Chapitre 8

Conclusion

Ce document correspond à une présentation de mes travaux récents autour de la théorie des graphes, de la complexité, et de la théorie algorithmique des jeux.

Ma recherche est essentiellement motivée par les applications liées à l'algorithmique pour les réseaux sous l'angle de l'optimisation des ressources des réseaux en garantissant certaines performances. Pour chaque problème, après sa modélisation sous forme de problèmes liés aux graphes, nous cherchons à comprendre s'il est possible d'obtenir des garanties sur l'optimisation des ressources. Ceci correspond à des travaux reliés à l'algorithmique en général, centralisée ou distribuée, à la théorie des graphes et à des problèmes de complexité. Depuis quelques temps, nous nous intéressons aussi à prendre en compte la dimension économique dans le sens où les acteurs du système peuvent avoir leur propre intérêt/coût.

Le chapitre 2 illustre par exemple ce dernier point : nous nous sommes intéressées au problème de la construction d'arbre de plus court chemin en présence de partenaires économiques ([10], [42]). Cette étude est une extension d'un problème de routage inter-domaine. Lorsque deux types de coûts concurrents existent, le problème devient NP-complet. En revanche pour des instances ayant une certaine propriété (ne contenant pas de roues avec conflits), l'algorithme réparti glouton calcule un tel arbre et en plus de façon auto-stabilisante.

Tout au long du document, nous avons discuté de quelques suites possibles aux différents travaux de recherche décrits.

Nous reprenons ici quelques éléments de discussions. Cette discussion peut se décliner autour de deux principaux axes

1. Compromis et approximations ;
2. Modèles et algorithmes en présence de partenaires économiques.

Compromis et approximations.

Depuis plusieurs années, un aspect de mon travail concerne l'étude de la frontière entre la NP-complétude et la polynomialité pour différents problèmes reliés aux réseaux et à l'algorithmique sur les graphes.

Le chapitre 3 se consacre à nos résultats relatifs à des algorithmes d'approximation. Ces résultats portent sur la numérotation de sommets sur les graphes d'intervalle, en liens avec la bio-informatique [27] et sur la gestion des ressources dans le contexte de la redistribution de données [14] et dans l'ordonnancement de messages dans les réseaux optiques (essentiellement [7], [20], [21], [33]). Dans ces travaux, nous avons caractérisé pour quels types d'instances le problème devient polynomial.

Le chapitre 4 correspond à une discussion sur les colorations de graphes non-classiques comme la b-coloration [6] et la f-coloration [34]. Tout au long de ce chapitre, nous étudions la frontière entre la polynomialité et la NP-completude pour ces problèmes. Cela inclue par exemple de décider si un graphe possède une telle coloration s'il possède des propriétés sur l'ensemble de ces types de coloration. Cette étude se fait de façon systématique en fonction de la structure des instances (graphe biparti ou pas, avec un degré maximal borné ou non, connaissance ou pas de l'existence d'une telle coloration). Cette étude s'est étendue vers l'inapproximalité des problèmes d'optimisation correspondant.

Il ressort fortement de tous ces travaux que la recherche de solutions exactes est souvent impossible. Il nous semble donc vital de poursuivre nos investigations actuelles sur des outils qui permettent de contourner cet obstacle en s'orientant vers la recherche de compromis ou d'approximations.

Comprendre la frontière entre la polynomalité et l'intractabilité permet de classifier les problèmes et aide à construire des méthodes adaptées pour les résoudre ou les approximer, ou à comprendre les paramètres pertinants. Un exemple connu est de considérer les graphes à largeur d'arbre bornée. En effet, de nombreux problèmes NP-complets sur les graphes généraux s'avèrent polynomiaux lorsque l'on se restreint à des largeurs d'arbres bornées : par exemple lorsque les problèmes peuvent se transcrire en logique monadique du second ordre. Le même type de phénomène se produit pour des types de graphes classiques comme les graphes d'intervalles : de nombreux problèmes NP-complets deviennent polynomiaux, même si des problèmes connus et intensivement étudiés comme la largueur de coupe ne sont pas encore résolus.

Même si un problème est polynomial, l'algorithme calculant la solution optimale peut être coûteux en termes d'opérations en pratique. La résolution par des algorithmes simples adaptés est nécessaire. Pour illustrer ces dires, les entités peuvent être limitées en termes de puissance de calcul (modèle des protocoles de populations) ou les entités peuvent avoir une ressource critique limitée (par exemple l'énergie pour les noeuds dans les réseaux sans fils, le temps pour les réseaux optiques). Avoir un algorithme simple et implémentable implique souvent une dégradation de la qualité de la solution. Cela implique de trouver un algorithme d'approximation correspondant à un compromis entre la simplicité et la qualité de la solution obtenue.

