Proposition d'action de recheche coopérative INRIA

Algorithmes itératifs asynchrones et protocoles de communication à tolérance de perte ajustable

IRISA, LaBRI, LIFC

UR: INRIA Rennes, INRIA Futurs

Mots clés

Algorithmes itératifs asynchrones, grille de calcul, réseaux longue distance, protocoles réseau à haute performance, tolérance de perte ajustable, applications.

1 Descriptions des activités scientifiques

1.1 Contexte et motivation

Les grilles de calcul offrent aux applications une puissance de calcul jamais atteinte auparavant. C'est donc un environnement a priori très séduisant pour la résolution d'applications scientifiques. Aujourd'hui, la programmation des grilles s'appuie majoritairement sur des bibliothèques de communication offrant des communications fiables, et ce quel que soit le paradigme adopté (RPC, passage de message, MVP, etc.). Ceci est vrai pour les modèles issus des recherches en parallélisme (MPI, PVM, PM2, etc.) mais aussi pour ceux issus des recherches en systèmes distribués (CORBA, services Web, etc.)

Pourtant, la réalisation de communications fiables introduit très souvent un surcoût qui augmente en fonction de la latence du réseau physique sous-jacent. Ainsi, sur des réseaux longue distance tels que ceux constituant les épines dorsales des grilles de calcul, la perte de données peut avoir un impact très néfaste sur les performances générales des communications, y compris sur le débit. Le comportement du protocole TCP/IP est très représentatif dans ce domaine, et il est clair que l'utilisation d'un protocole de communication tolérant un certain taux de perte de données permettrait d'augmenter significativement l'efficacité des communications sur des réseaux à grande échelle, à condition que l'application soit compatible avec ce type de fonctionnement bien entendu. En particulier, les données transmises par une application distribuée n'ont pas toute la même valeur sémantique. Si la perte de certaines données peut être acceptée, la perte de données de contrôle l'est beaucoup moins par exemple.

Le premier objectif de cette proposition d'action de recherche vise à définir et à implémenter une interface de programmation de niveau utilisateur acceptant une tolérance de perte ajustable: il s'agit d'offrir des primitives contrôlant quelles parties de données peuvent être perdues dans un message et avec quelle pourcentage. L'implémentation s'appuiera sur un protocole à perte ajustable de données.

La définition d'une interface de programmation, aussi efficace soit elle, est de peu d'intérêt si elle n'est

pas utilisée et validée par des applications. Dans le cadre des applications scientifiques où nous nous situons, les algorithmes itératifs sont souvent recommandés, ils sont même parfois incontournables pour la résolution d'applications scientifiques comportant un très grand nombre de vecteurs inconnus, et éventuellement de fortes non-linéarités. Parmi les avantages des algorithmes itératifs on peut citer leur souplesse de programmation, de parallélisation et leur relative stabilité vis-à-vis des erreurs d'arrondi.

Avec le développement des grilles de calcul, les méthodes itératives sont désormais implantées sur des sites distants, les processeurs échangeant leurs données via des réseaux non dédiés et soumis à des problèmes imprévisibles de fiabilité. Comme évoqué précédemment, les protocoles traditionnels permettent de fiabiliser ces liens, mais la perte importante d'efficacité d'utilisation de ces réseaux par ces protocoles nous amène à nous intéresser à l'utilisation de protocoles de communications non fiables. Il en découle de probables pertes partielles ou totales des messages transitant dans les réseaux.

Les méthodes itératives classiques implantées dans de telles grilles se trouvent à leur tour non fiables, puisque la perte d'une partie d'un vecteur itéré entraîne une modification du comportement de la suite d'itérés censée converger vers la solution du problème considéré. Ce scénario est valable lorsqu'il s'agit d'algorithmes itératifs synchrones à communications synchrones ou asynchrones.

