

Modélisation 3D du champ acoustique autour d'un véhicule par méthode TLM : préliminaires

Guillaume Dutilleux
LRPC de Strasbourg

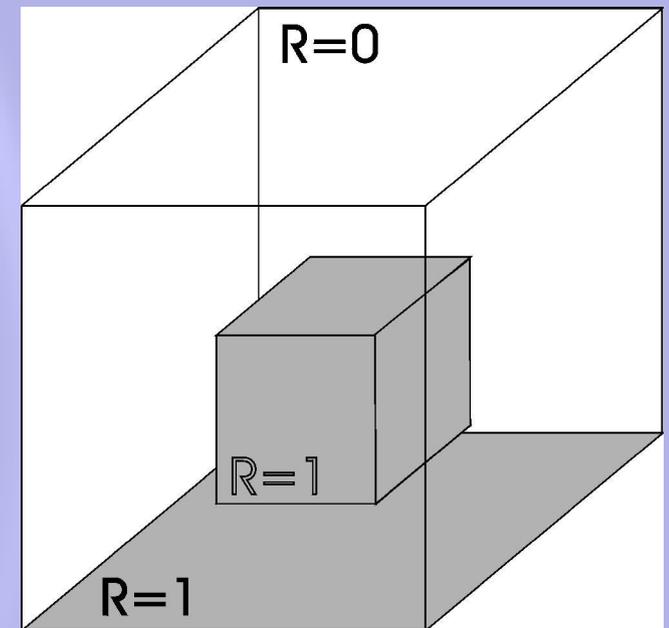


Plan de l'exposé

- Introduction
- Le principe d'Huygens
- Théorie de la TLM en acoustique
- TLM 3D – implantation et validation
- Modélisation d'un véhicule
 - Mailler une géométrie complexe
 - Traiter un problème de grande dimension
- Conclusions et perspectives

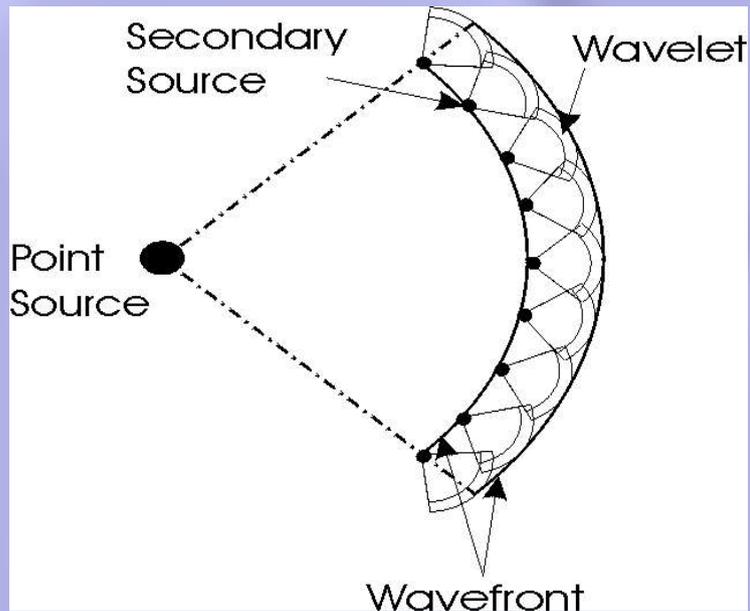
Introduction

- Contexte : développement de la mesure en continu du bruit de roulement
- Evaluer les contributions parasites
 - Roues non instrumentées
 - Réflexions dues à l'environnement
- Difficile de dissocier les sources expérimentalement : simuler
- Un problème 3D non réductible
- Disponibilité d'un code TLM au LRS

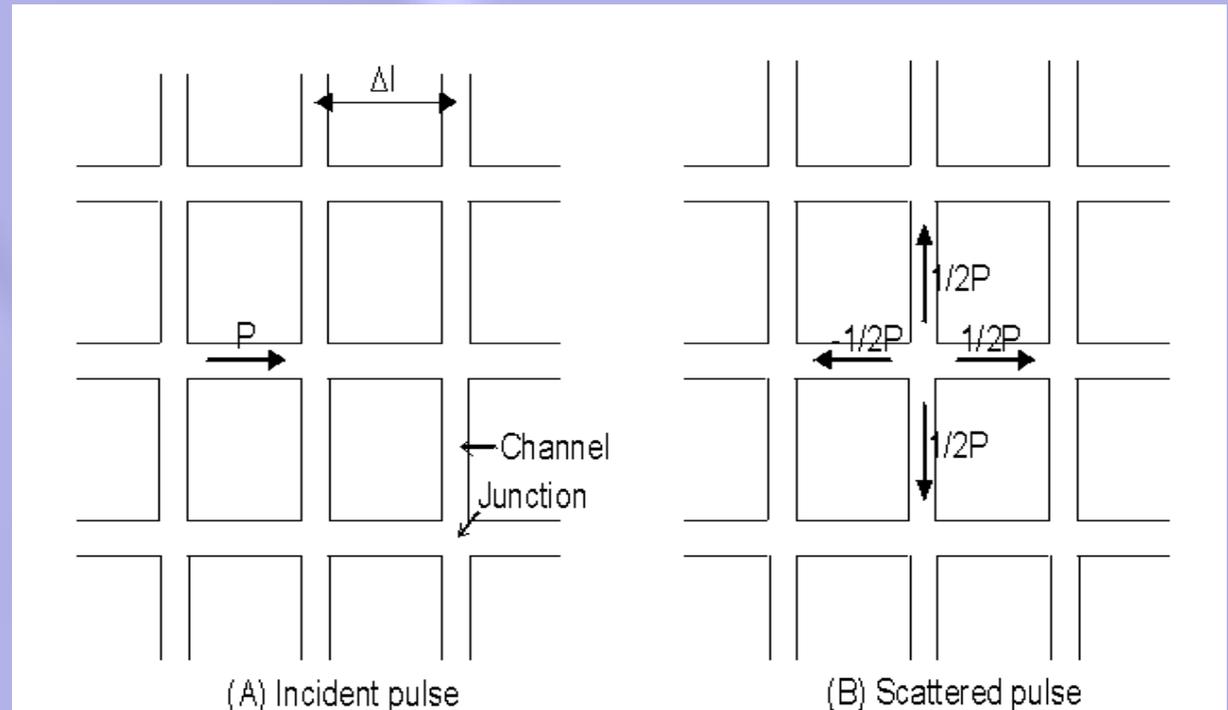


The Huygens' principle

- Huygens 1690



- Computer age (acoustics case)



