Projet de recherche

Nathanaël Fijalkow

Organisation du document. Ce document est divisé en quatre parties. La première partie est une introduction. Les deuxième et troisième parties développent les deux lignes directrices du projet : la notion d'approximation et celle de complexité spatiale en ligne. Enfin, la quatrième partie aborde la réalisation de ce programme de recherche : intégration dans les laboratoires et équipes, ainsi que collaborations nationales et internationales.

Introduction

Importance des systèmes aléatoires

L'aléa intervient en informatique dans deux contextes.
Le premier contexte est lors de la construction d'algorithmes, où l'aléa est utilisé pour des raisons d'efficacité ; en effet, dans de nombreux cas les algorithmes randomisés sont plus efficaces que leurs homologues déterministes. Les algorithmes randomisés sont aujourd'hui très répandus en pratique et constituent un sujet de recherche actif [MR95].
Le second contexte est l'étude de systèmes dont l'évolution dépend de facteurs extérieurs, qui sont modélisés mathématiquement par un aléa. Ce cadre dépasse naturellement l'informatique fondamentale, et les systèmes aléatoires sont étudiés dans de nombreux domaines, que ce soit en linguistique, en biologie ou en économie.

Ces deux contextes témoignent de l'importance des systèmes aléatoires et de la nécessité de les modéliser, puis de développer des outils mathématiques et informatiques pour les analyser.

Modélisation

Je décris un certain nombre d'aspects apparaissant dans la modélisation de systèmes aléatoires.
Le premier aspect est *l'évolution du système*, séparant systèmes continus et systèmes à temps discret : dans le premier cas, on observe l'évolution à tout moment, par exemple pour les systèmes temporisés, et dans le second, seulement à intervalles de temps fixés.
Le deuxième aspect est *le nombre d'agents* : plusieurs entités peuvent influencer l'évolution du système. Dans le cadre le plus simple, elle est entièrement aléatoire, il s'agit de chaînes de Markov. Selon les scénarios envisagés, on peut considérer un ou plusieurs contrôleurs, ainsi qu'un ou plusieurs agents extérieurs ; les modèles que l'on obtient sont appelés des jeux.
Le troisième aspect est *la notion d'observation* : chaque agent peut observer tout ou partie de l'évolution du système. Ainsi, s'il n'y a qu'un contrôleur, cette distinction implique deux modèles : d'un côté les processus de décision markoviens (Markov Decision Processes, MDP), où le contrôleur observe toute l'évolution du système, et de l'autre les automates pro-

babilistes, où le contrôleur n'observe rien. Ces deux modèles sont unifiés par les processus de décisions markoviens à observation partielle (Partially Observable Markov Decision Processes, POMDP).

Les distinctions ci-dessus décrivent une très grande variété de modèles et de travaux. Je m'intéresse en particulier aux modèles à observation partielle et à temps discret.

Problématique générale

L'étude des systèmes aléatoires en informatique fondamentale a deux objectifs principaux :

- *Vérification* (Model Checking) : étant données une instance et une propriété, déterminer si l'instance satisfait la propriété.
- *Synthèse* : étant donnée une spécification, synthétiser un système satisfaisant cette spécification.

Ces deux problèmes, décrits ici de manière générique, sont motivés par les méthodes formelles. L'objectif est de raisonner sur les modèles mathématiques, en décrivant leurs spécifications à l'aide de formalismes logiques. Cette approche, développée originellement par Clarke, Emerson et Sifakis, et récompensée d'un prix Turing en 2007, est aujourd'hui l'objet de nombreuses applications industrielles.

L'étude des systèmes aléatoires n'est pas confinée à l'informatique fondamentale, elle est également au coeur de domaines connexes, dont le traitement du signal, de l'image, de la vidéo, l'automatique et la théorie du contrôle. Je souhaite étudier et renforcer les liens avec ces domaines.

Systèmes aléatoires à observation partielle

Les méthodes formelles pour les systèmes aléatoires *à observation parfaite* forment aujourd'hui un ensemble de techniques mûr et cohérent. De nombreux problèmes sur les chaînes de Markov [AAGT15] et les processus de décisions markoviens [BFS12] ont été analysés et compris, permettant leur insémination dans différents domaines. Par exemple, une grande partie des algorithmes construits ont été implantés dans l'outil PRISM [KNP07].

La situation est différente pour les systèmes aléatoires *à observation partielle*, par exemple pour les processus de décisions markoviens à observation partielle (POMDP). Même dans le cas particulier des automates probabilistes, les méthodes connues ne s'appliquent pas [GO10]. Deux approches récentes ont permis d'avancer dans l'étude des systèmes aléatoires à observation partielle, ce sont le terreau de mon projet de recherche.