Bertsekas et Tsitsiklis d'une part et El Tarazi d'autre part ont développé en 1981 un modèle d'algorithmes itératifs asynchrones, ceci dans le but de paralléliser un programme sur un multiprocesseurs de la façon la plus flexible possible. Ces algorithmes itératifs introduisent deux notions fondamentales :

- la notion de déphasage entre les itérés qui permet à chaque processeur de calculer à son propre rythme;
- la notion de stratégie qui précise, à un instant t, quelles composantes ou blocs sont effectivement itérés.

Le second objectif de la présente proposition est de généraliser ce modèle au cadre récent des grilles de calcul. Ce modèle dont les itérations ainsi que les communications seront asynchrones, devrait permettre grâce à un environnement de programmation et un protocole de communications adéquat de développer des algorithmes performants et tolérants aux pertes de messages dans les réseaux.

En fait, on peut distinguer trois grands types d'algorithmes itératifs parallèles :

- **Itérations Synchrones - Communications Synchrones** (ISCS) : tous les processeurs commencent la même itération en même temps car les échanges de données sont faits de manière synchrone à la fin de chaque itération.
- **Itérations Synchrones - Communications Asynchrones** (ISCA) : tous les processeurs commencent la même itération en même temps mais les données sont envoyées de manière asynchrone sur les processeurs qui en ont besoin, dès qu'elles ont été traitées.
- **Itérations Asynchrones - Communications Asynchrones** (IACA) : tous les processeurs font leurs itérations sans se préoccuper de l'avancement des autres. Ils n'attendent pas de recevoir les données dont ils ont besoin, mais continuent le calcul avec les valeurs disponibles à ce moment là. Enfin, ils envoient leurs données nécessaires aux autres processeurs de manière asynchrone.

Dans le cadre de ce projet, nous nous intéresserons plus particulièrement aux algorithmes de type IACA. Ces algorithmes sont applicables à une large classe de problèmes d'évolution creux, linéaires ou non. Pour un exemple d'application, voir la section 1.4.5.

1.2 Participants

Les grilles de calcul constituent un nouveau type d'infrastructure qui commence à peine à être appréhendé. De part leurs spécificités, ces grilles remettent en cause les modèles classiques de programmation ainsi que les fonctionnalités offertes par les supports exécutifs. Par conséquent, elles remettent également en cause la conception des applications dans ce contexte.

En effet, les communications fiables à grande échelle exhibent des performances bien trop faibles pour satisfaire les besoins des applications réellement "communicantes". En l'occurrence, le comportement du protocole de transport fiable TCP/IP s'avère très pénalisant dès lors qu'un grand nombre de trame doivent être réémises. Le problème est qu'il est difficile d'envisager le développement d'une application basée sur des

protocoles de communications non fiables car il n'existe quasiment aucun support à part l'utilisation directe du protocole UDP. Cependant, le modèle de programmation offert par UDP est de beaucoup trop bas niveau pour les applications. De plus, le protocole ne permet pas de contrôler la perte des données.

Qui plus est, l'intérêt des communications non fiables ne se restreint pas au cadre strict des réseaux à grande échelle. Il s'étend également au domaine des réseaux SAN à très haute performance (Myrinet, SCI, Infiniband) où le coût de la fiabilisation des protocoles de bas niveau n'est pas négligeable (cf travaux sur BIP/Myrinet, ou sur GAMMA/Ethernet). Dans ce dernier cas, UDP se révèle bien sûr inadéquat, et il est clair que des travaux généraux sur la réalisation de *communications génériques* non fiables sont nécessaires.

Plus précisément, notre motivation est d'étudier comment maîtriser le caractère non fiable des réseaux aussi bien au niveau applicatif qu'au niveau exécutif. Vu la remise en cause des modèles de programmation, induite par les hypothèses précédentes, la présente proposition regroupe des équipes complémentaires pour aborder la problématique des pertes de données lors de l'exécution d'algorithmes itératifs.