TLM : Scattering matrix (2D)

- Impedance discontinuity

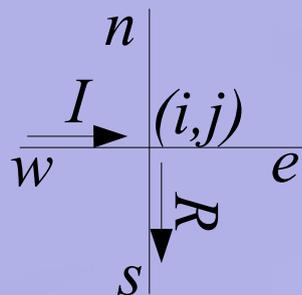
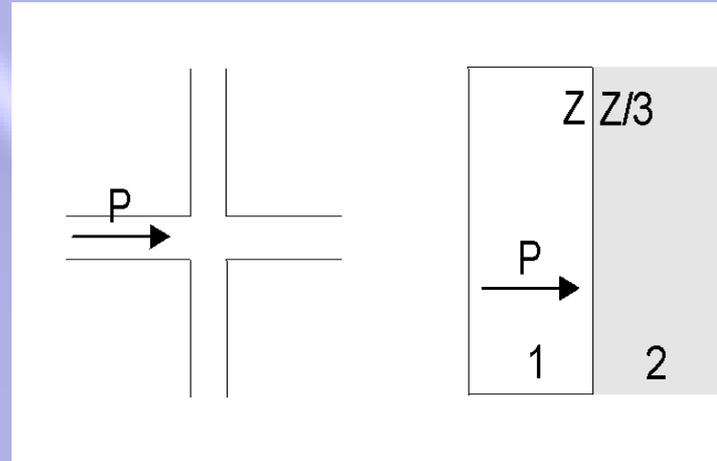
- Reflection coefficient

- 2D : $R = \frac{Z_2 - Z_1}{Z_2 + Z_1} = \frac{Z/3 - Z}{Z/3 + Z} = -\frac{1}{2}$

- Energy conservation

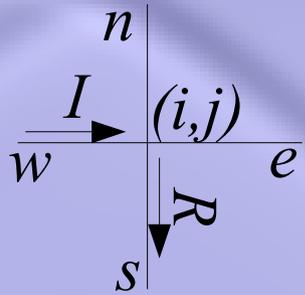
- Transmission coefficient : $R^2 + T_1^2 + T_2^2 + T_3^2 = R^2 + 3T^2 = 1 \rightarrow T = \frac{1}{2}$

- Scattering matrix



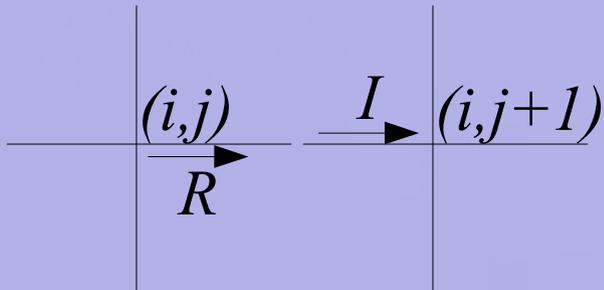
$${}_{t+\Delta t} \begin{bmatrix} R^w \\ R^n \\ R^e \\ R^s \end{bmatrix}_{i,j} = \frac{1}{2} \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix} {}_t \begin{bmatrix} I^w \\ I^n \\ I^e \\ I^s \end{bmatrix}_{i,j}$$

TLM : Propagation rules



- Fluid domain

- For all channels



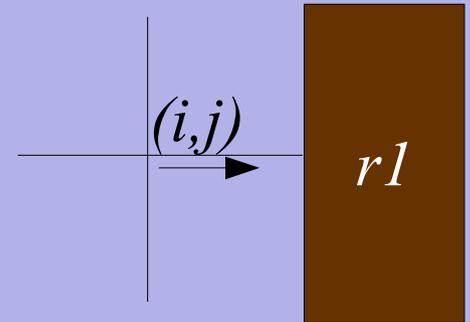
$${}_{k+1}I_{(i,j)}^e = {}_kR_{(i,j+1)}^w$$

$${}_{k+1}I_{(i,j+1)}^w = {}_kR_{(i,j)}^e$$

Boundaries

- Side

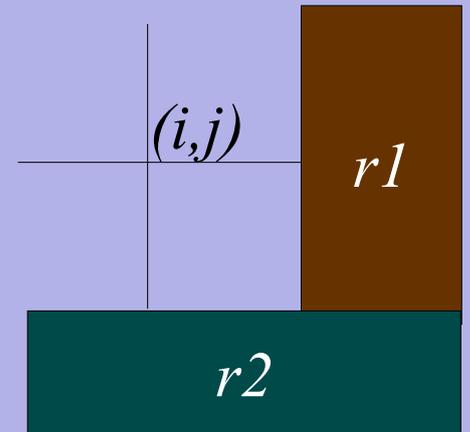
$${}_{k+1}I_{(i,j)}^e = r1 {}_kR_{(i,j)}^e$$



