Chapitre 5

Conclusions et perspectives

Dans ce document, nous avons présenté plusieurs résultats nouveaux, nous avons résumé l'état de l'art relatif à plusieurs domaines de recherches, et présenté quelques points de vue.

Tout d'abord, dans les chapitres 1 et 2, nous avons rappelé toutes la complexité et la richesse des systèmes dynamiques continus. Nous avons pour cela présenté un certain nombre de systèmes complexes, aux dynamiques intéressantes et souvent subtiles.

Les exemples du chapitre 1 sont presque tous issus de l'excellent ouvrage [Hirsch et al., 2003]. Les exemples du chapitre 2 illustrent des modèles de systèmes avec certaines concurrences, provenant de sources diverses comme la bioinformatique, la biologie, la virologie, et la virologie informatique. Il présente aussi plusieurs modèles venant de la théorie des jeux, ou en rapport avec l'algorithmique distribuée.

Par ces exemples, nous avons cherché à montrer plusieurs faits. Tout d'abord, que tous ces systèmes correspondent à une classe particulière d'équations différentielles, que nous avons nommé les problèmes de Cauchy polynomiaux, à propos de laquelle nous revenons à plusieurs reprises dans les chapitres ultérieurs. D'autre part, que de nombreux systèmes, naturels ou artificiels, peuvent intrinsèquement êtres considérés comme des modèles de calcul. Enfin, nous avons argumenté par plusieurs exemples, que même pour des systèmes à espace et temps discret, le bon modèle d'abstraction est souvent le continu.

Plus spécifiquement, dans le chapitre 2, nous avons cherché à montrer l'intérêt et le potentiel de l'utilisation des modèles continus, pour l'algorithmique distribuée. Nous avons discuté plusieurs problèmes importants qui se posent alors.

Dans le chapitre 3, nous avons présenté un survol, que nous estimons relativement complet, de la théorie, ou des théories, des calculs pour les systèmes à temps continu. Nous avons montré que ces théories permettent à la fois de comprendre la difficulté des questions relatives aux systèmes dynamiques à temps continu, et à la fois de comprendre la puissance des modèles analogiques à temps continu. Forts de ces deux perspectives, les résultats obtenus et présentés sont motivés par des domaines aussi variés que la vérification, la théorie du contrôle, la conception de circuits intégrés , les réseaux de neurones, l'analyse numérique, ou la théorie de la récursion sur les réels. Nous avons discuté à la fois les aspects calculabilité, complexité, robustesse aux bruits et aux imprécisions, en laissant une grande part dans le texte à des discussions de problèmes ouverts et de perspectives.

Les chapitres 4, 5, et l'annexe A présentent des panoramas synthétiques de certains de nos résultats.

Dans le chapitre 4, nous avons montré qu'il était possible de relier les fonctions solutions de problèmes de Cauchy polynomiaux aux fonctions GPAC calculables, ce qui était connu, mais aussi aux fonctions calculables en analyse récursive : les fonctions calculables par GPAC, dans un certain sens très naturel, correspondent exactement aux fonctions calculables au sens de l'analyse récursive. Nous avons par ailleurs, de façon orthogonale, montré qu'il était possible de caractériser les fonctions calculables (et élémentairement calculables) au sens de l'analyse récursive comme une classe de fonctions $\mathbb{R}$-récursives. À notre connaissance, c'est la première fois qu'une caractérisation algébrique des fonctions calculables au sens de l'analyse récursive aussi naturelle est obtenue.

Dans le chapitre 5, nous avons présenté un catalogue de l'ensemble de nos caractérisations des classes de complexité dans le modèle de Blum Shub et Smale. Toutes les caractérisations sont dans l'esprit de la caractérisation par Bellantoni et Cook du temps polynomial dans [Bellantoni and Cook, 1992]. Nous présentons ainsi une caractérisation de presque toutes les classes considérées dans le livre [Blum et al., 1998] sur le modèle de Blum Shub et Smale (à vrai dire, il ne manque essentiellement qu'une caractérisation de NC).

Dans l'annexe A, nous discutons de la question de l'éventuelle surpuissance des systèmes continus par rapport aux modèles classiques, comme les machines de Turing. Nous présentons plusieurs mauvaises compréhensions fréquentes de ce que dit la thèse de Church, et nous caractérisons la puissance de plusieurs classes de modèles à espace continu. Cette annexe contient plusieurs résultats originaux, et peut se voir comme une remise à jour des résultats qui existaient lorsque nous avions débuté notre thèse.

Perspectives

Chacun des chapitres est volontairement écrit avec une partie importante laissée aux perspectives : en particulier, le chapitre 3 possède toute une discussion des perspectives et travaux futurs qui nous semblent intéressants en rapport avec la théorie des calculs des systèmes à temps continu.

Mais à vrai dire, les chapitres 1, et 2 contiennent aussi de nombreuses questions explicitement citées comme telles. Chacune de ces questions fondamentales peut mener à des travaux pour plusieurs années. Le chapitre 4 offre lui aussi une partie conclusion et perspectives avec des pistes de travaux futurs.

