

# Cohérence des données dans les simulations de Réalité Virtuelle distribuées appliquées à la biologie

Mikaël Bourhis  
Centre Européen de Réalité Virtuelle  
25 rue Claude Chappe  
Plouzané, France  
bourhis@enib.fr

Vincent Rodin  
Université de Bretagne Occidentale  
20 avenue Victor Le Georgeu  
Brest, France  
vincent.rodin@univ-brest.fr

## Résumé

Dans le cadre de simulations de réalité virtuelle appliquées au domaine de la biologie, nous cherchons à mettre à profit un ensemble d'ordinateurs standards constitués en grille. Cette grille d'ordinateurs supporte les simulations distribuées. La méthode proposée pour réaliser la distribution s'appuie sur le modèle générique RéISCOP (modèle développé au sein de notre laboratoire) et sur la réplique de certains de ses constituants. Il s'agit d'une distribution spatiale où les données répliquées sont périodiquement remises en cohérence sans pour cela nécessiter de synchronisation forte. La mise en cohérence s'appuie sur la transmission des variations des états des données entre les nœuds de calcul concernés. Le support logiciel de ces simulations distribuées est DIVA, un agent informatique présent sur chaque nœud et qui confère à la grille son architecture en pair à pair.

## Mots-Clés

réalité virtuelle, distribution, pair à pair, cohérence, biologie

## 1. INTRODUCTION

L'étude des systèmes biologiques ne se limite pas aux expériences pratiquées sur la paillasse. De plus en plus, les biologistes utilisent les outils informatiques. C'est le cas notamment dans le domaine de la génétique où l'informatique est employée pour analyser des séquences de bases azotées et autres éléments constitutifs de l'ADN et des protéines. Plus généralement, l'informatique permet de traiter d'importantes quantités de données générées par les expériences biologiques. Toutes ces fonctions de l'informatique utilisées en biologie forment la définition commune du terme *bio-informatique*. L'informatique est aussi un précieux relais aux mathématiques lorsqu'il s'agit d'analyser, de comprendre et de simuler des réactions bio-chimiques.

Une autre orientation de l'emploi de l'informatique pour l'étude des systèmes biologiques qui est moins courante, peut être inattendue, mais qui possède nombre de qualités, est la Réalité Virtuelle.

Les travaux de recherche menés au Centre Européen de Réalité Virtuelle (CERV) ont pour vocation l'étude des systèmes complexes à l'aide d'outils de Réalité Virtuelle formant ainsi un véritable laboratoire virtuel d'études.

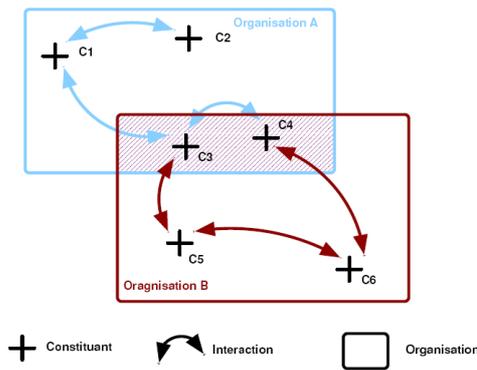
Une partie de ces travaux s'applique à la biologie pour simuler par exemple : la coagulation sanguine, les cellules myélomateuses, le phénomène d'urticaire, la vasorelaxation artérielle [11]. Dans le but de réaliser des simulations de plus grandes tailles, nous choisissons de grouper plusieurs ordinateurs pour former une grille de calcul. Toutefois, l'accès aux données hébergées sur un nœud distant demande des ressources supplémentaires (en plus de la simulation habituelle) qui ne sont pas négligeables lorsqu'on souhaite faire des synchronisations fréquemment.

Cet article détaille la solution que nous mettons en œuvre pour accéder à ces données tout en faussant le moins possible la simulation en cours. Dans une première partie, nous expliquerons le modèle générique RéISCOP sur lequel s'appuie nos simulations de biologie ainsi que la distribution de ces simulations. Dans un deuxième temps, nous rappellerons des généralités sur la distribution des calculs pour ainsi asséoir notre positionnement. Ensuite, nous détaillerons notre mécanisme de mise en cohérence des données réparties, et nous terminerons en présentant deux applications.

## 2. SIMULATIONS DE BIOLOGIE

Les projets du CERV appliqués au domaine de la biologie et des écosystèmes naturels sont l'apanage de l'équipe EBV (Ecosystème et biologie virtuelles). En exploitant les outils de réalité virtuelle, nous ne remplaçons pas seulement l'étude des systèmes biologiques dans le contexte de l'informatique comme le font les simulations *in silico* (résolution mathématique et informatique de problèmes biologiques) mais en plus, nous offrons à l'expérimentateur un environnement virtuel contenant son modèle et la possibilité pour lui d'interagir avec à n'importe quel moment : l'être humain devient partie intégrante de la boucle de simulation du modèle. Cette expérimentation est qualifiée d'expérimentation *in virtuo* [12] faisant ainsi le parallèle avec les expérimentations *in vivo* et *in vitro* qui prennent place respectivement sur le sujet vivant et dans les éprouvettes.