- Corner

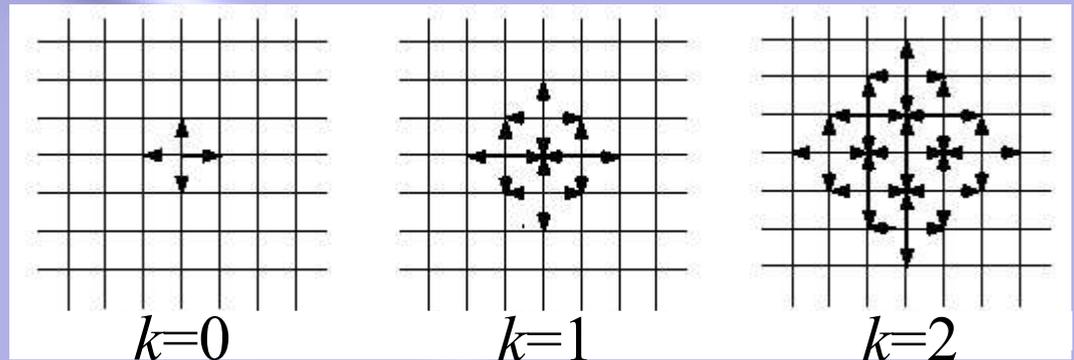
$${}_{k+1}I_{(i,j)}^e = r1 {}_kR_{(i,j)}^e$$

$${}_{k+1}I_{(i,j)}^s = r2 {}_kR_{(i,j)}^s$$

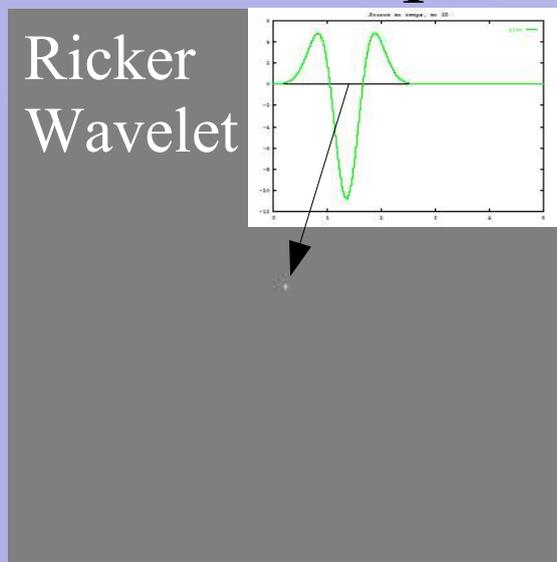


TLM : Putting it all together...

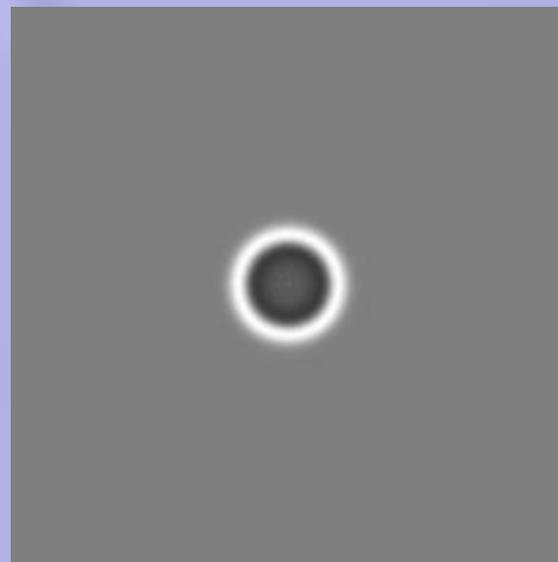
- Microscopic level



- Macroscopic level



$k=4$



$k=50$



$k=100$

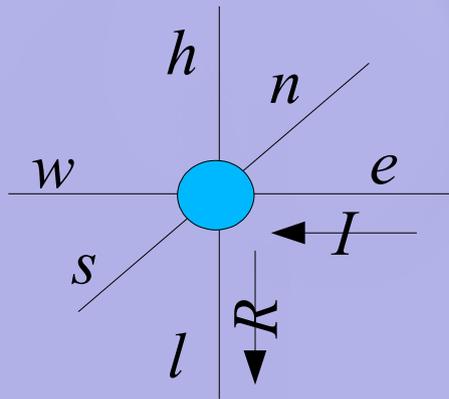
TLM : possibilités

- **Fluide inhomogène**
 - Isotrope / Turbulence : simple en principe
 - Anisotrope : pas de solution satisfaisante pour l'instant
- **Aspects numériques**
 - Stabilité : pas d'inversion de matrice
 - Répartition facile du calcul
- **Frontières**
 - Géométrie complexe
 - Condition de Sommerfeld et frontières dépendantes de la fréquence à mettre au point
- **Symétries**
 - cartésienne
 - cylindrique