- Le groupe *algorithmique numérique* dirigé par Jacques Bahi au LIFC (Besançon) a développé plusieurs algorithmes numériques de type IACA, pour différentes applications scientifiques : modélisées par des EDP, des ODE des systèmes linéaires ou non linéaires. Ces nouveaux algorithmes ne sont pas simplement une exécution asynchrones d'algorithmes synchrones classiques. En effet, ce type d'algorithmes à itérations asynchrones nécessite une étude minutieuse de leur convergence, domaine dans lequel le groupe est compétent. Des expérimentations préliminaires utilisant l'environnement multithreads PM2 sur des grilles réparties à l'échelle nationale ont montré l'efficacité de ce modèle d'itérations et de communications asynchrones sans perte de messages.

- Au LaBRI (Bordeaux), un des axes de recherche développés par le groupe de Raymond Namyst concerne les *supports exécutifs* pour architectures parallèles de type "grappes hétérogènes à haute performance" (réseaux d'interconnexion de type Myrinet ou SCI) et leur interfaçage avec les environnements de programmation de plus haut niveau (MPI, CORBA). La suite logicielle PM2 sert en outre de fondation pour la construction d'un environnement d'exécution pour la grille (PadicoTM) dans le cadre de l'ACI GRID RMI. Dans ce contexte, des résultats particulièrement intéressants ont été obtenus sur l'optimisation des communications (fiables) lorsque l'application exprime des contraintes de haut niveau sur la sémantique des transmissions de données.

- Le projet PARIS (IRISA) a pour objectif général la programmation des grappes de calculateurs pour des applications utilisant des techniques de simulation numérique distribuée. Le projet a en particulier développé la plate-forme de recherche PadicoTM dont le but est d'explorer les problèmes d'intégration de plusieurs intergiciels et exécutifs communicants qui sont nécessaires pour la programmation des grilles de calcul. Les résultats obtenus ont montré la pertinence de l'approche pour les communications (fiables) aussi bien pour la gestion des SAN que des WAN. Les travaux récent ont concerné le support de nouveau type de protocoles de communication comme les flots parallèles.

1.3 Objectifs

D'une manière générale, notre proposition est d'étudier les conséquences de l'utilisation d'un modèle de communication non fiable. Nous nous fixons deux objectifs principaux:

- Le premier objectif concerne l'étude d'une interface de programmation de niveau utilisateur permettant de contrôler la perte de données. En particulier, il s'agit d'offrir des primitives contrôlant quelles parties de données peuvent être perdues dans un message et avec quelle pourcentage. Cette interface doit présenter un certain nombre de propriétés, dont:
  - expressivité: l'interface doit être simple à utiliser,
  - adéquation aux besoins: l'interface doit être en accord avec les besoins du niveau applicatif,
  - performance: l'interface est attendue être plus contraignante à utiliser qu'une interface fiable. Une meilleure performance de communication est bien sûr une très forte raison de son existence.

- Le second objectif de la présente proposition est de généraliser le modèle des IACA au cadre récent des grilles de calcul. Ce modèle dont les itérations ainsi que les communications sont asynchrones, devrait permettre grâce à un environnement de programmation et un protocole de communications adéquat de développer des algorithmes performants et tolérants aux pertes de messages dans les réseaux.

1.4 Action et résultats attendus

Dans cette section, nous analysons les actions que nous comptons entreprendre et les résultats attendus. Les réalisations proposées dans le cadre de l'action sont les suivantes:

1. Définition des besoins/possibilités de communication non fiables
2. Réalisation d'une interface respectant ce point
3. Adaptation éventuelle du protocole de communication à fiabilité ajustable VRP
4. Développement d'algorithmes numériques supportant la perte de données
5. Expérimentation et évaluation

1.4.1 Besoin des applications

Nous souhaitons adapter les IACAs à la perte partielle de messages. Souvent, lorsque l'on envoie des données de grande taille, la perte de message n'est que partielle et concerne seulement une partie du message. Ainsi, plutôt que de considérer qu'un message arrivé incomplet est totalement perdu, il serait très intéressant d'avoir un système de récupération des parties du message effectivement arrivées à destination pour pouvoir les incorporer dans les calculs. Cela permettrait de diminuer encore les temps de mise à jour pour ces données. Une étude expérimentale permettra d'évaluer le gain ainsi obtenu par rapport au contexte précédent.