De plus, les recherches de Desmeulles[5] sur la modélisation basée sur la vision systémique de la biologie ont abouti à la réalisation d'une API (Application Programming Interface) nommée *RéISCOP* qui permet de réutiliser facilement des entités autonomes et autres objets caractérisant les expérimentations *in virtuo* (cellule, réaction chimique, phénomène



**Figure 1:** *Couplage structurel dans le modèle ReISCOP : la figure nous montre deux organisations A et B en couplage structurel. L'organisation A a pour structure l'ensemble des constituants {C1; C2; C3; C4} et l'organisation B {C3; C4; C5; C6}. L'ensemble {C3; C4} représente la zone de couplage structurel entre A et B. Les deux systèmes correspondant aux deux organisations s'influenceront mutuellement en modifiant l'état de C3 et C4.*

de diffusion, parseur SBML, ...). Ce modèle générique RéISCOP (Réification des Interactions; Structure; Constituant; Organisation; Phénomène) permet de décrire un système (qui pour nous est de nature biologique) comme la composition de ses sous-systèmes. Chacun de ces sous-systèmes (appelé Organisation) est constitué d'éléments actifs (appelés Interactions) qui agissent sur des éléments passifs (les Constituants). L'ensemble de ces éléments passifs peut être manipulé par des Interactions appartenant à des Organisations différentes. On dit alors que les Organisations sont en couplage structurel (figure 1). Un autre point essentiel de ce modèle est que les Interactions sont de véritables objets informatiques qui réalisent des actions. Elles ne sont pas la résultante d'une action d'un Constituant avec un autre de ses semblables. Les Interactions sont des objets qui possèdent des connaissances (leurs attributs) et des activités (actions réalisées périodiquement). C'est ce qui est appelé la *réification des interactions*. Ces Interactions agissent sur des Constituants précis. Ce sont les Phénomènes qui, lorsque les conditions nécessaires sont détectées, instancient les Interactions voulues entre des Constituants choisis.

De ce modèle générique découlent plusieurs modèles spécialisés dans le domaine de la chimie, de la mécanique et de la biologie.

Avant d'expliquer comment nous proposons de distribuer une simulation basée sur ce modèle générique, évoquons d'abord la distribution des simulations et des calculs de manière générale.

### 3. DISTRIBUTION DE CALCULS ET DE SIMULATIONS

Pour concevoir et réaliser notre logiciel de distribution, nous nous sommes intéressés d'une part à la réalité virtuelle distribuée et d'autre part aux grilles de calcul.

#### 3.1 Généralités

##### 3.1.1 Réalité Virtuelle Distribuée

Il s'agit de l'interconnexion et de la coordination de simulations de Réalité Virtuelle par un réseau. Les utilisateurs ont la liberté d'avoir accès à l'environnement simulé depuis des postes différents. Sur le plan historique, c'est le département américain de la Défense qui débuta les recherches dans le domaine en créant *SIMNET* [2]. Dans ce système, chaque site diffuse la position et la vitesse des entités qu'il simule. Pour limiter ces transferts incessants sur le réseau, les différents sites tentent d'estimer eux-même les positions des entités simulées par les autres : c'est la technique du *dead reckoning*. Cette technique a ensuite été normalisée (*DIS*<sup>1</sup>) et étendue à la notion de *Réel-Fantôme*. Depuis, beaucoup de systèmes ont été développés en utilisant le protocole issu des premiers travaux américains -HLA [4]- ou bien CORBA. On peut citer DIVE[7], VIPER[13]. Nous retiendrons que les concepts fondamentaux sont : la simulation d'objets physiques dans un environnement commun, la transmission et la propagation des événements intervenant sur ces objets. De plus, les systèmes sont multi-utilisateurs, synchronisés, et temps-réel.

##### 3.1.2 Grilles de calcul

Les grilles de calcul peuvent être classées en deux groupes : le *calcul global* et le *metacomputing*. Alors que le *calcul global* consiste à récupérer des cycles de temps de processeurs inutilisés sur des machines de particuliers qui sont connectées à l'Internet, le *metacomputing* quant à lui, utilise des ordinateurs dédiés, souvent hébergés dans des centres de recherche. Le *metacomputing* est basé sur le principe suivant : un client qui possède un problème à résoudre le transmet au ou aux serveurs disponibles et compétents. Plusieurs environnements sont aujourd'hui exploités : DIET[10], Globus[6] par exemple. Un autre type d'architecture, provenant des grilles de données est aussi maintenant employé : le pair à pair. Chaque machine joue un rôle semblable aux autres. Cette grille est par conséquent ouverte et dynamique. Plusieurs équipes travaillent à construire des API ou des intergiciels fournissant les services de base à la construction d'un réseau P2P ([3]). Cependant des difficultés inhérentes aux grilles de calcul restent présentes (répartitions des charges, ordonnancement des tâches, synchronisation) et devront être résolues de manière équitable par la communauté des pairs.