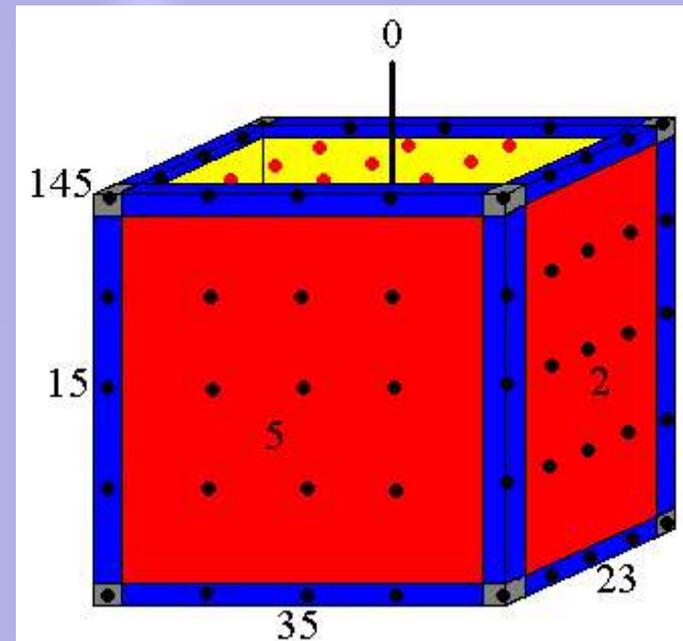
TLM 3D cartésienne et homogène

- Matrice de dispersion

$${}_{k+1} \begin{bmatrix} R^w \\ R^n \\ R^e \\ R^s \\ R^h \\ R^l \end{bmatrix} = \frac{1}{3} \begin{bmatrix} -2 & 1 & 1 & 1 & 1 & 1 \\ 1 & -2 & 1 & 1 & 1 & 1 \\ 1 & 1 & -2 & 1 & 1 & 1 \\ 1 & 1 & 1 & -2 & 1 & 1 \\ 1 & 1 & 1 & 1 & -2 & 1 \\ 1 & 1 & 1 & 1 & 1 & -2 \end{bmatrix} {}_k \begin{bmatrix} I^w \\ I^n \\ I^e \\ I^s \\ I^h \\ I^l \end{bmatrix}$$

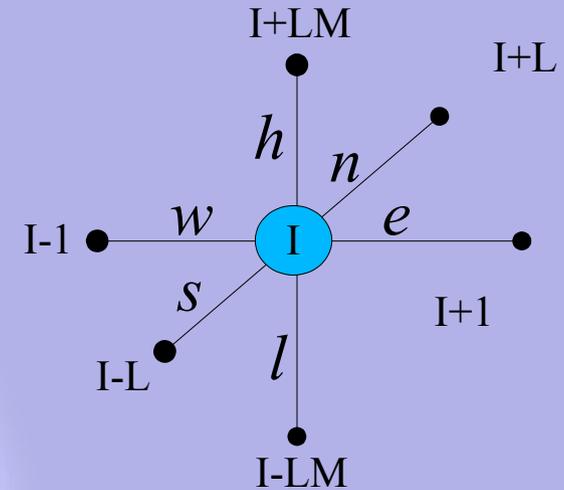


- Conditions de frontière
 - 6 de « côté »
 - 12 d' « arête »
 - 8 de « sommet »



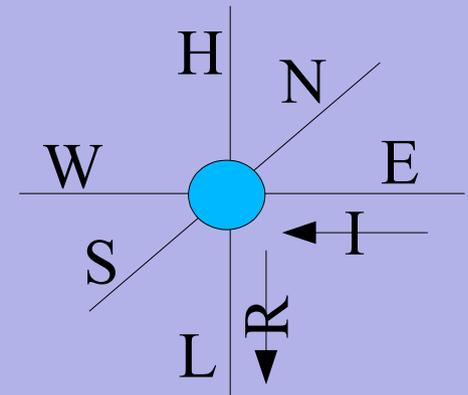
3D TLM in Scilab

- Conventions
- Data structure
 - 1 matrix for geometry
 - 1 matrix by direction I/R
 - 1 matrix by orientation (E,W,N...)
- Use of Scilab language
 - Global matrix index $I = i + (j - 1)L + (k - 1)LM$
 - `hypermatrix` (Scilab 3.0 or later)
 - *same implementation for 2D and 3D*
 - `find` → grouping nodes by propagation rule



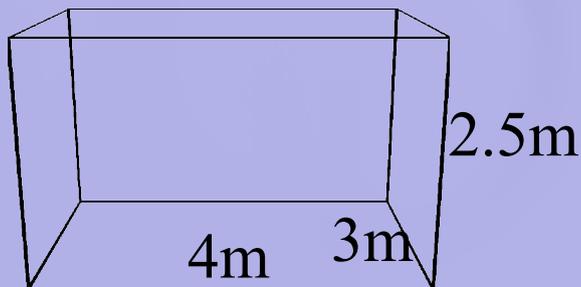
3D TLM in Scilab : main loop

- For each iteration do :
 - When it applies : input energy in the grid
 - Compute the overall pressure
 - $P = 1/3 * (WI + NI + EI + SI + LI + HI) ;$
 - Compute the scattered components
 - $WR = 1/3 * (-2 * WI + NI + EI + SI + LI + HI) ;$
 - $NR = 1/3 * (WI - 2 * NI + EI + SI + LI + HI) ;$
 - $ER = 1/3 * (WI + NI - 2 * EI + SI + LI + HI) ;$
 - $SR = 1/3 * (WI + NI + EI - 2 * SI + LI + HI) ;$
 - $LR = 1/3 * (WI + NI + EI + SI - 2 * LI + HI) ;$
 - $HR = 1/3 * (WI + NI + EI + SI + LI - 2 * HI) ;$
 - Apply the propagation rules by group



