

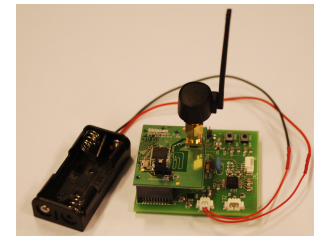
# Simulation de réseaux de capteurs et mobiles sur processeurs graphiques,

Bernard Pottier\*  
Thibault Failler+, Nicolas Melot+, ..

*Université de Brest (UBO)*  
*\*LabSTICC, UMR 3192*  
*+ Master info. Dept Informatique, UFR Sciences*

# Cadre des recherches

- **UMR LabSTICC** (2008) : recherche « des capteurs à la connaissance »
- **WSN** : équipe en gestation sur réseaux de capteurs : LabSTICC+..
- **PucesCom** : projet régional U.E.Bretagne
- **ReCap Fac. Sciences pluridisciplinaire**: info., élec/télécom, chimie, physique, géophysique ..
- **Relations internationales** : Pk, V-N, All, Esp, Sn...



# Capteurs, Réseaux de capteurs et applications en agriculture

DES FERMES EXPÉRIMENTALES, OU L'ON PILOTE L'USAGE DES RESSOURCES, AUX INDUSTRIES AGRO ALIMENTAIRES, LES CAPTEURS SONT LA CLÉ DE LA PRODUCTIVITÉ ET DE RELATIONS MEILLEURES AVEC NOTRE ENVIRONNEMENT.

Le 30 Septembre, en amphi G :

- 14h, Mr Aurélien Jacquot, Centre Cemagref d'Aubiere:  
'Les réseaux de capteurs sans fil au service des applications agro-environnementales',
  - Plateformes matérielles couramment utilisées dans ces réseaux
  - Présentation des aspects logiciels pour les RCSFs appliqués à l'agriculture
  - Quelques exemples d'applications concrètes dans l'agriculture
  - Retour d'expériences avec les contraintes rencontrées et les résultats obtenus
- Téléchargement : PDF résumé de l'intervention + présentation Cemagref.
- Téléchargement : planches de l'exposé


Projet

explo-  
ratoire

ReCap

**Le Cemagref**

- ▶ **Le Cemagref : EPST**
  - 9 centres sur la France + 2 antennes
  - 20 UR, 5 UMR, 1600 personnes dont 200 doctorants
- ▶ **3 départements scientifiques**
  - Eaux
    - Ressources, milieux, usages et risques
  - Écotechnologies
    - Réseaux, épuration, déchets
  - Territoires
    - Développement territorial, biodiversité, risques et vulnérabilités
- ▶ **Travaux entrepris**
  - Systèmes d'information et de collecte géographique
  - Robotique mobile
  - Economie
  - Pratiques d'élevage
  - Machinisme agricole



Brest  
Penfeld

Salon  
Dévelop.  
Durable

9 posters  
Gd Public

ForumDurable



Salon du Durable

Posters 'Réseaux de Capteurs sans fil' présentés au Salon du développement durable.  
<http://www.salond-d-ouest.fr/> Mars 2010

[▶ Lancer le diaporama](#)

[📡 S'abonner](#)



Animations  
Web

Ligne  
de  
Bus  
« smart »

**NetGen simulation**

**Bus and bus stops**

A mobile interacts with sensor networks fetching  
information on the fly

# Animation: déploiement côtier 2 portées différentes

VisualWorks NonCommercial /home/ber... Pick points Mobile

File Edit Simulation

System Browse Debug Printer

seau : 8 Node : 2  
seau : 9 Node : 3  
seau : 10 Node : 5

Options

Outputs

- Graphic layout
- Occam code
- Dot File
- Dot graph
- Animated view
- Display code
- Quannet model
- Code code
- Mobile

Generate

Generate/Compile

Mobile is moving and processing each encountered network  
The result is a bounding box produced by distributed computations, shown with sensors and path

Mise en mémoire tampon : 100,00%

2:00 /

# Plan de l'exposé

- NetGen, un flot de conception « top down »
- Topologies, simulation et exécution
- Mobiles et capteurs
- Simulation sur GPU
- Conclusion



# I Flot descendant, et ses étapes



# Approche flot descendant

## Applications capteurs

- Evènements locaux imprévisibles, alertes
- Défaillances des systèmes et des communications
- Nécessité de considérer les pires circonstances
- Supervision indispensable

**Flot descendant**

## Réseaux généraux

- Notion de service présent ou absent prépondérante
- Défaillances tolérées réparées
- Best effort
- Pas de supervision

**Approche ascendante**

# Etapes et boucles de conception

1. **Choix d'implantations physiques**
2. Production automatique de la structure logique du réseau (topologie)
3. **Programmation du comportement collectif**
4. Assemblage du système en un simulateur
5. Simulation, analyse, rebouclage vers 1, ou 3.
6. Production du code des capteurs et tests



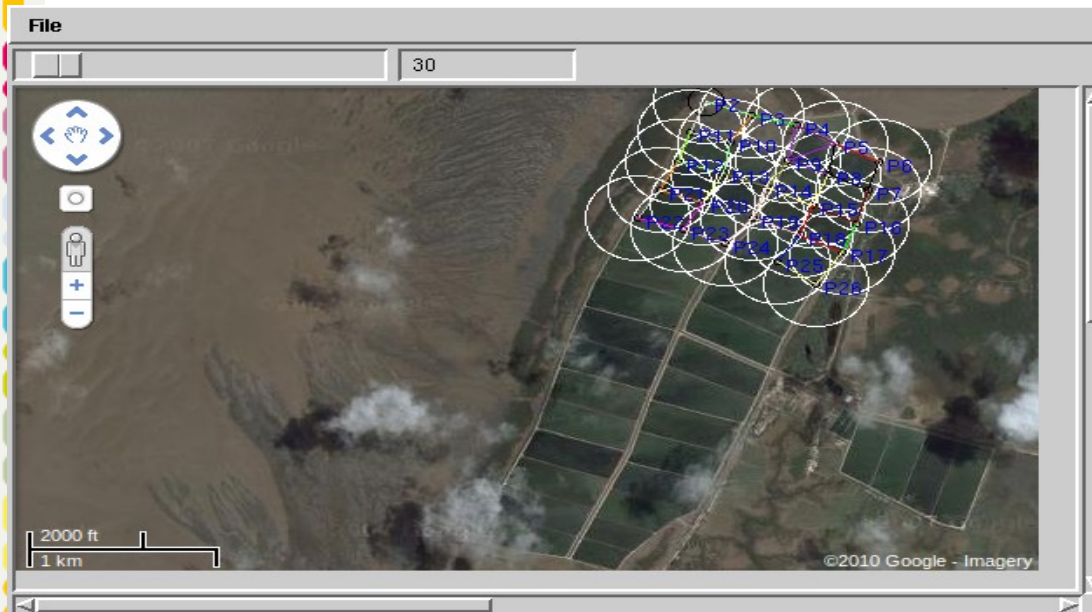
# *Etape de saisie environnement*

## 1. Description physique :

- Environnement de déploiement
- Couverture du terrain, nombre de capteurs, passerelles, communications
- Redondances
- Risques