Devant ce foisonnement de questions, nous nous permettons ici de souligner celles qui nous paraissent essentielles à l'heure où nous écrivons ces lignes.

Comprendre s'il existe un concept unificateur pour les modèles à temps continu comme la thèse de Church.

La situation est loin d'être aussi claire que pour les modèles discrets. Bien qu'il ait été montré que certains modèles à temps continu possèdent des capacités hypercalculatoires, tous ces résultats se basent sur l'utilisation d'une certaine quantité infinie de ressources, comme le temps, l'espace, la précision ou l'énergie. En général, il est souvent conjecturé que les modèles à temps continu "raisonnables" ne peuvent pas calculer plus que les machines de Turing.

Puisque les systèmes à temps continu robustes et analytiques peuvent simuler les machines de Turing avec un espace non-borné, nous pensons que les calculs digitaux et analogiques sont également puissants du point de vue de la calculabilité. En outre, comme nous l'avons vu, plusieurs résultats récents établissent l'équivalence entre les fonctions calculables par des équations différentielles polynomiales, les fonctions calculables par GPAC, et les fonctions calculables dans le sens de l'analyse récursive. Ce type de résultats renforce l'idée qu'il pourrait y avoir un cadre unifié pour les calculs à temps continu, similaire à celui qui existe en théorie de la calculabilité classique.

Nous avons argumenté tout au long de ce document de la puissance de modélisation, et des propriétés remarquables des fonctions solutions de problèmes de Cauchy polynomiaux. Nous pensons que cette classe de fonctions est un réel candidat pour fournir un concept unificateur, et des bases solides, à une théorie unifiée des calculs pour les systèmes continus.

Comprendre s'il existe une théorie de la complexité bien fondé, élégante et robuste pour les modèles à temps continu.

Nous avons vu dans le chapitre 3, que plusieurs pistes ont été considérées dans la littérature pour construire des théories de la complexité pour les systèmes continus. À vrai dire, à ce jour, il n'y a pas d'accord général entre les auteurs sur les définitions de base comme le temps de calculs, ou la taille des entrées. Les résultats établis à ce jour sont dérivés de concepts intrinsèques aux systèmes à temps continu considérés.

Puisque l'analyse récursive est un cadre bien établi et bien compris pour l'étude des problèmes de complexité pour les systèmes continus, nous pensons que mieux comprendre les relations qui existent entre les différentes approches et l'analyse récursive est de première importance.

En relation avec le chapitre 4, nous pensons qu'une façon d'attaquer le problème est de tenter de caractériser les fonctions calculables en temps polynomial en analyse récursive comme une classe de fonctions

$\mathbb{R}$-récursives. Peut-on étendre des caractérisations à la Bellantoni et Cook [Bellantoni and Cook, 1992] à l'analyse récursive ? Nous avons montré que les fonctions calculables par GPAC correspondaient aux fonctions calculables en analyse récursive. La simulation utilisée semble relier fortement le temps de calcul du GPAC et de la machine de Turing utilisée. Peut-on établir une telle équivalence au niveau de la complexité, et pas seulement de la calculabilité ?

Si la classe des fonctions solutions de problèmes de Cauchy polynomiaux permet, comme nous le conjecturons plus haut, de caractériser de façon élégante la notion de calcul "raisonnable" en temps continu, ne peut-on formuler simplement une théorie de la complexité basée sur ces fonctions ?

Mieux comprendre les modèles, et leurs propriétés.

Nous avons vu dans le chapitre 3 que très peu de travaux de recherches ont été faits à ce jour en ce qui concerne les effets du bruit ou des imprécisions sur les calculs à temps continu. À ce jour, l'essentiel des résultats concerne les systèmes à temps discret.

Nous avons évoqué plusieurs façons de modéliser le bruit ou l'incertitude : par exemple, en utilisant des modèles de bruit probabiliste, ou des modèles de bruit non-déterministe. Nous avons montré que chacune de ces façons mène à des résultats réellement différents. Mieux comprendre tout cela nous semble fondamental.

Par exemple, de nombreuses questions ouvertes surgissent si l'on demande si les résultats d'indécidabilité pour les systèmes à temps continu restent vrais pour les systèmes robustes. Cela est de première importance dans le domaine de la vérification par exemple, puisque cette question est reliée de façon très forte à la terminaison des procédures automatiques de vérification. Une meilleure compréhension des hypothèses avec lesquelles du bruit mène à de la décidabilité ou de l'indécidabilité nous semble nécessaire.

Nous pensons en outre qu'à ce jour, la question a uniquement été adressée avec un point de vue de la calculabilité, et pas de la complexité. Lorsque les problèmes sont prouvés décidables, quelle est l'influence de ces notions de bruit sur la complexité des problèmes ? Comment croit-elle avec la précision, ou le bruit ?