Théorie des nombres algébriques. Au cours des années précédentes, les travaux du groupe de Ouaknine et Worrell ont permis de relier certaines résultats et conjectures de théorie des nombres algébriques avec des problèmes sur des systèmes aléatoires à observation partielle. Par exemple, l'article [AAOW15] montre l'équivalence entre l'étude des chaînes de Markov et le problème de Skolem, dont la décidabilité est un problème ouvert depuis plus de 80 ans.

Algèbre et Topologie. L'étude des automates à compteurs, ainsi que menée par Bojańczyk et Colcombet [BC06], a mis en évidence la notion de monoïde de stabilisation [Col09]. Cet outil algébrique, dont l'utilisation s'appuie sur le théorème de factorisation de Simon [Sim90] et les relations de Green [Col11], a permis de résoudre un des principaux problèmes ouverts de la théorie des automates, la hauteur d'étoile.
Les espaces profinis ont été introduits en théorie des automates par Almeida, Weil et Pin [Pin09]. Cette approche topologique a permis d'obtenir des résultats génériques sur les caractérisations de classes de langages [GGP10].
Au cours de ma thèse, j'ai adapté ces deux outils pour l'étude des automates probabilistes. Les monoïdes de stabilisation sont au coeur de l'algorithme de Markov [FGO12; FGKO15], et les espaces profinis [Fij16a] permettent une analyse précise de cet algorithme en introduisant la notion de vitesse de convergence.

Les systèmes aléatoires à observation partielle sont utilisés dans de nombreux domaines tels que la recherche opérationnelle, l'intelligence artificielle et la planification de mouvements en robotique. Le développement de méthodes formelles pour analyser ces systèmes est un enjeu important, qui est en plein essor depuis une dizaine d'années.

Mon projet de recherche participe de cet effort, en suivant deux lignes directrices : la notion d'approximation et la complexité spatiale en ligne.

Première ligne directrice du projet : approximation

Nous étudions des modèles mathématiques de systèmes aléatoires. Pour plusieurs raisons, le comportement du système peut s'écarter de la loi de probabilité décrite par le modèle : soit le système en question est infiniment plus compliqué, soit il est sujet à des perturbations extérieures imprévisibles, soit encore il n'est pas connu de manière très précise. La notion d'approximation est donc centrale dans la modélisation de systèmes aléatoires. Elle apparaît sous deux aspects.

Le premier est la perturbation du modèle. Le système pouvant subir des perturbations, l'analyse doit prendre en compte la possibilité de faibles déviations du modèle.

Le second est la spécification de propriétés souples. Il n'est pas pertinent de spécifier une propriété de manière "rigide", par exemple en demandant que la probabilité de succès soit exactement 50%. Une propriété souple introduit un intervalle de confiance, prenant en compte le caractère changeant du modèle.

L'approximation est reliée à la robustesse d'un système, qui quantifie l'impact de modifications légères du système sur son comportement. Un système est robuste si ses approximations ont un comportement proche du système original.

Les questions d'approximation et de robustesse ont été identifiées comme un enjeu fondamental pour les systèmes temporisés par Henzinger et Sifakis [HS06], elle est aujourd'hui un axe de recherche porteur, développé en particulier par Sankur [San13]. Bien que suggérée par exemple par Rabin pour les automates probabilistes [Rab63], cette même notion pour les systèmes aléatoires est restée longtemps plus marginale. Elle est aujourd'hui étudiée dans de nombreux modèles, par exemple pour les jeux stochastiques temporisés [ORS14].

Direction 1 : perturbations des transitions d'un automate probabiliste

Dans ce paragraphe, $\mathcal{A}$ et $\mathcal{B}$ sont des automates probabilistes. Le problème suivant est fondamental dans l'étude des automates probabilistes : étant donné $\mathcal{A}$, existe t-il un mot accepté par $\mathcal{A}$ avec probabilité avec moins $\frac{1}{2}$? Malheureusement, ce problème est indécidable [Paz71].

Condon et Lipton ont considéré une variante de ce problème, introduisant de la souplesse dans la propriété ; leur résultat majeur [CL89] énonce qu'il n'existe pas d'algorithme qui étant donné $\mathcal{A}$ a le comportement suivant :

— s'il existe un mot accepté par $\mathcal{A}$ avec probabilité au moins $\frac{3}{4}$, alors l'algorithme répond oui ;

— si tous les mots sont acceptés par $\mathcal{A}$ avec probabilité au plus $\frac{1}{4}$, alors l'algorithme répond non.