### 3.2 Notre positionnement

Comme nous souhaitons distribuer les calculs de simulations de réalité virtuelle, le schéma de fonctionnement des grilles de calcul nous intéresse beaucoup. De plus, l'architecture en pair à pair est séduisante car cela confère à chaque nœud de la grille une certaine autonomie, indépendance et responsabilité. Dans l'article [1], nous avons expliqué plus en détail ce choix d'une architecture en pair à pair et les justifications qui nous ont permis d'étendre notre grille de calcul à un système multi-agents. La simulation distribuée sur la grille de PC est supervisée par le logiciel DIVA. Ce sont des fonctionnalités communes telles que la répartition des calculs, la synchronisation, le maintien de la cohérence temporelle et spatiale, qui font que les nœuds de la grille ont tous des rôles semblables et sont des pairs. Ces pairs sont actifs et proactifs grâce à des activités exécutées régulièrement et à des décisions prises de leur propre initiative dans

<sup>1</sup>Distributed Interactive Simulation

le but de coopérer pour exécuter du mieux possible la simulation *in virtuo* : ce sont bien des agents. Cela justifie le nom que nous avons donné à ces logiciels : "*agents in virtuo distribués*" (Distributed In Virtuo Agents). De plus, ce sont bien des simulations de réalité virtuelle dont il s'agit et ce qui nous préoccupe est la résolution de problèmes comme la synchronisation et le partage des données. C'est pourquoi, la partie suivante traitera de la mise en cohérence des données de l'univers de synthèse réparti sur plusieurs nœuds.

## 4. DISTRIBUTION DE REISCOP PAR DIVA

### 4.1 Modèle de la mise en cohérence

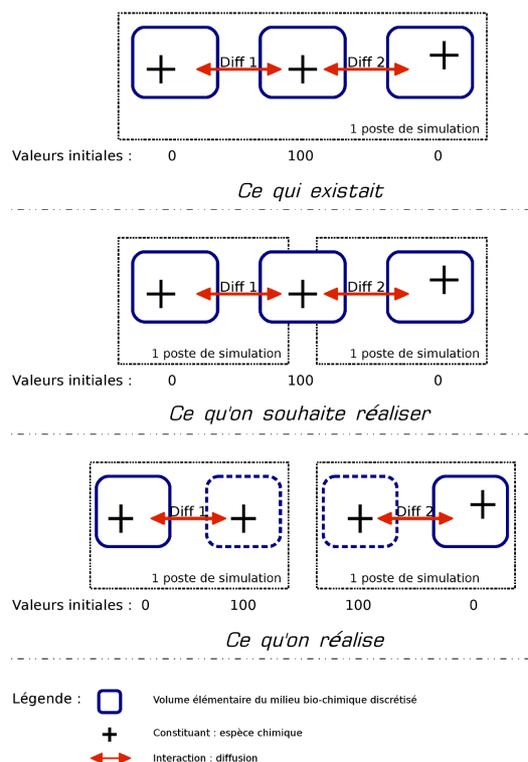
Dans notre modèle, les agents ou entités autonomes ne sont pas dupliqués. Seules des parties d'environnement peuvent avoir plusieurs images. Il est fondamental que ces images d'une même zone de l'environnement ne provoquent pas de biais conséquents dans la simulation. Il faut que ces images restent cohérentes. Comme l'écrit Simon [9], la cohérence dans un monde virtuel, c'est lorsqu'une scène est identiquement perçue par toutes les entités qui l'observent. Il y a donc décohérence quand une modification apportée à l'une des images n'est pas répercutées sur les autres. Ces répercutions, ou ces mises à jour des images ne sont rien d'autre qu'une resynchronisation des images.

La distribution spatiale est peut-être l'idée la plus naturelle, celle qui nous vient d'abord à l'esprit lorsqu'on est face à un univers virtuel. En effet, une simulation qui occupe un volume cubique sera divisée en une collection de sous-cubes et la simulation d'un environnement longiligne (comme toutes sortes de vaisseaux ou tubes que l'on peut rencontrer en biologie) sera divisée en tranches.

C'est l'approche qui est choisie dans les simulations de réalités virtuelles distribuées telle que HLA (High Level Architecture). C'est cette approche qui est aussi retrouvée dans les jeux vidéos distribués. De manière générale, cette approche est appelée *décomposition basée sur les données*. Et dans le cadre des jeux vidéos comme de la réalité virtuelle, les données considérées sont les objets physiques et ceux-ci sont localisés dans l'espace. Ce découpage induit une consommation importantes de ressources informatiques à cause des synchronisations.

En effet, lorsque au moins deux entités interagissent ensemble alors qu'elles sont simulées par des ordinateurs différents n'ayant aucune mémoire en commun, il faut répercuter aussi fidèlement que possible les agissements de chacune d'elles sur l'autre ordinateur. Cela peut très vite engendrer une quantité de communications très importantes entre ces deux postes de simulation.