3D TLM : Validation against Sabine's theory

- Shoebox-shaped cavity
 - Reflection coefficient
 - $r=0.8$
 - Dimensions



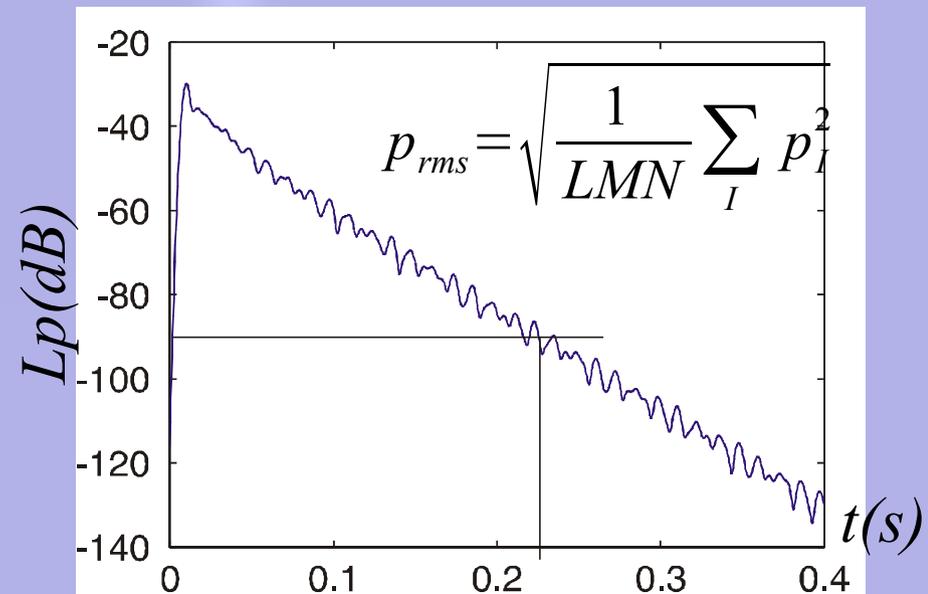
$$\Delta l = \lambda / 10$$

- Reverberation time

- Sabine

$$RT = \frac{0.16 V}{S(1-r^2)} = 0.23 \text{ s for } r=0.8$$

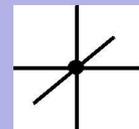
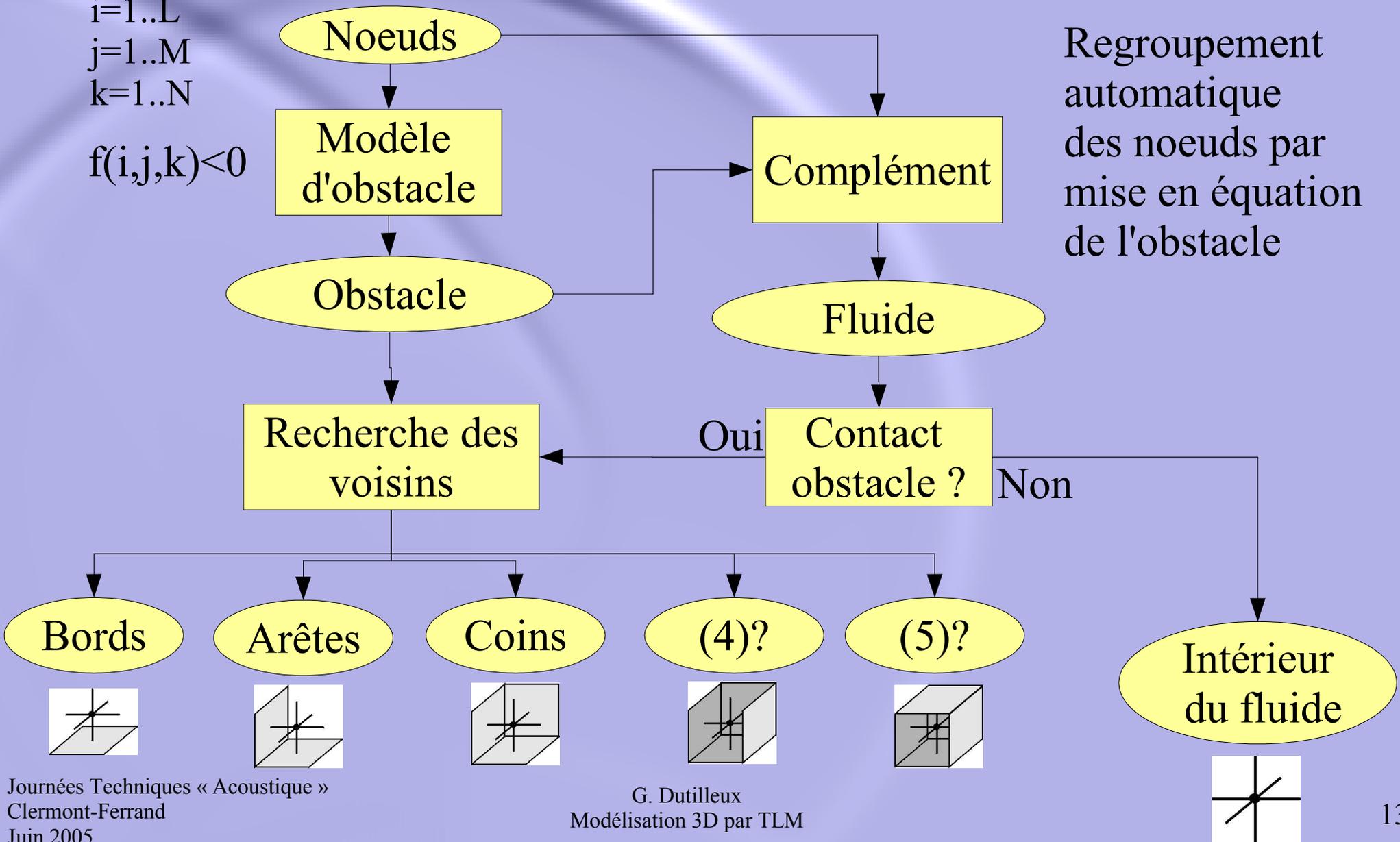
- TLM