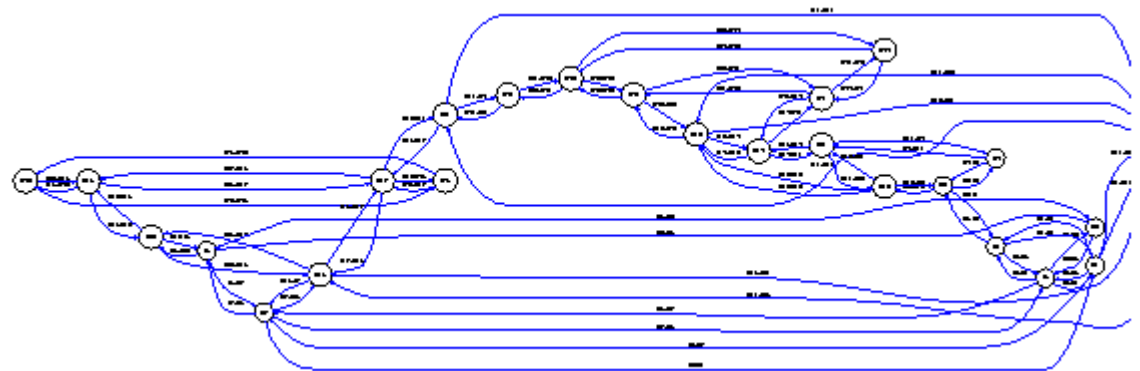
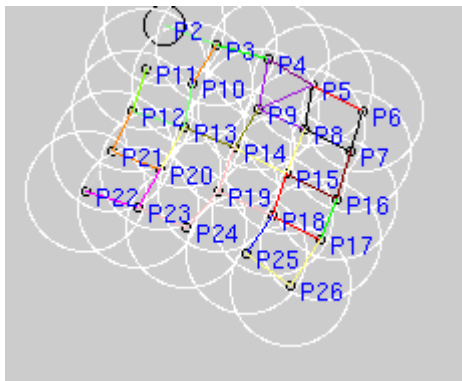


# Etape de construction réseau



## 2. Structure logique

- Placements, connectivités
- Modélisation en système concurrent



# Etape de programmation

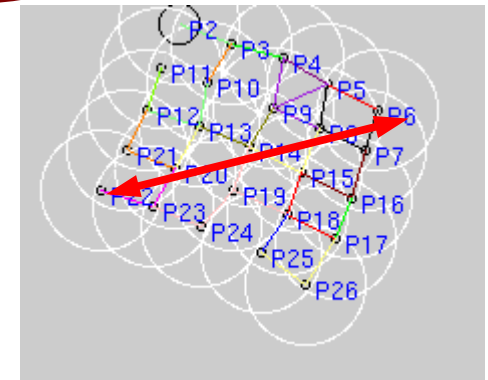
## 3. Comportements

- Modèle synchrone
- Algorithmes distribués
- Phases structurantes
  - Calcul de diametres
  - Leaders, étiquetage
  - Arbres, routes, routages

Répéter (maxNode)

```
SEQ
PAR
  PAR i=0 FOR SIZE in
    in[i] ? msgIn[i]
  PAR i=0 FOR SIZE out
    out[i] ! msgOut[i]
  CalculDiametre(msgIn,msgOut)
```

**Diametre = ?**  
*Nombre de pas  
pour propager  
l'information*



# Etape d'assemblage: simulateur

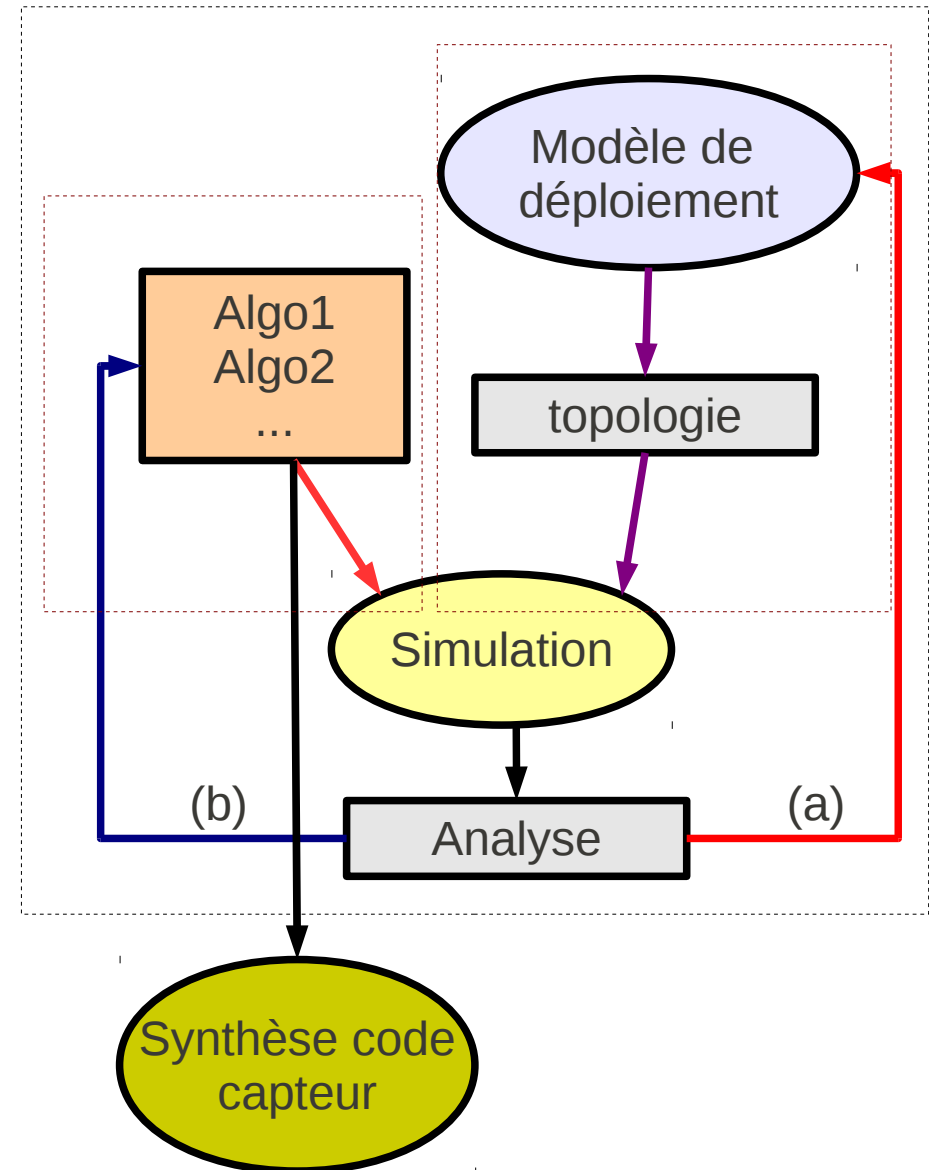
## 4. Algorithmes + *topologies* = systèmes