Dans le cas des Expérimentations In Virtuo, nous choisissons de mettre en œuvre une synchronisation faible. Contrairement à la synchronisation forte dans laquelle les données portant sur les entités de la simulation et leurs états sont transmises et actualisées à chaque changement et à chaque modification, dans la synchronisation faible ces données ne sont transmises que périodiquement et de temps en temps. Précisons davantage la chose : dans le cadre de nos simulation, où un ordonnanceur passe la main à tour de rôle à chacune des activités programmées pour s'exécuter à un instant donné, une synchronisation sera forte lorsqu'elle sera exécutée à chaque fois, c'est à dire à chaque pas de l'ordonnanceur. Ainsi, les entités de la simulation n'auront changé d'état qu'une seule fois entre 2 actions de synchronisation. Avec une synchronisation faible, on relâche cette contrainte,



**Figure 2:** Exemple de la distribution d'un milieu bio-chimique sur 2 postes de simulation : En haut : simulation non distribuée. Au milieu : avec plusieurs postes de simulation, les interactions sont déplacées mais pas dupliquées. La question du volume élémentaire partagé entre les 2 postes est ouverte. En bas : l'espèce chimique (et plus généralement les Constituants) du volume partagé est dupliquée : 2 images, avec les mêmes valeurs initiales.

et on provoque une action de synchronisation à une fréquence plus faible que celle de l'ordonnanceur. Ce procédé permet de réduire les échanges véhiculés via le réseau entre deux postes de simulation mais n'est pas sans risque : on sait pertinemment que sur l'un des postes on a une vision erronée des entités simulées par les autres postes distants. Dans notre cas, les entités de nos simulations sont les Constituants du modèle RéISCOP. Ce sont des objets passifs qui changent d'état en subissant les activités des Interactions. Comme nous l'avons évoqué plus haut, une Interaction n'est présente que sur un poste de simulation à la fois, tandis que les Constituants peuvent avoir une image sur plusieurs postes. Il s'agit donc de synchroniser *de temps en temps* ces images d'un même Constituant. Prenons l'exemple de deux Interactions de diffusions agissant sur une espèce chimique (c'est le Constituant) dans un milieu de dimension  $1 * 1 * 3$ . A l'instant  $t = 0$  seconde, la concentration de l'espèce chimique de la zone centrale est initialisée à 100. L'activité des Interactions de diffusion est de rééquilibrer les concentrations des espèces chimiques à chaque pas de l'ordonnanceur. La figure 2 récapitule notre démarche de distribution pour cet exemple. La figure 3 montre le déroulement de la simulation distribuée pour trois cas de figures de synchronisation. En haut, c'est le cas de la synchronisation forte avec propa-

gation de l'état des Constituants. Au centre, la synchronisation est relâchée : la synchronisation des Constituants (par la transmission de leur état) lorsque l'action de synchronisation n'est pas rigoureusement calée sur l'ordonnanceur des Interactions est complètement faussée. Si l'état des Constituants n'est pas retransmis à toutes ses images à chaque fois qu'une modification ou perturbation a lieu, la synchronisation ne peut aboutir à des résultats corrects.

Pour dépasser ce problème, nous avons construit un mécanisme appelé *mise en cohérence des images* qui permet de fausser à minima la simulation même lors d'une synchronisation faible. Ce mécanisme repose non pas sur la transmission de l'état du Constituant mais sur la variation de son état. Cet algorithme nécessite une mémoire supplémentaire sauvegardant l'état que le Constituant avait à la dernière synchronisation faible. Voici le détail de cet algorithme :

#### Algorithm 1: Mise en cohérence

```

begin
  var ← etat_courant - etat_derniereSynchro
  sommeVar ← 0
  for i ∈ Images do
    sommeVar ← sommeVar + var_i
  etat_synchro ← etat_derniereSynchro + sommeVar
  etat_derniereSynchro ← etat_synchro
end

```

Reprenons le même exemple que précédemment et appliquons lui cet algorithme : figure 3, schéma du bas. Non seulement, les deux images de l'espèce chimique sont cohérentes mais le phénomène de diffusion n'est pas faussé.

Avec ce mécanisme nous sommes en mesure de simuler un milieu bio-chimique réparti sur plusieurs postes de simulation. En effet, après avoir découpé le milieu en tranches ou en cubes, et réparti les Phénomènes et les Constituants de chaque volume élémentaire du milieu sur les différents postes, il n'y a que les images des Constituants à superviser (et RéISCOP se charge de la simulation habituelle). Les réactions chimiques sont pour leur part confinées dans les volumes élémentaires du milieu.

## 4.2 Implémentation

Pour réaliser une mise en cohérence d'un Constituant, il est nécessaire que les DIVA qui partagent ces images s'arrêtent un moment, le temps de la mise en cohérence, et échangent les données utiles (comme décrit dans l'algorithme 1). Pour faire en sorte que les DIVA s'arrêtent ensemble et fassent ces échanges en même temps (sinon cela ne peut fonctionner) sans qu'aucun organe central et superviseur leur intime l'ordre, les DIVA implémentent chacun un mécanisme de Rendez-Vous. Une fois les DIVA au rendez-vous, il exécute l'algorithme de mise en cohérence. Dans la version actuelle, les DIVA se partagent et se transmettent un jeton. Lorsqu'ils sont en possession du jeton, ils organisent le Rendez-Vous. Une fois à la fin du Rendez-Vous, ce qui inclut la mise en cohérence, le DIVA passe le jeton à un de ses proches. Cela s'entend car dans le cas de la distribution spatiale, les DIVA se partagent l'ensemble du milieu par tranches ou plus globalement par des volumes parallélépipédiques. Les DIVA discrétisent ainsi le milieu à leur manière : l'espace et la portion du milieu bio-chimique qui