Représenter une géométrie complexe

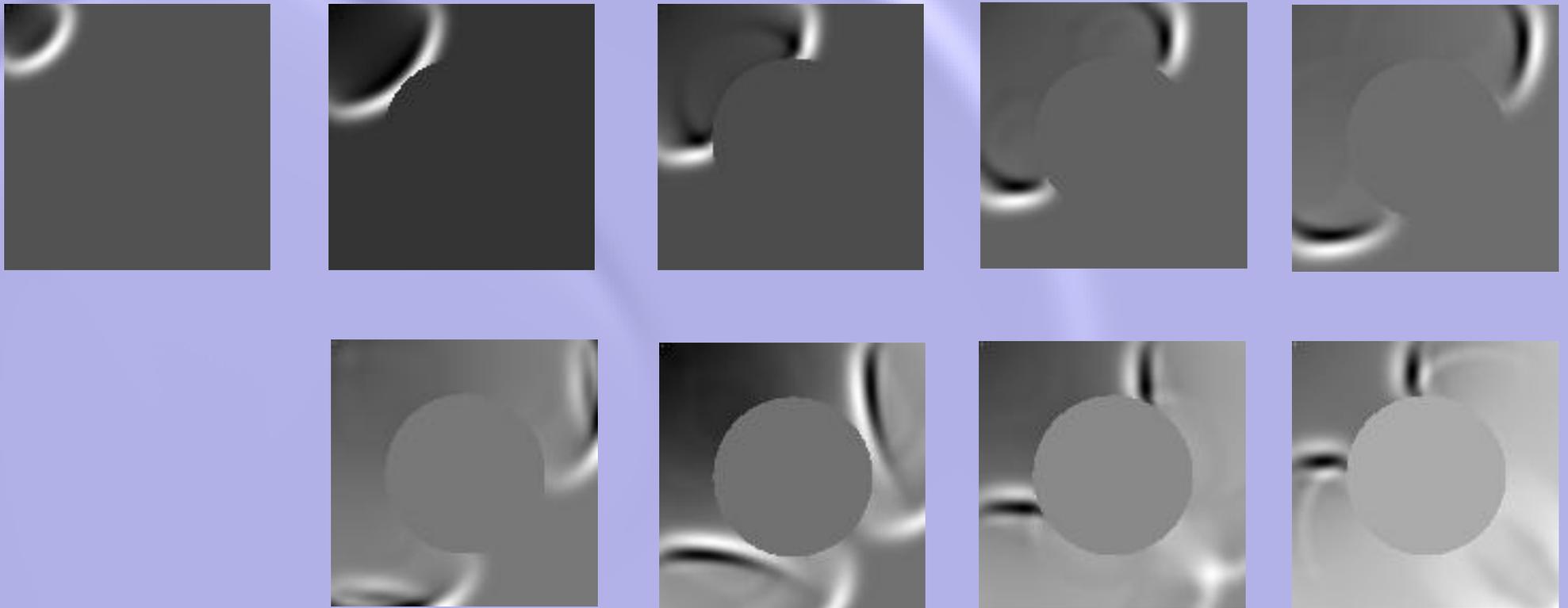
$i=1..L$
 $j=1..M$
 $k=1..N$
 $f(i,j,k) < 0$