Nous pensons ces questions au coeur de problèmes fondamentaux reliés à des questions très profondes sur les modèles mathématiques actuels de notre monde physique : comment modéliser le bruit ? Qu'est-ce qu'un modèle robuste ? Comment modéliser l'incertitude ? Est-il pertinant, et quel est le sens, de chercher à faire des preuves formelles, donc d'une certaine façon certaines, à propos de phénomènes ou de systèmes incertains ?

Utiliser les modèles continus en algorithmique distribuée.

Nous avons volontairement insisté dans le chapitre 2 sur le fait que les systèmes continus apparaissent naturellement lorsqu'on cherche à discuter de systèmes de grandes tailles.

À ce jour, l'algorithmique distribuée n'est pas traitée avec ces outils. Une question fascinante est de comprendre si les théories des calculs des systèmes continus peuvent aider à comprendre, à programmer, et à maîtriser les modèles massivement parallèles. Étant données la taille des réseaux actuels, et la remise en cause de plusieurs hypothèses fortes de l'algorithmique distribuée classique dans certaines applications basées sur des modèles comme les réseaux de capteurs, il nous semble urgent de s'intéresser à cette question.

Nous avons présenté dans le chapitre 2 plusieurs modèles allant dans ce sens. Par exemple, nous avons montré que le modèle des protocoles de populations possède des propriétés calculatoires originales en termes de relations définissables en arithmétique de Presburger. Nous avons vu que ces protocoles peuvent être généralisés pour parler de grandes populations. La puissance exacte de modèles dans cet esprit reste très largement à comprendre.

Plus globalement, pour déterminer si les systèmes continus peuvent contribuer de façon profonde à l'algorithmique distribuée, plusieurs pistes semblent prioritaires.

Tout d'abord, il convient de bien et mieux comprendre les hypothèses qui permettent de passer, dans des domaines comme la physique, la biologique, la virologie, d'un système discret à une abstraction continue, pour comprendre de façon fine quand cela peut être possible en algorithmique distribuée. Passer au continu sur des algorithmes distribués n'est pas sans poser plusieurs difficultés spécifiques. Par exemple, jusqu'à quel point peut-on ignorer les informations topologiques, comme cela est fait dans beaucoup de modèles de populations ?

Étant donné que dans un système distribué de grande taille, on ne peut espérer programmer, ou contrôler qu'une partie des agents, il nous semble que les modèles basés sur la théorie des jeux constituent de vraies pistes pour cela. Ces modèles sont intrinsèquement continus, et la littérature très récente foisonne d'exemples de systèmes qui se modélisent naturellement par ces outils.

Cependant, à ce jour, ces théories ont été développées dans un cadre mathématique assez abstrait, et parfois qui se mélange mal aux modèles et concepts classiques en algorithmique distribuée.

Par exemple, en algorithmique distribuée, l'évaluation de la complexité des algorithmes est souvent réalisée au pire cas. Le pire cas correspond à une certaine notion d'adversaire, mais qui est relativement différente de celle utilisée en théorie des jeux. Arriver à mélanger les notions d'adversaire de la théorie des jeux et de l'algorithmique classique est un vrai challenge.

En outre, à ce jour, il n'y a quasiment aucuns travaux sur les aspects dynamiques. La théorie des jeux permet de discuter les notions d'équilibre rationnel, mais pas le dynamisme. Des modèles comme ceux qui sont évoqués dans le chapitre 2, basés sur les jeux répétés ou la théorie évolutionnaire des jeux sont conçus pour modéliser le dynamisme de situations de concurrence. Mais presque aucune application de ces théories à l'algorithmique distribuée n'a été faite à ce jour. Réussir à modéliser le dynamisme, pour l'exploiter, des situations de concurrence dans des algorithmiques distribués nous semble un autre challenge de première importance.

Nous pensons que l'ensemble des résultats dans ce document peut apporter de façon significative à la compréhension de l'algorithmique, et de la complexité des modèles actuels et futurs. Cela est le sens de plusieurs de nos travaux scientifiques en cours autour de l'utilisation de la théorie algorithmique des jeux pour l'algorithmique distribuée.

Nous croyons en particulier fermement que le spectre futur des applications de la théorie des calculs des systèmes à temps continu est loin d'être limité à ses applications à ce jour, en particulier à la liste d'applications citée dans le chapitre 3.

Bibliographie

[Bellantoni and Cook, 1992] Bellantoni, S. and Cook, S. (1992). A new recursion-theoretic characterization of the poly-time functions. *Computational Complexity*, 2 :97–110.

[Blum et al., 1998] Blum, L., Cucker, F., Shub, M., and Smale, S. (1998). *Complexity and Real Computation*. Springer-Verlag.

[Hirsch et al., 2003] Hirsch, M. W., Smale, S., and Devaney, R. (2003). *Differential Equations, Dynamical Systems, and an Introduction to Chaos*. Elsevier Academic Press.