1.4.2 Mise en œuvre d'une interface de programmation

Un autre aspect concernant le développement des IACAs est, bien entendu, l'implantation sur machine. Actuellement, nous utilisons l'environnement de programmation PM2. Cet environnement est assez bien adapté à nos besoins de par le fait qu'il est multithreadé et qu'il dispose d'un système de RPC. Ces deux fonctionnalités nous permettent de programmer assez aisément les algorithmes itératifs asynchrones.

Néanmoins, il a un certain nombre de limites qui peuvent être des obstacles à une mise en œuvre efficace de nos algorithmes, telles que le manque d'hétérogénéité ou les problèmes de déploiements multi-sites. Ainsi, il apparaît nécessaire de mettre au point une API spécifique aux IACAs qui soit la plus légère possible et qui permette une mise en œuvre efficace sur plusieurs sites et avec des machines hétérogènes. Cela nous permettra de spécifier toutes les fonctionnalités nécessaires aux différentes actions d'un processus impliqué dans un IACA : communication, équilibrage, ajout, suppression...

1.4.3 Mise en œuvre d'un protocole à tolérance de perte ajustable

Alexandre Denis, doctorant dans le projet PARIS, a implémenté le protocole VRP[1] (*Variable Reliability Protocole*) lors d'un séjour à l'ISI/USC (Californie). Les expériences réalisées en 1999 ont montré un gain de bande passante de l'ordre de 400% sur des liaisons longue distance ayant un taux de perte de 5% à 10%. Alors que le débit obtenu entre Los Angeles et Chicago avec TCP/IP était compris entre 1 et 2 Mbit/s, le débit obtenu via VRP oscillait entre 4 et 8 Mbit/s.

Les spécifications du protocole ont besoin d'être revues car son cadre d'utilisation a légèrement varié. Notamment, la recherche conduite sur la sémantique des messages dans l'environnement PM2 devrait permettre de mieux appréhender la nouvelle interface. De même, alors que l'implémentation actuelle de VRP permet d'ajuster le taux et la répartition des perte par message, des extensions peuvent être étudiées afin de mieux supporter les IACAs. Par exemple, il est possible de garantir que tous les éléments d'un vecteur sont effectivement reçus lors une sucession de n messages.

[1] Publié en tant que rapport interne LIP n^{o}2000-11.

Le protocole VRP sera en particulier mis en œuvre dans la plate-forme PadicoTM qui fournira l'abstraction nécessaire pour choisir automatiquement le protocole de communication à utiliser. Si le protocole VRP sera évidemment choisi pour les communications sur les réseaux longue distance, d'autres protocoles peuvent s'avérer plus pertinent pour les réseaux locaux.

Sur une grille de calcul contenant des réseaux de natures très différentes (WAN, LAN, SAN), des protocoles différents sont nécessaires afin d'exploiter au mieux chaque réseau. Cependant, nous ne voulons pas remonter cette hétérogénéité au niveau applicatif. D'un côté, il y a le modèle de programmation proposé aux applications. D'un autre côté, il y a les protocoles des réseaux. L'objectif est d'avoir un modèle de programmation utilisable par les applications qui permette d'exploiter au mieux les réseaux.

1.4.4 Adaptation des applications

Dans le cadre de cette ARC, nous nous intéresserons aux processus d'ajout et de suppression dynamique de noeuds de calcul dans nos IACAs. L'ajout permettrait notamment le déploiement hiérarchique du programme sur les différents sites puis machines. La suppression permettrait quant-à-elle une plus grande souplesse d'utilisation. Par exemple, si l'on utilise un ensemble de machines sur un site donné et que pour une raison quelconque ces machines doivent être libérées, on voudrait pouvoir supprimer ces processus du calcul en déplaçant leur charge de travail sur d'autres machines.