- 2 degrés de liberté
- Les algorithmes collectifs découvrent les caractéristiques topologiques *dynamiquement*
- Génériques, ils fonctionnent quelque soit la topologie
- Usage de topologies *arbitraires et complexes*
- En composant topologie et algorithmes on obtient un *simulateur représentant un déploiement.*

# Boucles dans le flot

## 5. Mise au point

- Utilise les 2 degrés de liberté
- Ajuster la topologie (a)
- Affiner les algorithmes (b)
- Analyser les résultats de simulation





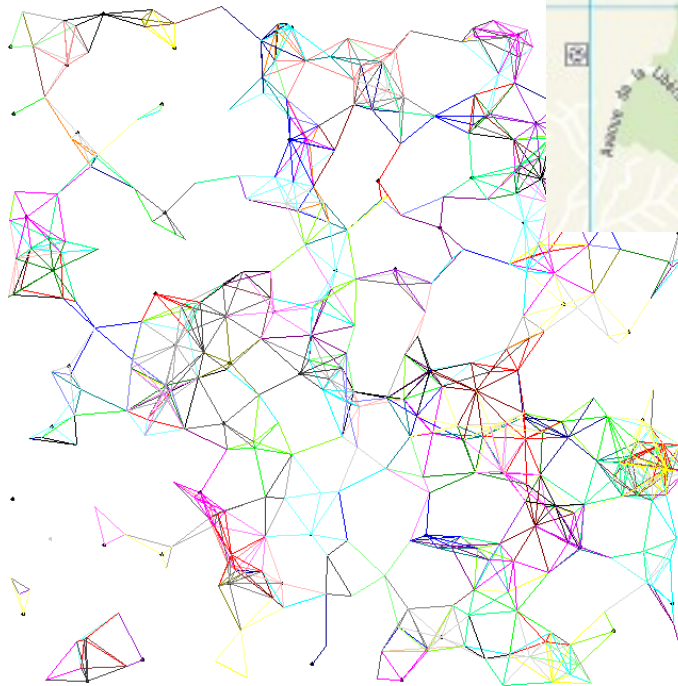
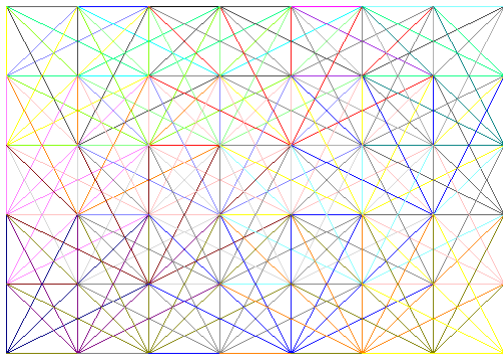
# II Topologies, simulation, exécution