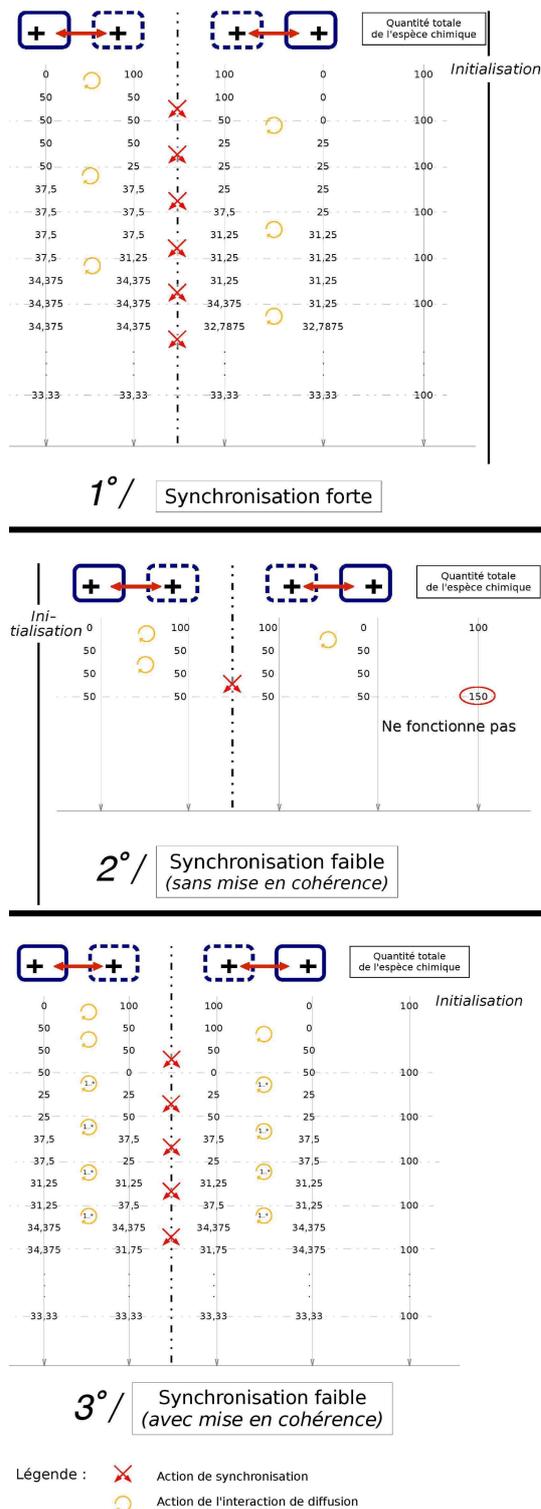


Figure 3: Exemple de la synchronisation entre 2 postes de simulation dans le cadre de la distribution d'une espèce chimique et de deux interactions de diffusion. 1/ Synchronisation forte : à chaque activité des interactions, il y a transmission de la concentration de l'espèce chimique (état du Constituant). 2/ Synchronisation faible : la transmission de la concentration se fait indépendamment des activités des interactions. 3/ Synchronisation faible avec l'algorithme de mise en cohérence.

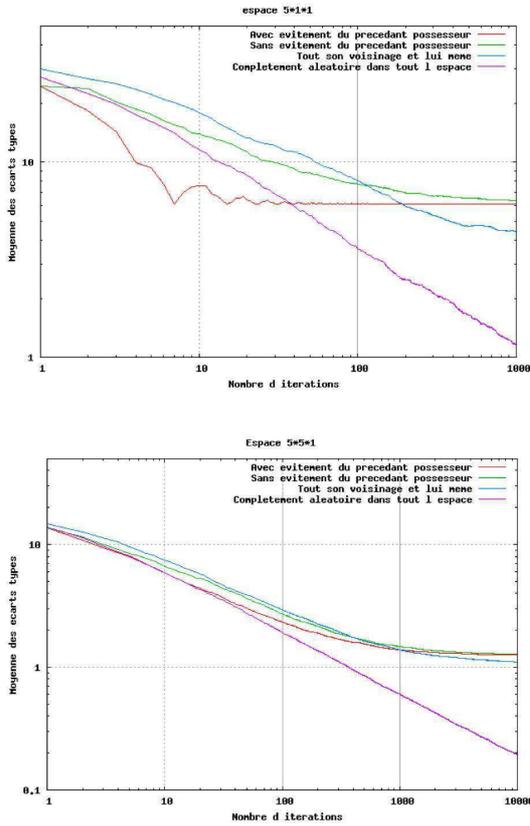


Figure 4: Expériences du parcours d'un jeton sur deux grilles de dimensions respectives  $5*1*1$  et  $5*5*1$ .

leur est dédié forme l'une des mailles d'une grille. Le jeton se transmet donc de voisin en voisin sur cette grille. Le choix du voisin peut être complètement aléatoire ou bien être un peu réfléchi, en évitant par exemple le DIVA qui venait de lui transmettre le jeton précédemment.