Poursuivre nos investigations sur l'utilisation de la théorie de l'approximation nous paraît nécessaire pour l'optimisation des ressources des réseaux de télécommunications en vue de l'émergence des nouveaux services gourmands en qualité de service.

Cela peut se faire autour de nombreux problèmes. Par exemple, de manière anecdotique, le mécanisme de transport de données MPLS (*Multiprotocol Label Switching* en anglais) correspond à une technique utilisant la commutation de paquets. Cette technique qui garantit une réservation de la bande passante met au goût du jour des travaux de ma thèse qu'il faut étendre à cette problématique particulière.

Plus généralement, comme il n'existe pas dans ce contexte de recette universelle pour résoudre tous les problèmes, nous proposons de continuer à travailler sur les problèmes d'optimisation dans ce contexte au fil des rencontres scientifiques ou via des contrats. Par exemple, nous sommes impliquées dans le contrat européen ETICS qui débute. Notre part du travail correspond à la réservation de ressources à deux niveaux basée sur des techniques de résolution du problème du sac-à-dos multidimensionnel. En outre, chaque niveau du réseau correspond à des entités ayant leur propre intérêt et ce qui complique la modélisation puisqu'il faut introduire la notion de concurrence.

Modèles et algorithmes en présence de partenaires économiques

L'accroissement de la taille des réseaux actuels, fait que le comportement global émergeant des réseaux n'est plus réellement celui prévu par l'algorithmique classique. En effet, les différents partenaires impliqués dans les réseaux sont des sociétés commerciales en nombre croissant qui ont des intérêts économiques propres et divergents. Aussi, il est de moins en moins réaliste de supposer que chacun agit uniquement dans l'intérêt des performances globales de l'ensemble, et l'algorithmique classique répartie, qui présuppose généralement que chacun effectue la tâche qui lui incombe dans l'intérêt général, est de plus en plus mise en défaut.

La prise en compte de la concurrence des acteurs dans un système introduit des difficultés supplémentaires : en particulier, sur la modélisation des systèmes, la construction de solutions satisfaisantes, et sur l'interprétation des solutions. Par exemple, modéliser un jeu correspond souvent à quantifier l'intérêt d'un joueur par le biais d'une fonction d'utilité ou de coût et aussi celle du coût social du système. Cette modélisation est souvent assez techniquement difficile à rendre réaliste. Comment quantifier précisément les préférences lorsqu'elles peuvent être subjectives ?

Le chapitre 5 est un survol des jeux principaux étudiés dans la littérature informatique comme par exemple les jeux basés sur le placement de tâches, ou les jeux de congestion.

A ce jour, les différents travaux en théorie algorithmique des jeux se focalisent principalement sur :

1. la caractérisation des équilibres de Nash purs (ou stables) s'ils en existent ;
2. le calcul de la qualité des équilibres par rapport au coût social (prix de l'anarchie, de la stabilité) ;
3. la construction de tels équilibres de Nash ;
4. l'évaluation des équilibres ainsi calculés en considérant le coût social.

Les deux derniers points peuvent se réaliser de manière soit centralisée ou soit répartie. Une solution centralisée implique souvent un algorithme calculant un équilibre de Nash ayant une vision complète du jeu. L'approche répartie fait souvent l'hypothèse que chaque joueur choisit dynamiquement sa propre stratégie par apprentissage, en supposant que le jeu est réitéré/répété.

En fait, il existe plusieurs façons d'introduire des aspects dynamiques en théorie des jeux. Les chapitres 6 et 7 illustrent deux approches.

Le chapitre 6 porte sur l'étude la répétition d'un même jeu, avec des joueurs qui évoluent selon certains comportements. Nos contributions personnelles relatives à ce chapitre concerne d'une part le calcul de stratégies comportementales pour obtenir des stratégies Pareto optimales dans le contexte du routage inter-domaine [18]. Nous avons étudié le comportement des joueurs fictifs dans ce contexte. D'autre part, nous avons obtenu une caractérisation partielle de la puissance des protocoles de population utilisant des dynamiques de jeux myopes [24].

Le chapitre 7 porte sur l'apprentissage d'équilibres de Nash, selon une classe de dynamiques stochastiques d'apprentissage particulière : le choix stochastique du joueur i

dépend d'une fonction des comportements des autres joueurs et du passé du joueur i. Des algorithmes simples d'apprentissage sur des jeux de potentiel ordinal ont été décrits. Les principaux résultats de convergence d'équilibres purs se basent essentiellement sur la décroissance (en moyenne) de la fonction de potentiel.