# Réseau: régulier/Aléatoire/Spécifique

## 1. Modèle logique

- Saisie interactive, textes
- Distributions aléatoires,
- Grilles,
- Pipelines



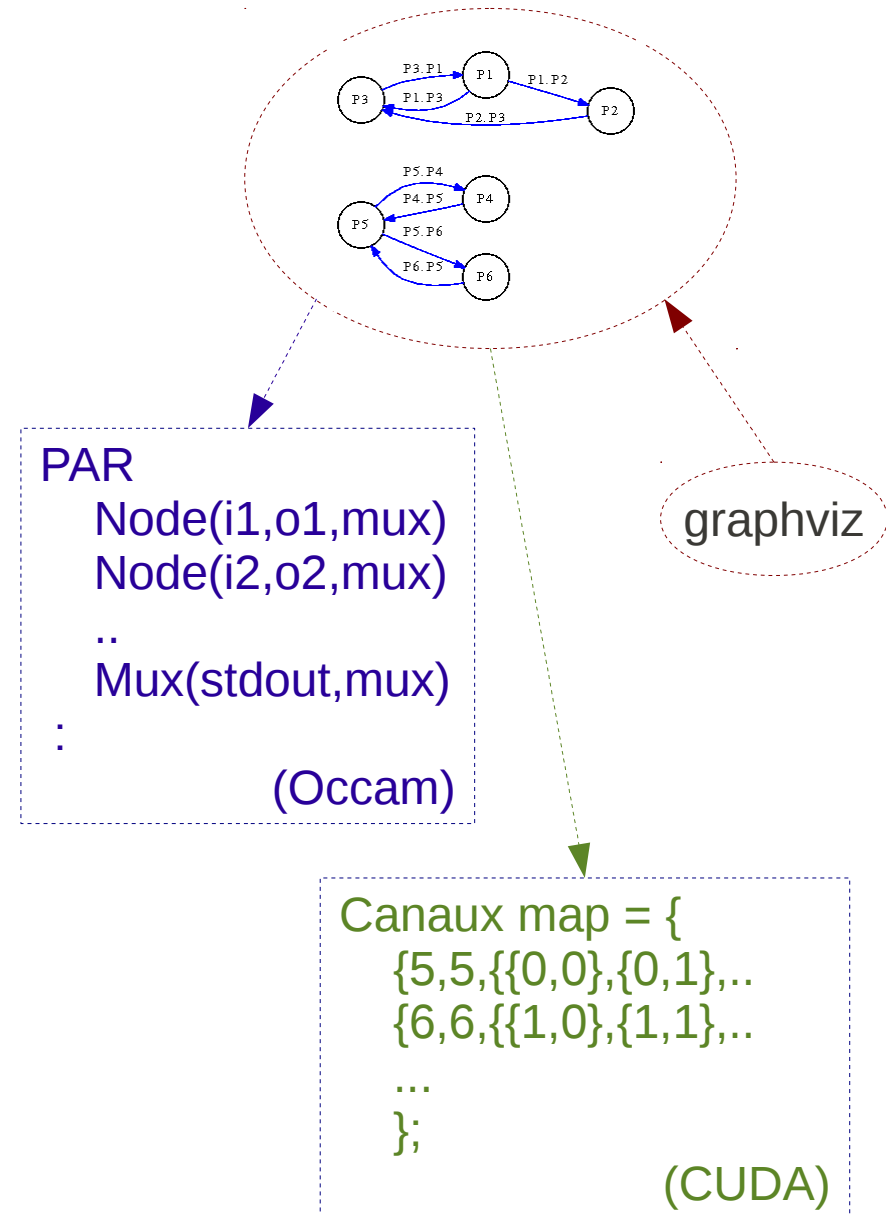
netSampleSpec

```
P1 { P2 P3 } Node
P2 { P3 } Node
P3 { P1 } Node
P4 { P5 } Node
P5 { P4 P6 } Node
P6 { P5 } Node
```

# Topologies et traductions

## 2. Un modèle et des traducteurs topologiques

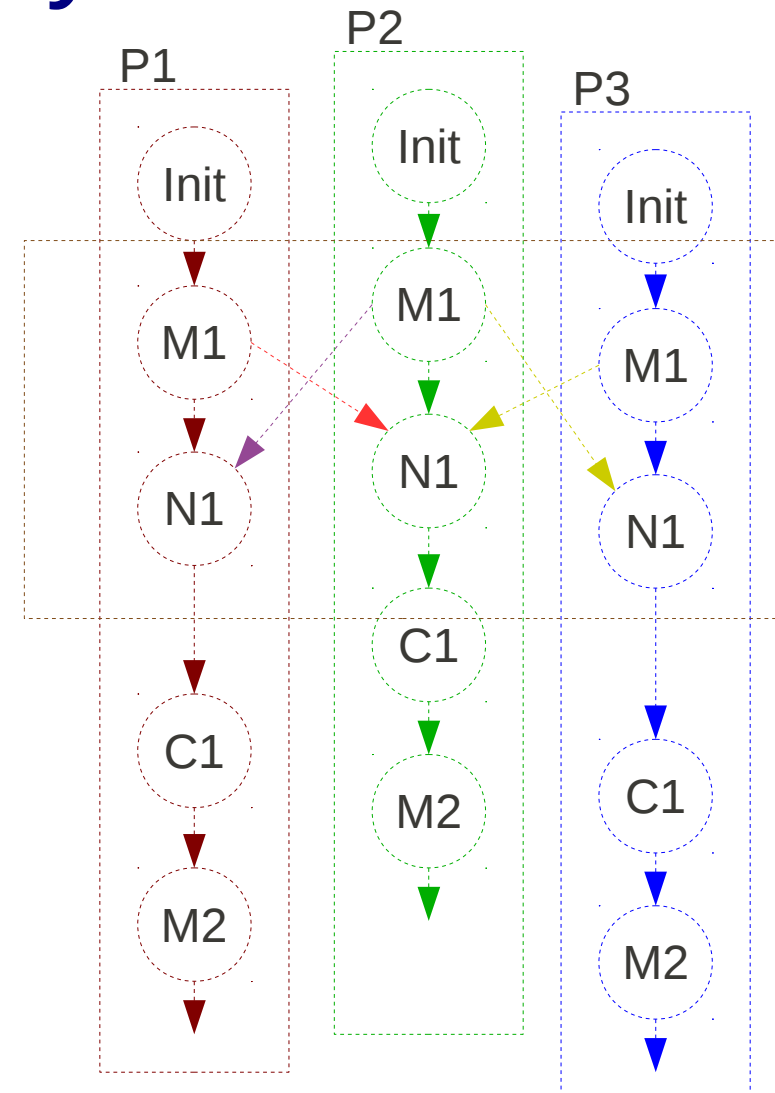
- *Modèle : Graphe, processus, liens*
- *Traduction en structure de processus Occam/CSP, ou*
- *Buffers et adressages mémoires pour CUDA*
- *Graphiques, textes*



# Exécution: modèle synchrone

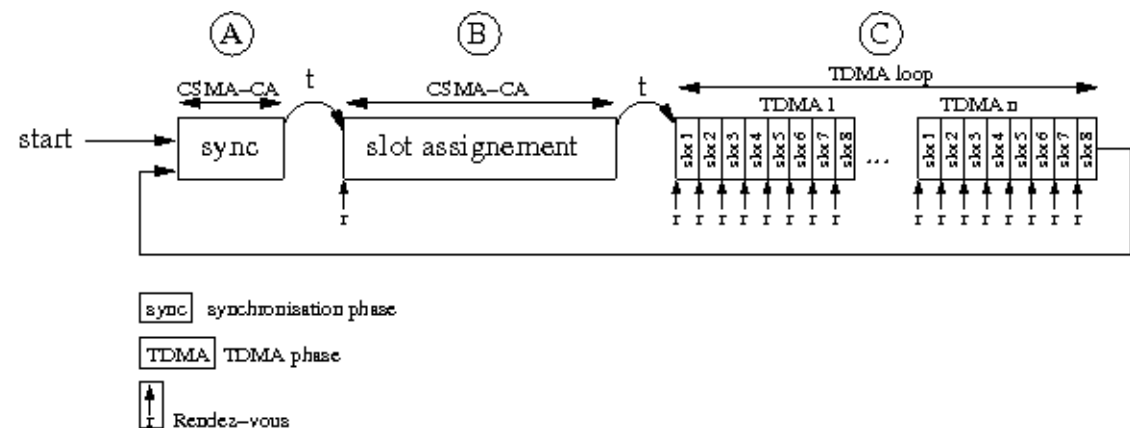
## 3. Comportements

- Modèle synchrone  
(« canaux vides »).  
Répétition de phases de coms MN et de changements d'état C
- Algorithmes distribués :  
diametre, leaders, arbres de recouvrement, routages, consensus..