Cet algorithme de transmission du jeton a été construit et testé dans un programme informatique indépendant et dédié à cette tâche. Nous cherchions à connaître la proportion d'occupation du jeton dans chacune des mailles de la grille. Le jeton passe bien par toutes les mailles mais parfois plus souvent par certaines que par d'autres. Les résultats présentés dans la figure 4 montre en quelque sorte l'homogénéité de la répartition du jeton sur l'ensemble de cette grille grâce au calcul des écarts-types : lorsque l'écart-type est égal à zéro, toutes les mailles (et dans le contexte des DIVA, cela signifie *tous les DIVA*) ont possédée le même nombre de fois le jeton. Ces résultats sont obtenus en moyennant à chaque fois 100 expériences similaires. De manière générale, cet algorithme du passage de jeton est viable car tous les DIVA sont quasiment tous autant en possession du jeton que les autres et de ce fait sont les organisateurs des Rendez-Vous, amenant à la mise en cohérence (seuls les DIVA qui sont aux extrémités ont un peu moins souvent le jeton).

## 5. APPLICATIONS

Dans ce chapitre, nous souhaitons en premier lieu montrer les résultats d'une simulation très simple qui, sur le plan

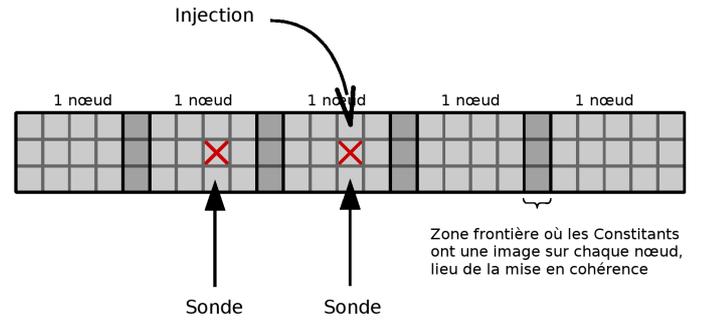


Figure 5: Schéma de l'expérience de diffusion : un milieu de dimension  $25*3*3$  et deux sondes. Injection de la substance chimique au centre.

biologique n'a pas grand intérêt, mais qui met à l'œuvre le principe de mise en cohérence des images des Constituants. Il s'agit de la diffusion d'une espèce chimique dans un milieu. Ensuite, nous montrerons une application de taille et de complexité plus importantes : il s'agit de la vasorelaxation d'une artériole.

### 5.1 Expérience de diffusion

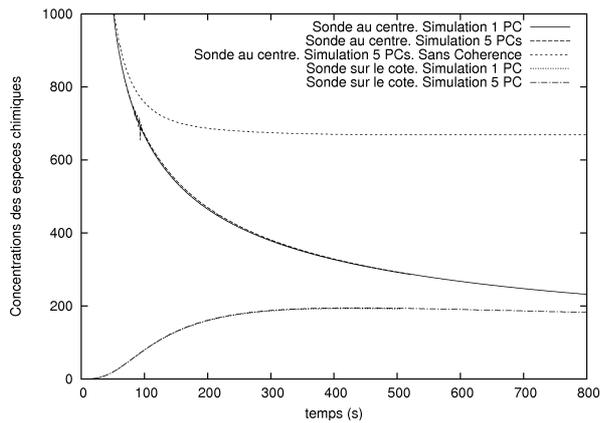
Voyons une simulation qui n'exploite que l'interaction de diffusion dans un milieu bio-chimique. Dans un milieu discrétisé de dimension  $25 * 3 * 3$ , nous injectons à l'instant  $t$  égal 0 seconde une quantité de  $36.0 * 10^3$  unités (l'unité a peu d'importance ici car on ne se préoccupe pas de la signification biologique de la simulation. En général, l'unité est la mole) d'une espèce chimique au centre du milieu simulé. Nous laissons les interactions de diffusion (qui agissent entre chaque maille du milieu discrétisé) exécuter leur activité. Pour obtenir les résultats suivants, nous avons placé deux sondes virtuelles dans cet environnement de synthèse : l'une au centre du milieu et l'autre excentrée, à 32% de la longueur (voir figure 5).

La simulation a été réalisée 3 fois. La première fois avec un seul nœud de calcul, la seconde avec cinq nœuds de calcul et la dernière fois, avec cinq nœuds de calcul mais sans la réalisation de la mise en cohérence.

Les résultats du graphique de la figure 6 montre que les courbes sont superposées dans le premier et deuxième cas de figure. En revanche, lorsque les DIVA n'organisent pas de *Rendez-Vous* dans le but de réaliser les mises en cohérence, la quantité d'espèce chimique injectée ne peut se diffuser que dans le milieu simulé par le nœud de calcul placé au centre. Au lieu que sa quantité s'équilibre à 160.0 (valeur asymptotique théorique à  $t = +\infty$ ), elle tend vers 666.7. La courbe correspondant à la deuxième sonde (sur le côté) n'est pas représentée dans ce cas de figure car la valeur mesurée est toujours égale à zéro : l'espèce chimique ne peut se diffuser jusqu'à cet endroit.