Ceci implique donc d'avoir un mécanisme d'équilibrage de charge entre les différents processus impliqués dans le calcul. En fait, nous voyons principalement deux raisons pour coupler l'équilibrage de charge avec les IACAs :

- lorsque la charge de travail est bien équilibrée sur le système, l'asynchronisme permet de faire un recouvrement calculs-communications efficace, surtout sur des réseaux très fluctuants,
- même si les IACAs semblent les plus efficaces des algorithmes itératifs parallèles, ils ne prennent pas en compte la répartition de la charge de travail sur les processeurs qui, si elle était efficacement gérée, permettrait d'espérer des performances encore meilleures.

Un de nos objectifs est donc d'adapter un mécanisme d'équilibrage de charge aux IACAs. Étant donné l'aspect décentralisé de nos algorithmes, il apparaît que ce mécanisme devra lui aussi être décentralisé.

1.4.5 Expérimentations

Dans le cadre de l'ARC, nous souhaitons étudier l'impact, sur les algorithmes asynchrones que nous allons développer, de l'utilisation de protocole de communication à tolérance de perte ajustable sur des liens distants à débits variables. Une étude comparative avec des protocoles fiables nous permettra d'évaluer l'intérêt qu'il y a à différencier les protocoles de communication selon les liens entre les machines.

Pour effectuer et valider toutes ces études, on se propose de traiter l'application particulière de transport de polluants dans l'air. Ce problème est modélisé par des équations de convection-diffusion. Sa taille, qui peut être considérable, dépend du nombre d'espèces chimiques considérées et de la dimension de la zone géographique dans laquelle se fait l'étude. Il sera nécessaire de réaliser des tests de grande envergure sur plusieurs sites aussi bien au sein de l'INRIA qu'à l'extérieur. Il serait notamment intéressant de tester nos algorithmes sur plusieurs types de réseaux, allant du très rapide (VTHD) au plus lent (Internet) et aussi en mixant ces deux types de liens.

Il est à noter la difficulté de mettre en œuvre de telles expériences notamment à cause des problèmes d'accès aux différents sites ainsi qu'aux systèmes de gestion des grappes. Dans beaucoup de cas, une grande partie des machines ne peuvent communiquer directement avec l'extérieur. Bien que surmontables, ces problèmes vont influencer notre stratégie de déploiement et donc aussi indirectement la conception de nos IACAs.

1.5 Mise en œuvre des travaux

Calendrier

L'action proposée est planifiée sur **deux ans**. Les différents partenaires arrivent dans l'action avec une expérience certaine. L'action a pour but de valider la pertinence des propositions. L'application et une version préliminaire d'un protocole à tolérance de perte ajustable existent déjà.

Plate-forme

Il est envisagé d'utiliser les machines existantes du projet PARIS, du LaBRI, du LIFC. Le protocole VRP requérant l'utilisation du protocole UDP, les pare-feux des différents sites devront être assouplis. Il se peut que cela pose des problèmes sur certains sites.

Il est également envisagé d'utiliser des réseaux haut débit longue distance comme le réseau nationale VTHD. Dans cette situation, nous envisageons d'utiliser ponctuellement les machines d'autres centres INRIA, comme l'ENS Lyon, Grenoble ou Nice.

Coordination des travaux

Il est proposé que la coordination des travaux soit assurée par Christian Pérez (Projet PARIS).

1.6 Ouverture et perspectives

Intégration et validation à grande échelle À plus long terme, nous espérons étendre les résultats obtenus vers une utilisation globale de la grille dans le cadre de calculs scientifiques intensifs. En effet, de par le protocole de communication utilisé ainsi que le modèle algorithmique asynchrone, notre système ne sera pas limité à des applications pas ou très faiblement couplées. Il nous paraît ainsi pertinent d'étudier l'intégration du concept de communication à tolérance de perte ajustable dans les modèles de programmation des grilles de calcul tels que GridRPC, les composants logiciels et/ou le calcul pair-à-pair.