Regroupement automatique des noeuds par mise en équation de l'obstacle

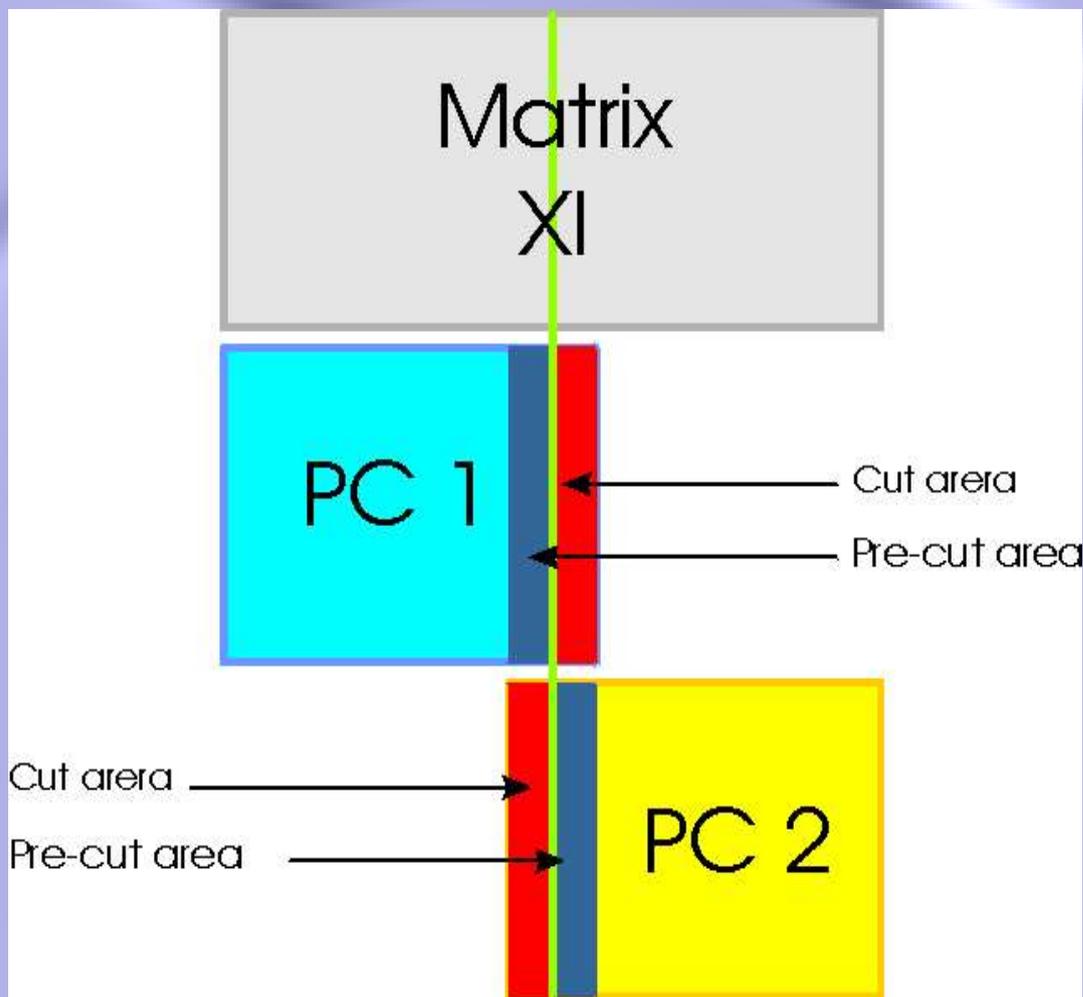


Représenter une géométrie complexe

- Test : ondelette de Ricker en 2D sur un disque



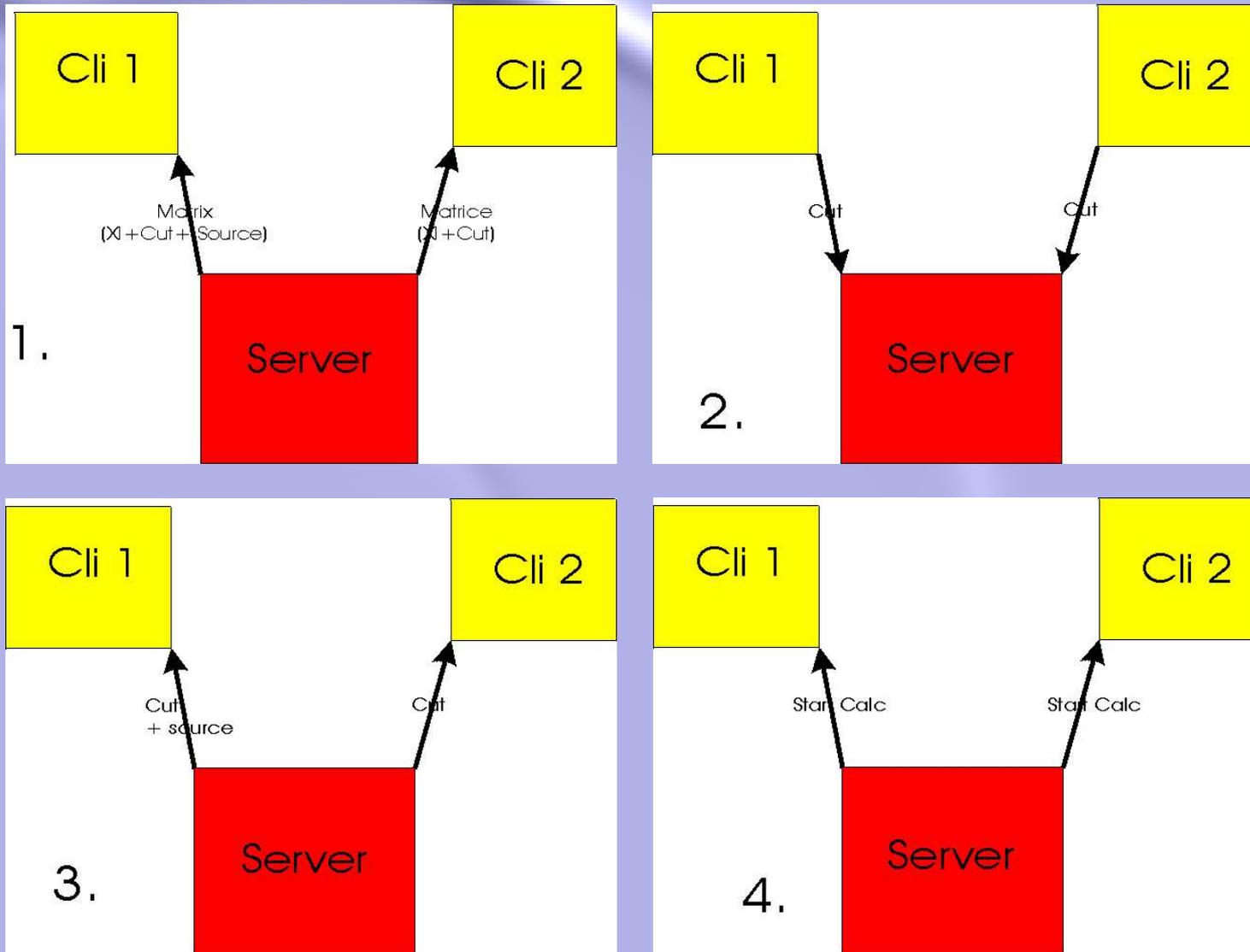
Traiter un problème de grande taille : la TLM *distribuée*



- matrix is sent to PC1 and PC2 by the server
- points located in the cut zone are never computed
- clients send their own pre-cut zone
- when the server has received both, it dispatches the one from PC1 to PC2
- pre-cut zone of PC1 becomes the cut zone of PC2

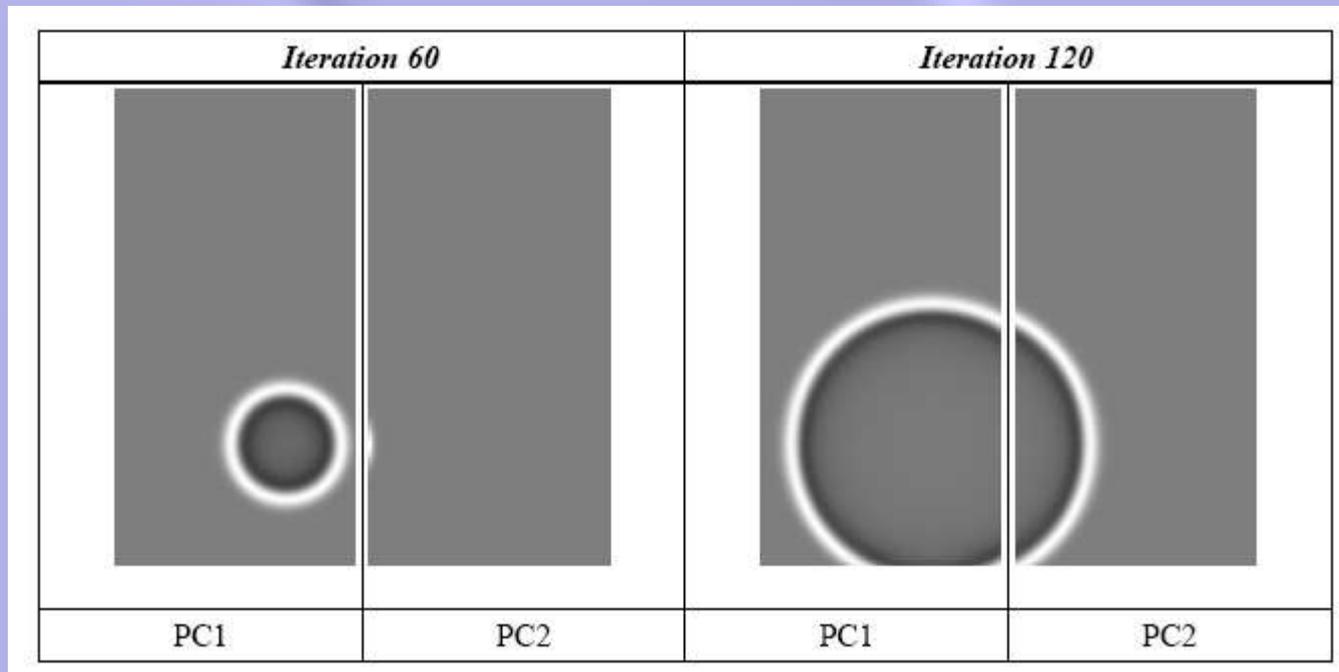
[Dutilleux-Waechter2004]