### 5.2 Endothélium

Dans cette simulation de réalité virtuelle appliquée à la biologie, nous travaillons en collaboration avec des chercheurs de l'Unité de Physiologie Comparée et Intégrative de l'Université de Bretagne Occidentale qui nous fournissent les données biologiques. Cette simulation correspond à une ex-

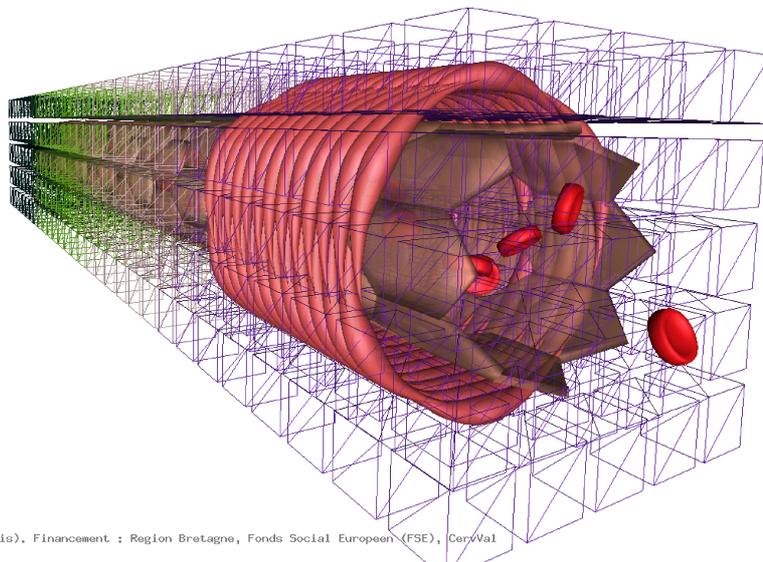


**Figure 6:** Ensemble de courbes représentant les résultats mesurés au cours des trois cas de figure de l'expérience de diffusion. De haut en bas : 1/ mesures de la sonde centrale quand la mise en cohérence est désactivée, 2/ mesures de la sonde centrale pour la simulation non distribuée superposées aux mesures de la sonde centrale pour la simulation distribuée sur 5 nœuds, 3/ mesures de la sonde excentrée pour les simulations distribuée et non distribuée (superposition des résultats)

périence *in vitro* dans laquelle, les biologistes injectent dans le milieu sanguin une substance (acétylcholine) qui sera captée par les cellules endothéliales (ensemble de cellules formant la première paroi des vaisseaux sanguins, directement au contact du sang) et qui provoquera chez elles toute une cascade de réactions bio-chimiques. Au final, ces cellules relarguent une substance (NO) qui va déclencher la relaxation des cellules musculaires voisines (formant une paroi cellulaire autour des vaisseaux sanguins) [8].

Dans cette simulation, trois types de cellules sont implémentés, en plus du milieu sanguin. De ces trois cellules, les globules rouges n'ont qu'une activité de déplacement et ne comportent aucuns mécanismes internes. En revanche, les cellules musculaires captent la substance NO et se dilatent en conséquence. Quant aux cellules endothéliales, ce sont les cellules les plus riches du point de vue comportemental : une quinzaine de réactions chimiques et faisant intervenir autant de composés chimiques sont à l'œuvre au cœur de chacune d'entre elle. La distribution de cet application se fait actuellement sur 5 nœuds de calcul. Ainsi on peut voir sur la figure 7 des ensembles de volumes de couleurs différentes (violet, blanchâtre, vert et noir). Cela permet de se représenter l'espace simulé par chacun des nœuds. Dans le tableau récapitulatif en haut de l'image, on constate que l'ensemble des cellules endothéliales sont à peu près équitablement réparties sur les nœuds de calcul. Malgré des résultats quantitatifs non conformes aux expériences *in vitro* (nos paramètres ne sont pas entièrement bien calibrés), les résultats qualitatifs sont à la hauteur de nos attentes : les cellules musculaires se dilatent bien en fonction de la quantité de NO produite par les cellules endothéliales.

Host	XCPU	Cells	Comp	Diff	React	Net IN	Net OUT	F visu	F RdV
demo	60	0	770	0	12			0,5	0,0
bourhis	13	32	393	495	1000			0,0	0,0
rodin	21	32	393	455	950			0,0	1,5
beryllium	18	40	466	520	1156			0,0	4,0
beryllium	20	24	345	455	798			0,0	3,5
5 DIWA	492256	128	2367	1925	3906				2,2



Projet: DIWA (M. Bourhis). Financement : Région Bretagne, Fonds Social Européen (FSE), CerVal

**Figure 7:** Copie d'écran de la simulation distribuée de l'endothélium.

## 6. CONCLUSIONS

Dans le cadre des simulations distribuées sur une grille, la cohérence des entités et des données répliquées (avec plus ou moins de fidélité) est un des principaux enjeux car il faut trouver un compromis entre une cohérence la plus rigoureuse possible et les ressources (CPU, réseau, mémoire) nécessaires à sa mise en œuvre. En effet, la cohérence entre deux données distantes nécessite un échange par réseau. Et lorsqu'il est question de minimiser ces échanges, on entreprend de coder un algorithme de prédiction qui lui, nécessitera des ressources calculatoires.

C'est dans ce contexte que nous proposons notre mise en œuvre de la cohérence des données dans les simulations de réalité virtuelle distribuée. Dans un premier temps, nous avons décrit le modèle générique sur lequel nous nous appuyons pour construire nos simulations de biologie. C'est aussi sur ce modèle que s'appuient nos techniques de distribution et notamment la distribution spatiale. Pour cela, nous ne répliquons pas les éléments actifs de la simulation (les Interactions) mais nous répliquons les éléments passifs (les Constituants). Ces éléments ont donc plusieurs images qui sont modifiées les unes indépendamment des autres. Pour que ces images gardent un sens du point de vue de la simulation globale, il est nécessaire de les "synchroniser" ou de rétablir leur cohérence.