Nouvelles interfaces Il sera également nécessaire de repenser les interfaces de communications pour mieux maîtriser les transferts à fiabilité paramètrable. Les travaux sur la couche de communication Madeleine, à titre exemple, ont progressivement pu dégager l'importance de l'expressivité laissée aux applications pour la qualité des optimisations permises au sein du support exécutif.

Nouveau modèle de programmation De même, il faudra certainement introduire de nouveaux concepts/paradigmes de programmation permettant d'appréhender plus facilement ce type d'applications parallèles, car les environnements/langages actuels ne sont pas adaptés.

Annexes

A Identité et rôle des participants

Projet PARIS

Christian Pérez, chargé de recherche INRIA, gère l'action pour le projet PARIS.

Alexandre Denis, doctorant (3e année), s'intéresse à la gestion des communications nécessaires, en particulier, au support de modèle de programmation adapté aux grilles de calculs. Ainsi, plusieurs exécutifs communiquant peuvent être simultanément et efficacement utilisés. Dans le cadre d'un séjour à l'ISI/USC (Californie), il a travaillé sur les protocoles à tolérance de perte de messages ajustables.

LaBRI

Raymond Namyst, professeur, sera le coordinateur de l'action au LaBRI.

Vincent Danjean, doctorant (3e année), s'intéresse dans le cadre de sa thèse aux problèmes de réactivité des applications aux entrées/sorties, et en particulier au contrôle fin de l'ordonnancement des threads de communications dans les supports exécutifs distribués. Ces aspects sont vitaux dès lors qu'il s'agit de maîtriser le caractère asynchrone des communications dans des applications complexes.

Guillaume Mercier, doctorant (2e année), s'intéresse dans le cadre de sa thèse à l'optimisation des communications sur réseaux rapides (sur des grappes de grande taille) tout en assurant un bon degré de résistance aux pannes. C'est dans ce cadre qu'il sera nécessaire d'étudier comment intégrer des fonctionnalités permettant des communications non fiables dans l'environnement PM2 tout en enrichissant l'expressivité de l'interface utilisateur.

LIFC

Jacques Bahi, professeur, sera le coordinateur de l'action au LIFC.

Sylvain Contassot-Vivier et Raphaël Couturier, maîtres de conférence, travaillent sur le développement des IACA et ont déjà obtenu des résultats sur l'efficacité et la fiabilité de ceux-ci.

Concernant la partie intervenant directement dans cette ARC, l'équipe s'intéresse à la convergence dans des systèmes massivement parallèles asynchrones. Ces travaux de recherche impliquent de nombreux développements algorithmiques et des études de complexité en tenant compte de différents paramètres tels que les communications, l'hétérogénéité, la convergence, l'équilibrage de charge.

De nombreux phénomènes naturels ou technologiques sont modélisés par des systèmes massivement parallèles fortement couplés, c'est le cas dans le domaine des réseaux globaux de machines interconnectées. L'évolution de tels réseaux est souvent non déterministe : l'ordre d'évolution de chaque élément du réseau est non déterminé à priori et peut être aléatoire. L'équipe s'intéresse à l'étude algorithmique de ce type de systèmes en tenant compte de leur aspect chaotique et de l'asynchronisme qui leur est inhérent.

B Aspects budgétaires

Demande des participants

Le soutien financier demandé pour l'ensemble des participants pour mener à bien cette action de recherche est de **36 KE**, réparti équitablement entre les partenaires. Le projet PARIS étant le seul projet INRIA[2], il est prévu qu'il gère les crédits des participants.

Le soutien financier demandé est prévu pour supporter les réunions de travail entre les membres de cette action et pour les présentations de nos résultats dans des conférences. Il est prévu de réaliser 4 réunions par an.

[2]NB: le groupe de Raymond Namyst va déposer cette année une candidature à la création d'un nouveau projet INRIA sur le site de Bordeaux.