# TDMA : une implémentation

- TDMA : rendez-vous périodiques pour les communications (on dort entre temps)
- Allocation de slots pour émission réception
- Prises de contact par un protocole asynchrone
- 1 fente temporelle = 1 canal
- Slots libres

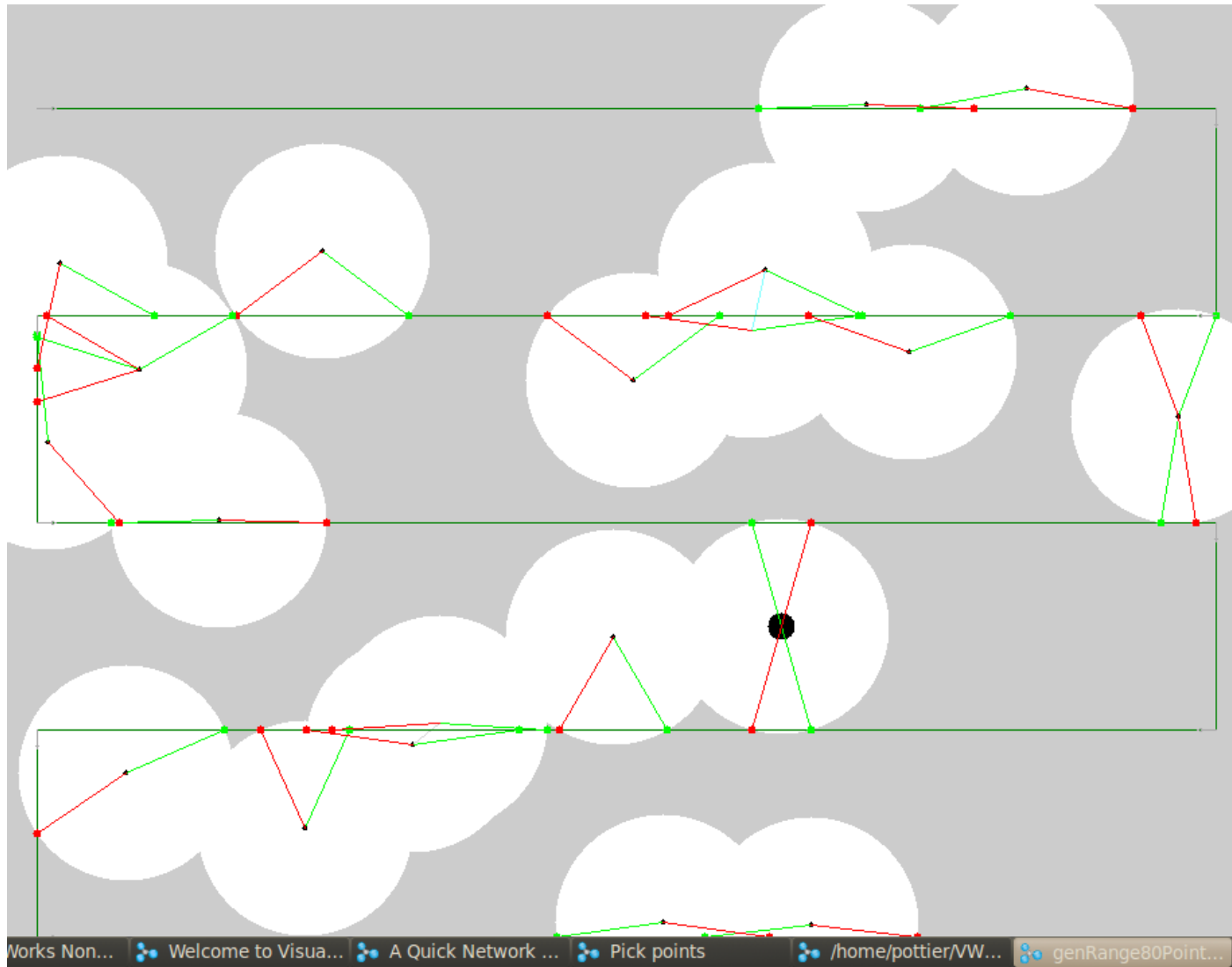


# II Mobiles et capteurs

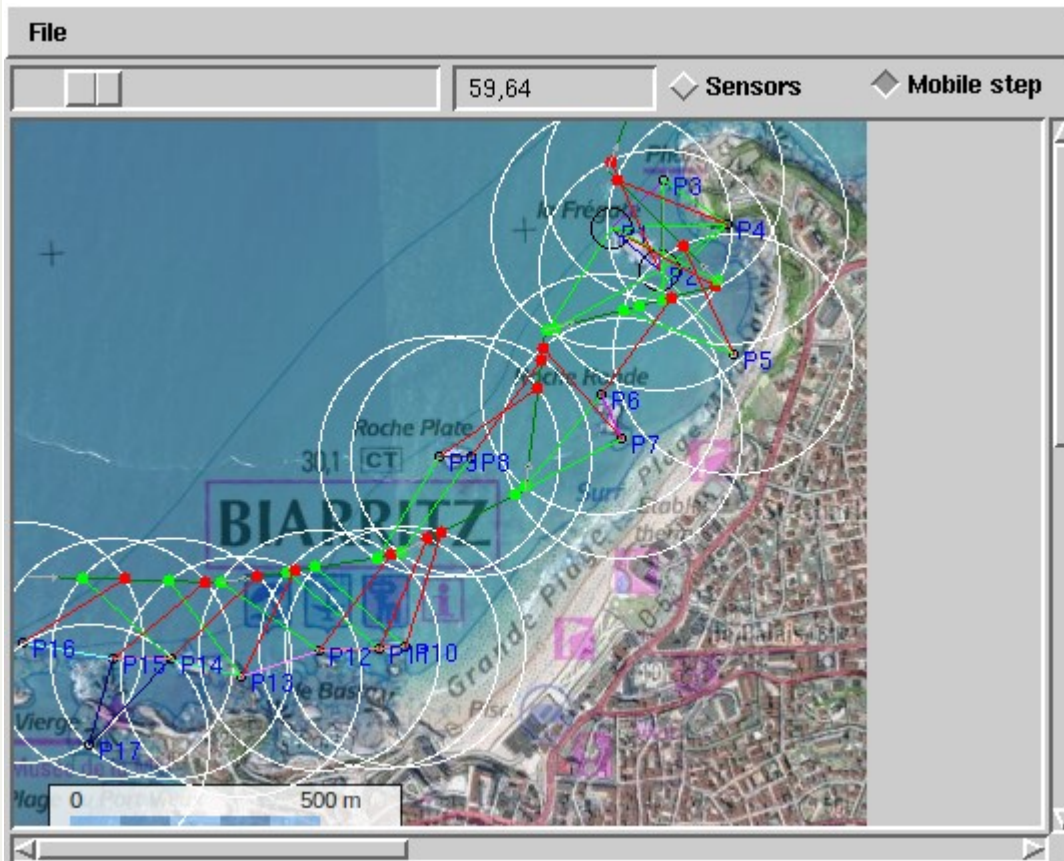
# Classes d'usages

- 1 mobile : *pilotage d'un drone, d'un bateau pour visiter ou en s'appuyant sur des balises sans fil*
- Plusieurs mobiles : *bus circulant sur une ligne, un réseau, personnes à domicile, animaux*
- Mobiles dominants : *flottes, traffics, animaux...*
- Comportement **calculé** : *balayage systématique*
- Comportement **spécifié** : *bus sur sa ligne*