Nous avons montré que pour une synchronisation forte c'est à dire à chaque cycle de l'ordonnanceur qui active les Interactions, la seule transmission des états des données répliquées étaient suffisante mais conduisait à l'échec dans le cas d'une synchronisation plus lâche ou plus faible. Cette synchronisation faible, nous l'avons choisie car elle laisse plus de liberté aux nœuds de calcul (qui sont pour nous autonomes et qui prennent la forme et la fonction d'agents informatique : DIVA) et réduit la charge du réseau. Pour cela, notre algorithme de mise en cohérence des images est basé sur la transmission des variations des états des images. Pendant cette transmission, pour ne pas fausser la mise en cohérence, les simulations locales concernées doivent être figées. Pour réaliser ceci, nous avons mis en place un mécanisme de rendez-vous basé sur la possession d'un jeton que les DIVA se transmettent de proche en proche. Pour valider notre approche, nous avons ensuite présenté une simulation où n'intervient, pour l'essentiel, que ces mécanismes. Nous obtenons des résultats identiques (courbes se superposant) pour la simulation non distribuée et la simulation distribuée sur une grappe de 5 nœuds de calcul. Et pour terminer, nous avons brièvement présenté la simulation de la relaxation artérielle pour laquelle la crédibilité biologique a toute son importance car elle est le fruit d'une collaboration interdisciplinaire entre informaticiens et biologistes. Cette simulation valide à nouveau notre approche. Toutefois, pour améliorer encore la cohérence des images, nous allons prochainement y inclure un algorithme de dead-reckoning qui aura pour rôle d'estimer les variations d'états des images distantes pour les ajouter aux variations locales, entre deux étapes de synchronisation.

## 7. REMERCIEMENTS

Ce projet bénéficie du soutien financier de la "région Bretagne" et du Fonds Social Européen via une bourse de doctorat.

## 8. REFERENCES

- [1] M. Bourhis and V. Rodin. A multiagent view of grid computing for biological simulations in virtual reality. In *The 21st annual European Simulation and Modelling Conference (ESM'07)*, pages 447–451, St Julian, Malte, 22–24 October 2007.
- [2] J. Calvin, A. Dickens, B. Gaines, P. Metzger, D. Miller, and D. Owen. The simnet virtual world architecture. In *Virtual Reality Annual International Symposium*.
- [3] F. Cappello, S. Djilali, G. Fedak, T. Herault, F. Magniette, V. Néri, and O. Lodygensky. Computing on large-scale distributed systems: Xtrem web architecture, programming models, security, tests and convergence with grid. *Future Gener. Comput. Syst.*, 21(3):417–437, 2005.
- [4] Department of Defense. High Level Architecture Interface Specification – Version 1.3. *IEEE P1516.1, Standard for Modeling and Simulation (M&S)*, Avril 1998.
- [5] G. Desmeulles. *Réification des interactions pour l'expérience in virtuo de systèmes biologiques multi-modèles*. PhD thesis, Thèse de doctorat en Informatique de l'Université de Bretagne Occidentale, Brest (France), Décembre 2006.
- [6] I. Foster and C. Kesselman. Globus: A metacomputing infrastructure toolkit. *The International Journal of Supercomputer Applications and High Performance Computing*, 11(2):115–128, Summer 1997.
- [7] E. Frécon and M. Stenius. Dive: a scaleable network architecture for distributed virtual environments. *Distributed Systems Engineering Journal (special issue on Distributed Virtual Environments)*, 5(3):91–100, 1998.
- [8] E. Heylen. *Effet de l'exercice physique sur différents mécanismes médiant la relaxation artérielle endothélium-dépendante : Approche in vivo et in vitro chez l'homme et l'animal*. PhD thesis, Thèse de doctorat en Sciences et Techniques des Activités Physiques et Sportives de l'Université de Bretagne Occidentale, Brest (France), 25 octobre 2005.
- [9] J. Keller and G. Simon. Toward a peer-to-peer shared virtual reality. In *ICDCSW '02: Proceedings of the 22nd International Conference on Distributed Computing Systems*, pages 695–700, Washington, DC, USA, 2002. IEEE Computer Society.
- [10] B. Lombard. *Spécification et mise en œuvre d'une plateforme de metacomputing multi-agent*. PhD thesis, Thèse de Docteur en Informatique de l'UFR des Sciences et Techniques de l'Université de Franche-Comté, 22 Décembre 2002.
- [11] V. Rodin. *Contribution à l'utilisation de l'informatique en biologie*. Habilitation à diriger des recherches, spécialité informatique, Université de Rennes I, Rennes (France), 6 décembre 2004.
- [12] J. Tisseau. *Réalité Virtuelle – autonomie in virtuo –*. Habilitation à diriger des recherches, spécialité informatique, Université de Rennes I, Rennes (France), 6 décembre 2001.
- [13] P. Torquet. *VIPER : Un modèle de calcul réparti pour la gestion d'environnements virtuel*. PhD thesis, Thèse de Doctorat en Informatique, Université Paul Sabatier, Toulouse (France), 1998.