Notre objectif est de comprendre ce qu'il est possible de garantir sur des algorithmes en présence de partenaires aux intérêts propres.

Pour celà, il faut, dans le contexte de l'algorithmique réparti :

1. comprendre les notions d'équilibres ou de stabilités pertinentes ;
2. comprendre s'il existe de tels équilibres ;
3. comprendre s'il existe des algorithmes qui les construisent de façon répartie ;
4. comprendre si ces équilibres ainsi construits peuvent être évalués par rapport à un coût social ;
5. comprendre les limites des modèles et méthodes associés.

Nous pensons intimement, que toute l'algorithmique répartie est à repenser selon ces aspects.

La théorie des jeux est un outil naturel. Il faut toutefois avoir conscience que les solutions de la théorie des jeux sont généralement pas directement utilisables. En fait, les développements historiques de la théorie des jeux en mathématiques et économie, ont souvent oublié la nécessité de solutions constructives, efficaces et implémentables. C'est pour cette raison que l'on distingue maintenant souvent la théorie des jeux de la théorie algorithmique des jeux. Par exemple, en théorie des jeux, le théorème de Nash affirme que tout jeu fini possède un équilibre de Nash. Ce théorème garantit l'existence mais n'est pas constructif, et ne dit pas comment ils peuvent se calculer. Des travaux récents de la théorie algorithmique des jeux discutent de la difficulté de les calculer.

Une orientation possible de nos travaux est de transformer (si c'est possible) les problèmes classiques de l'algorithmique répartie (construction d'arbre couvrant de poids minimal, construction de couplages, construction d'ensemble stable dans un graphe,. . .) en prenant en compte ces aspects.

Par exemple, en poursuivant les explorations sur la construction d'arbres de plus court chemin en liens avec le chapitre 2. Dans le chapitre 2, nous avons extrait une famille d'instances qui possèdent des équilibres de Nash purs. La preuve de ce résultat semble possèder des fortes similitudes avec certaines preuves dont l'esprit est de définir un ordre sur les différentes configurations du système et de construire (à partir d'un algorithme) une suite de configurations décroissante en fonction cet ordre. Si un tel ordre peut exister pour la construction répartie d'arbres, cela impliquerait que ce jeu correspond à un jeu de potentiel ordinal. Ce qui impliquerait qu'il existe d'autres algorithmes répartis (comme par exemple ceux décrit dans les chapitres 6 et 7) convergents vers des équilibres de Nash purs.

Il est à noter qu'en algorithmique classique, comparer deux algorithmes peut se faire en évaluant le nombre d'opérations, de messages, ... En théorie algorithmique des jeux, apprendre un équilibre de façon répartie permet de prédire si un système va converger vers une situation stable. Savoir s'il existe plusieurs algorithmes répartis qui le permettent est déjà une information en soit pour comprendre mieux le système et pour pouvoir classifier les comportements des acteurs en fonction de la convergence ou non du système.

Un autre axe est de comprendre quels sont les algorithmes d'apprentissage d'équilibres de Nash mixtes et les propriétés des jeux pour lesquels ils convergent. A ma connaissance, la majorité des algorithmes d'apprentissage (à l'exception des joueurs fictifs) en théorie algorithmique des jeux calculent en fait uniquement des équilibres de Nash purs. En particulier, la majorité de ce type de travaux porte sur des jeux où il en existe toujours au moins un équilibre pur. Des questions naturelles sont de comprendre s'il existe des algorithmes d'apprentissage répartis permettant de calculer un équilibre de Nash mixte et si dans ce contexte, calculer un tel équilibre a du sens. Par exemple, dans le problème de la construction d'arbre de plus court chemin du chapitre 2, la notion d'équilibre de Nash mixte n'est pas réellement pertinante puisqu'à chaque étape, le chemin change et cela implique un routage instable.

Un autre axe est de comprendre si les résultats sont robustes au fait que les acteurs puissent ne pas avoir le même comportement. Dans les différents types d'algorithmes d'apprentissage d'équilibres, bien que les acteurs aient des intérêts différents, on suppose qu'ils utilisent tous le même algorithme de comportement. Uniquement sous ces hypothèses, le système est prouvé converger vers un équilibre. Cette hypothèse est forte dans le sens où maintenant beaucoup d'acteurs peuvent intervenir dans un système en pratique. Il est iréaliste de supposer que tout le monde exécute le même programme (aie le même comportement). Il peut aussi plus généralement être naturel de supposer que les acteurs puissent avoir un comportement byzantin (dans le sens imprévisible ou nuisible). Il faut comprendre quels sont les comportements comptatibles pour atteindre une situation stable, quelle est la robustesse des algorithmes en présence d'acteurs byzantins.