- Comportement **dynamique** : *décisions programmées dépendant de l'historique du mobile*

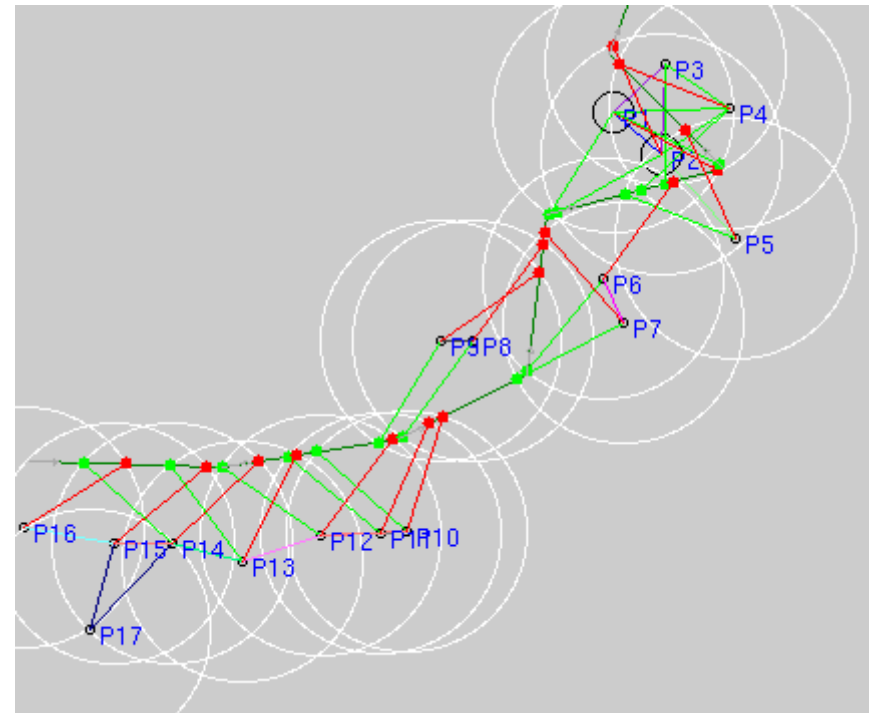
# Parcours calculé (N.Melot)



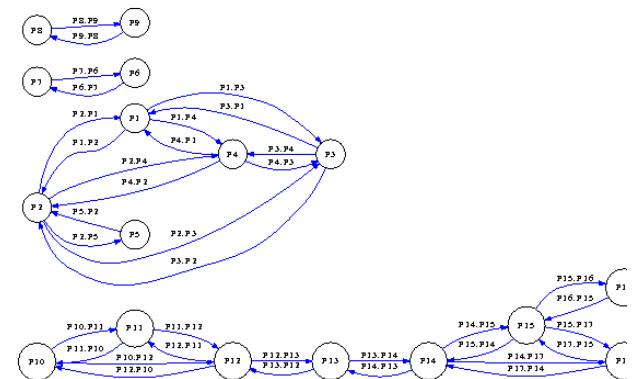
# Parcours spécifié



Portée 250m, 4 réseaux rencontrés



- Définition du système de capteurs
- Définition de la visite (chemin)





# Mobiles : simulation et exécution

- Provision de canaux libres additionnels :
  - Fentes temporelles TDMA supplémentaires
  - Buffers CUDA supplémentaires pour les mobiles
  - (Canaux mobiles Occam-Pi )
- Ces canaux sont « écoutés ». Ils permettent aux mobiles d'appeler un protocole dans le réseau visité
- Le mobile exécute ensuite des fonctions *distribuées*: collecte, calcul, contrôle.