TLM distribuée : chronologie



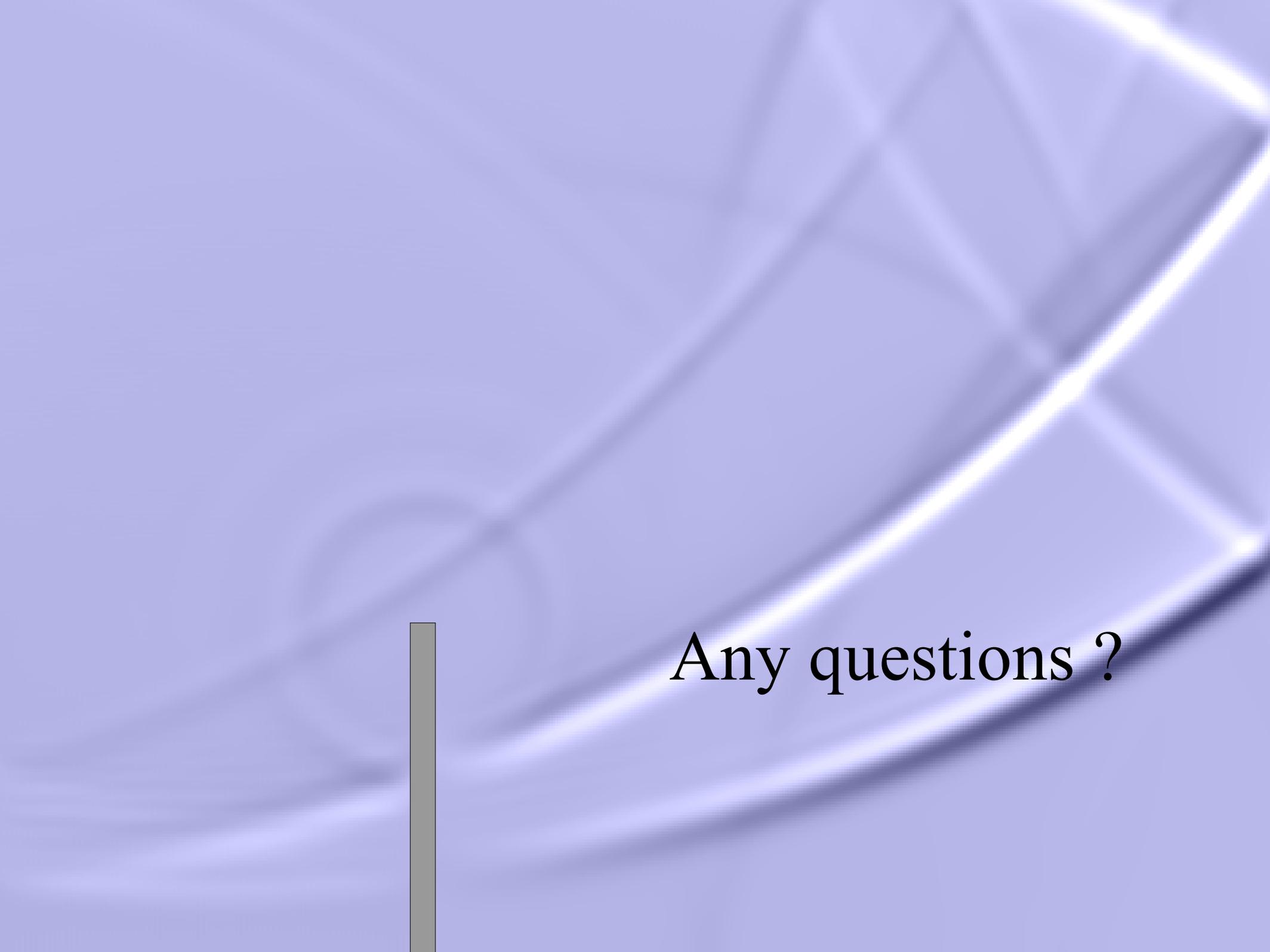
Distributed TLM : operation in 2D

- The simulation has been performed with two computers running Windows 2000 (PC1) and Windows NT4 (PC2) :



Conclusions et perspectives

- Modèle TLM 3D cartésien homogène disponible
- Obstacles complexes : détermination des types de noeuds
 - Approche par mise en équation
 - Application en 2D réalisée
 - Généraliser à 3D
- DTLM : une implémentation répartie interopérable
 - Problèmes de grande taille sur réseaux hétérogène
 - Pas encore appliquée en 3D
- D'autres applications possibles



Any questions ?