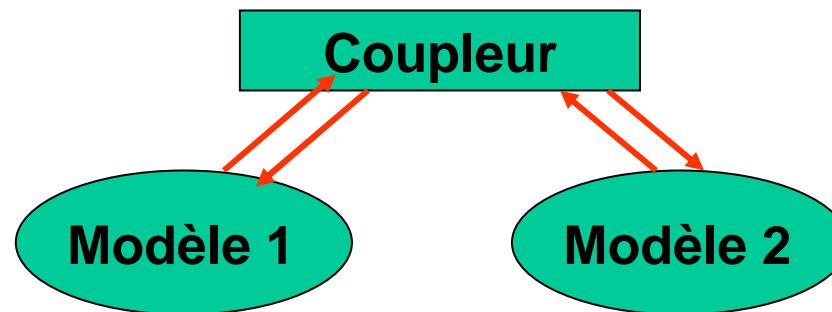
Critères : facilité de développement, capacité à s'adapter à la complexité du problème physique, exactitude et fiabilité, robustesse, efficacité, portabilité

Les applications utilisent des technologies dans un **environnement homogène distribué** ou **non-distribué** (même plate-forme, système, langage, format de données,...) ou dans un **environnement distribué hétérogène**



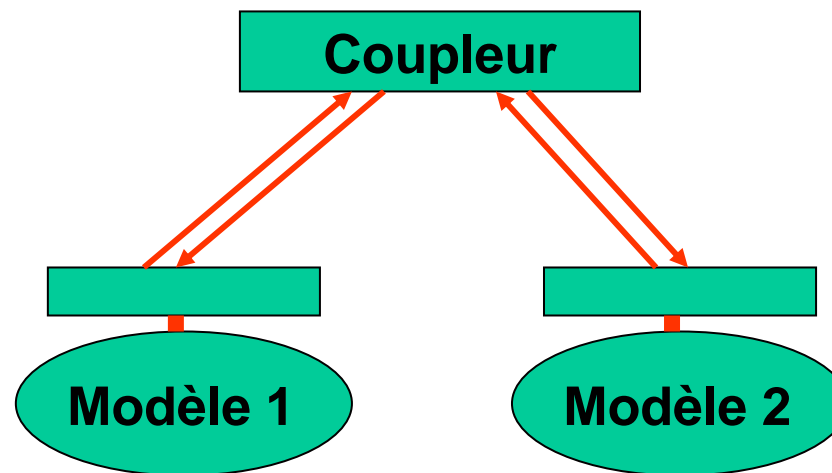
Aspects informatiques : couplage en environnement homogène

- **Intégration de codes existants dans un code unique, la communication entre les codes se faisant par échange de messages (MPI)**
 - Suffisante lorsque les applications sont développées suivant un modèle commun ou par une même équipe
 - **Méthode intrusive** : met souvent en défaut les critères d'exactitude, de fiabilité et de réutilisabilité
 - Difficile sinon impossible lorsque les applications sont volumineuses et de complexité différentes



Aspects informatiques : couplage en environnement homogène

- **Vers une méthode moins intrusive via l'utilisation des concepts objets**
 - **Gestion de la complexité physique** du problème par la définition d'une hiérarchie de classes
 - La **définition d'une interface générique du couplage** (héritage, polymorphisme) apporte une flexibilité pour l'implémentation des méthodes de couplages testées
 - **Fiabilité et réutilisation** grâce à l'encapsulation
 - **Insuffisante pour une gestion simple de l'interopérabilité**



Aspects informatiques: couplage en environnement hétérogène

Couplage via des technologies orientées objets dans un environnement distribué hétérogène

CORBA apporte:

- **Interopérabilité**: indépendance vis-à-vis des plates-formes, des systèmes d'exploitation, du langage de programmation, des formats de données; des couches réseau
- **Approche objets**: définition du couplage par son interface. Langage d'interfaçage indépendant du contexte d'implantation
- **Approche client-serveur**
- **Bus logiciels**: les différents codes vont interagir via un bus et des interfaces

Le bus **CORBA** assure **la transparence des communications** entre les objets distribués et implantés à l'aide de différents langages

JAVA permet l'appel de méthode à distance (Remote Method Invocation) mais il est limité à un seul langage de programmation

Aspects informatiques: couplage en environnement hétérogène

Les inconvénients pour nos applications de calcul scientifique :

- Pas d'implémentation efficace d'objet CORBA parallèle: c'est le processus maître de l'objet CORBA qui communique via le bus (PACO ?)
- Pas de projection du langage d'interface sur le langage Fortran ce qui implique une encapsulation dans des objets C++ ou Java
- Peu adapté à la communication de gros volumes de données entre objets distribués

Développement de cas tests simples permettant de résoudre les difficultés propres à nos applications

Application en océanographie

Sensibilisation à la conception objets et à CORBA : formation permanente CNRS, présentation Mirage

Adéquation de la plateforme Mirage

Mise au point, exploitation “intermédiaire”, expérimentations

- ✘ un important travail de développement, préalable à toute exploitation
- ✘ des modèles de taille “intermédiaire” (qq heures à qq jours de calcul)
- ✘ mise au point de versions parallèles et de versions Corba