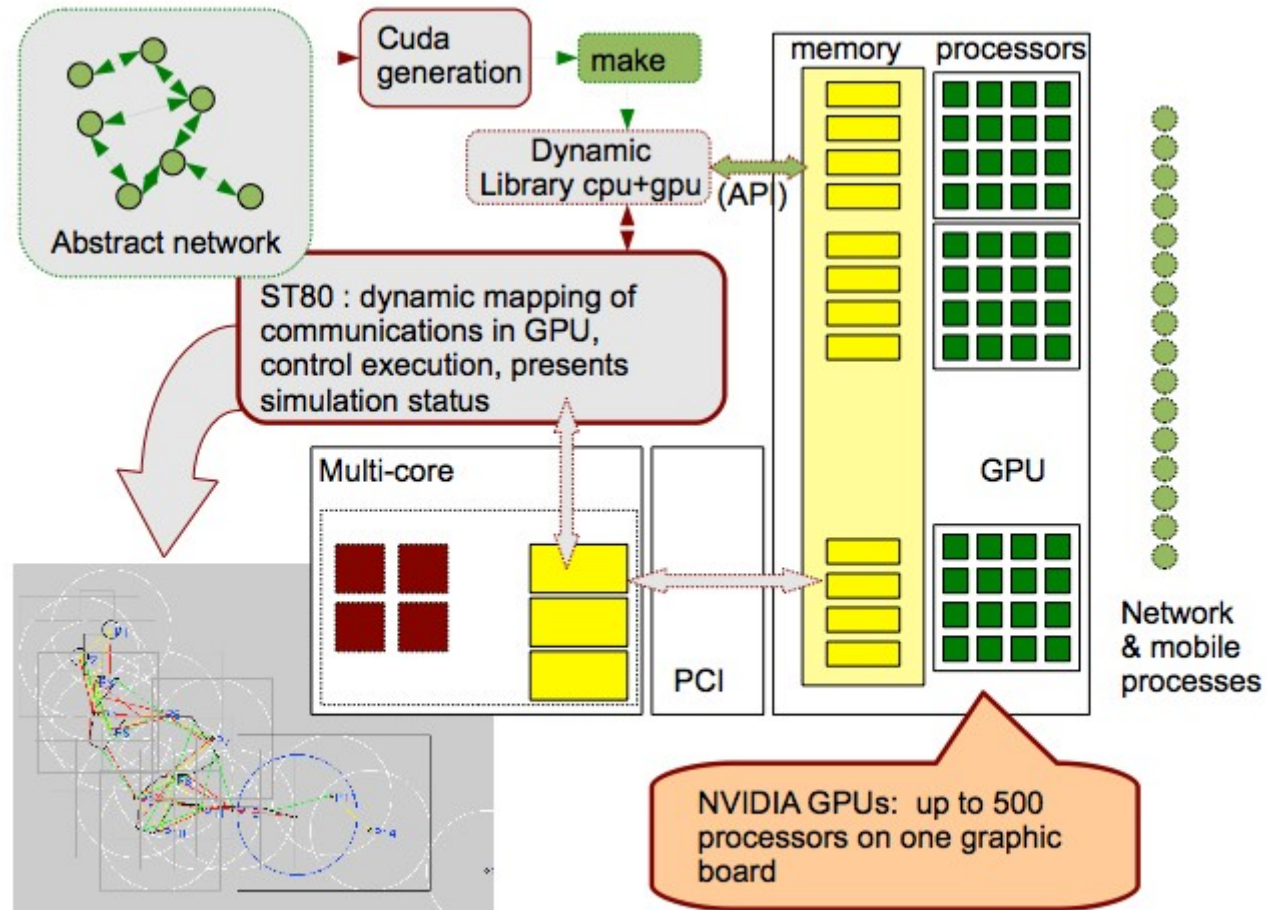


# IV Simulation sur GPU (CUDA)

# Principes

- Capteurs : peu de comportements différents  
*=> parallélisation possible sur GPU*
- Structure planaire de graphe irrégulier, mobiles ?  
*générer un mécanisme d'adressage interprété à l'exécution*
- Réseau dynamique (mobiles, erreurs) ?  
*Permettre à un contrôleur de haut niveau (Smalltalk) d'observer et intervenir sur le mécanisme d'adressage*

# Architecture logicielle

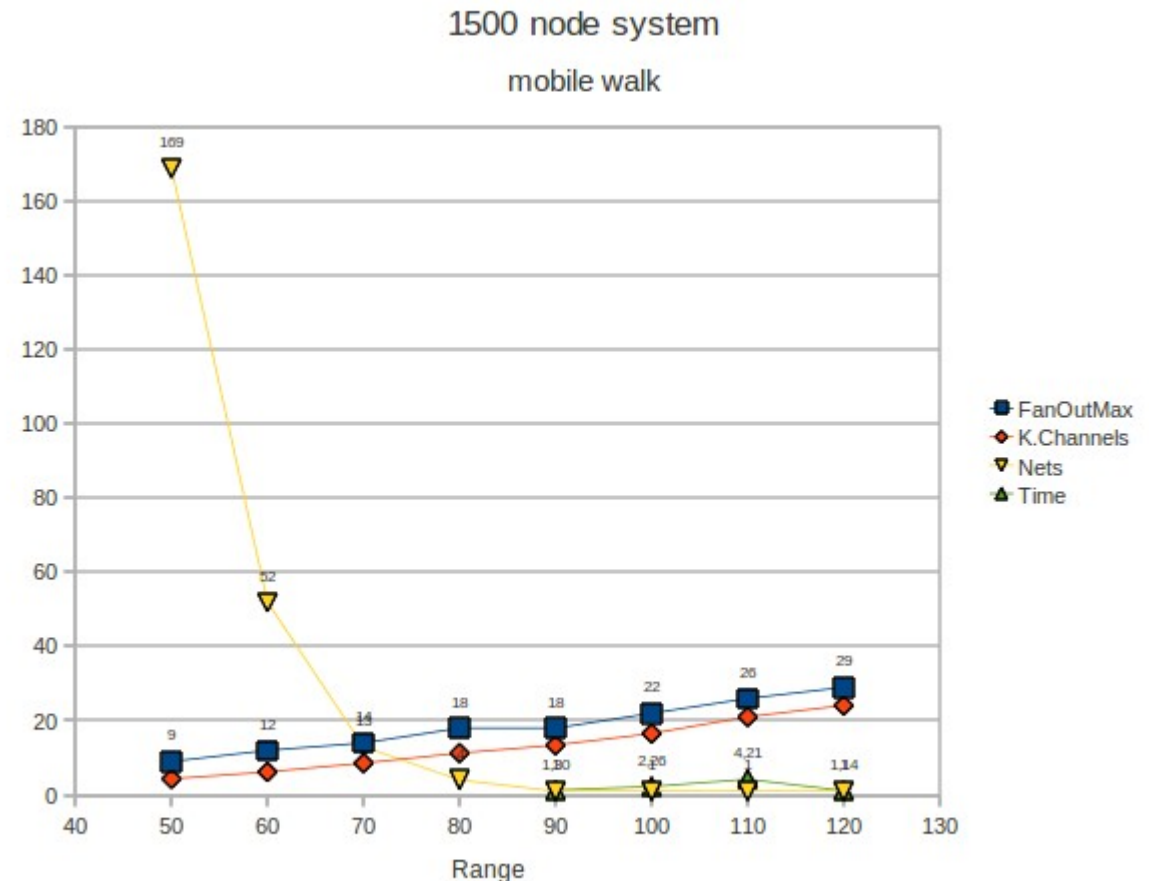


# Relation VM haut niveau - GPU

- **Génération du programme parallèle :**
  - Mécanisme de communication calculée représentant la topologie (structures C désignant index et limites)
  - Fonctions suivant les phases du modèle synchrone :
    - M : écriture « parallèle » dans les buffers de communication,
    - N : lecture « parallèle »,
    - C changements d'états, en parallèle
- **Commande et observation du programme parallèle**
  - API d'accès aux variables GPU
  - Le haut niveau commande, patche, prend les résultats

# 1500 capteurs, parcours calculé

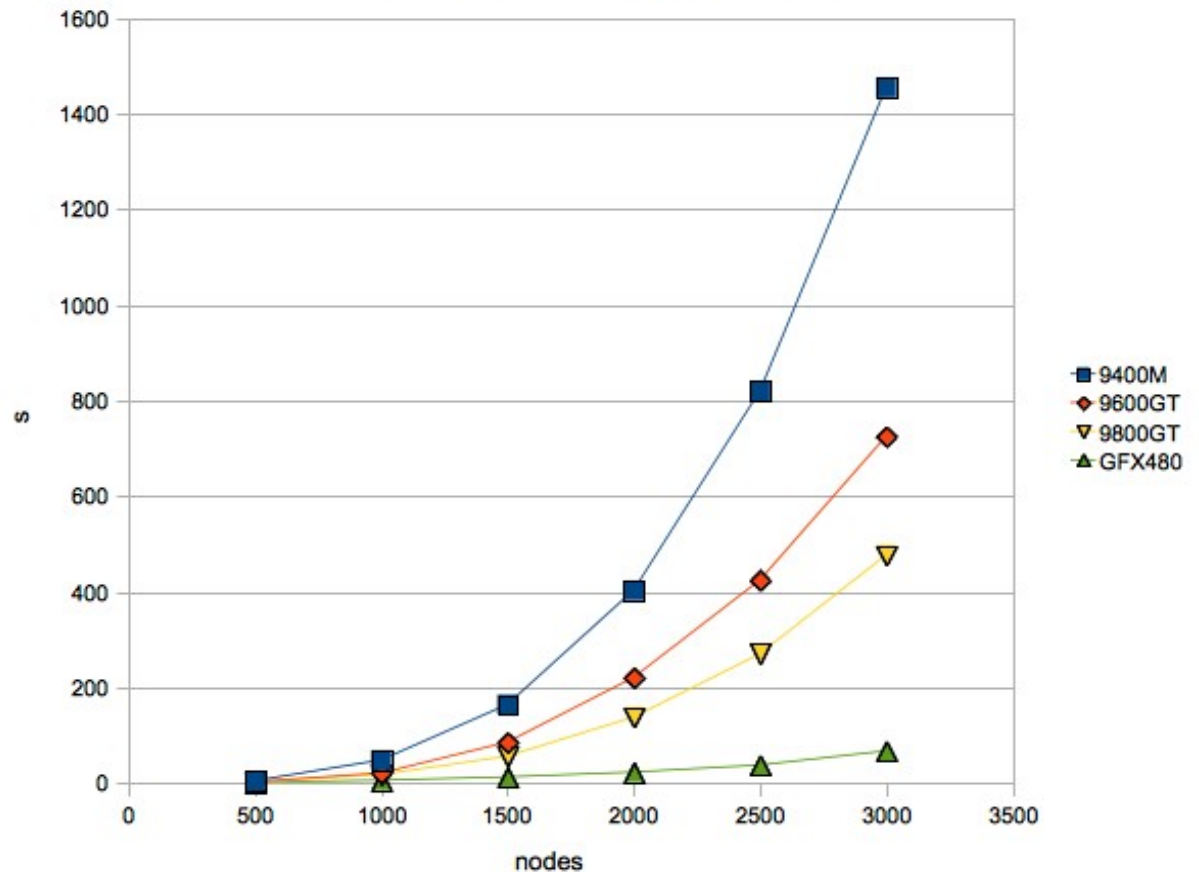
- Génération d'un système de 1500 capteurs
- Portée variant de 50 à 120
- Connectivité croissante
- Nb réseaux décroissant



# 6 réseaux dynamiques – 4 GPUs

- De 500 à 3000 noeuds
- Diametre, leader, modification reseau, diametre, leader
- GFX480, 9800GT  
9600GT, MacMini  
9400M

6 random networks on 4 GPUs



# IV Conclusion



# Statut des travaux

- Simulation
  - MIMD multi-thread Occam, compilation lente
  - SIMD CUDA, compilation rapide
- Outils interactifs génération et contrôle du GPU
- *Ebauche* d'implémentation TDMA sur noeuds TI
- Mobilité :
  - – étude de générateurs pour Occam-Pi,
  - ++ implantation/démonstrable sur CUDA (DLL)

# Statut des travaux

- Applications tests : passagers et transports en commun (Ville de Brest, 2010)
- Optimisation offline avec UBS (2010).
- Plein de projets de projets, plusieurs mobiles, flottes, ...



# Merci

<http://wsn.univ-brest.fr>  
<http://wsn.univ-brest.fr/pottier>

# Animation: déploiement côtier 2 portées différentes

Mobile is moving and processing each encountered network  
The result is a bounding box produced by distributed computations, shown with sensors and path

Mise en mémoire tampon : 100,00% 2:00 /

# Des questions

- Q1 : Sait-on implémenter l'exécution conformément au modèle ?
- Q2 : Quelle crédibilité a la connectivité du modèle physique ?
- Q3 : Le flot est-il assez flexible pour permettre l'intégration des mobiles ?