Dans tout autre cas, l'algorithme n'est pas spécifié, et peut ne pas terminer.

Récemment, nous avons considéré des automates probabilistes dont les transitions sont perturbées, en collaboration avec Hugo Gimbert, Florian Horn et Youssouf Oualhadj [FGHO14].

Nous avons montré que le problème suivant est indécidable : étant donné $\mathcal{A}$ et $\varepsilon > 0$, existe t-il $\mathcal{B}$ obtenu en perturbant les transitions de $\mathcal{A}$ d'au plus ε tel que $\mathcal{B}$ accepte un mot avec probabilité au moins $\frac{1}{2}$?

Ces deux résultats montrent que ni considérer des propriétés plus souples ni autoriser des perturbations du modèle ne permettent d'obtenir des résultats de décidabilité.
Cependant, la combinaison de ces deux approches est prometteuse ; le problème suivant étudie une propriété souple, en introduisant des perturbations dans le modèle.
Existe t-il un algorithme ayant le comportement suivant : étant donné $\mathcal{A}$ et $\varepsilon > 0$, existe t-il $\mathcal{B}$ obtenu en perturbant les transitions de $\mathcal{A}$ d'au plus ε tel que :

- s'il existe un mot accepté par $\mathcal{B}$ avec probabilité au moins $\frac{3}{4}$, alors l'algorithme répond oui ;
- si tous les mots sont acceptés par $\mathcal{B}$ avec probabilités au plus $\frac{1}{4}$, alors l'algorithme répond non.

Un tel algorithme formerait la pierre angulaire pour étudier les propriétés algorithmiques des automates probabilistes et des processus de décision markoviens à observation partielle. En effet, de nombreux algorithmes de vérification pour des logiques temporelles probabilistes se réduisent à ce problème. De même, ceci ouvre la possibilité d'étudier la synthèse de programmes avec observation partielle.

Il est possible que ce problème s'avére indécidable ou trop difficile. Il existe plusieurs leviers permettant de le simplifier, sans perdre les applications entrevues. Par exemple, on peut imposer des conditions structurelles sur les automates, ce qui est la démarche que j'ai suivi dans mes travaux de thèse pour l'étude des automates "résistants aux fuites" (leaktight).

Direction 2 : distances entre systèmes aléatoires

La notion d'approximation peut être vue de manière plus générale ; il s'agit de définir une notion de distance entre systèmes aléatoires.

De nombreuses notions de distance, et en particulier d'équivalence ou de bisimilarité, ont été étudiées au cours des années précédentes, par exemple par Chen et Kiefer [CK14]. Dans le cadre de mon post-doctorat, nous étudions avec Stefan Kiefer et Mahsa Shirmohammadi la notion de bisimilarité pour les processus de décision markoviens. Des premiers résultats ont donné lieu à la publication [FKS16].

Un deuxième exemple est donné par la pseudo-distance de Kulback-Leibler, qui quantifie l'écart entre les distributions de probabilité générées par deux systèmes. Des premiers résultats existent pour approcher un système aléatoire par un système plus simple à analyser, tout en minimisant la distance des deux systèmes vis-à-vis de cette pseudo-distance. Éric Fabre et Blaise Genest de l'équipe SuMo utilisent ces techniques pour étudier les réseaux bayésiens dynamiques, avec des applications en biologie moléculaire.

Obtenir des notions de distance entre systèmes aléatoires est un enjeu important ayant de nombreuses applications. Par exemple, ceci ouvre la voie vers la minimisation de systèmes : étant donné un système, peut-on construire un système proche, plus petit, et donc dont l'analyse est facilitée ? Une autre application est l'estimation de la perte de précision lors de l'abstraction : en évaluant la distance entre le modèle et le système réel, on obtient des garanties sur la qualité du modèle.

Direction 3 : approximer pour contourner le problème de Skolem

Ainsi que l'ont observé Ouaknine et Worrell, l'étude des chaînes de Markov est reliée au problème de Skolem. Ce problème étant ouvert depuis 80 ans, et l'objet de nombreuses conjectures en théorie des nombres algébriques, il n'est pas raisonnable de chercher à le résoudre de front. De plus, il est probable que les instances difficiles du problème de Skolem ne soient pas pertinentes en pratique.

La notion d'approximation apparaît donc ici comme une solution ; afin d'éliminer les instances difficiles, on considère le système à perturbations près. J'ai commencé à aborder ces questions avec Nathalie Bertrand, Blaise Genest et S. Akshay, pour les chaînes de Markov et pour les processus de décisions markoviens.

Objectif à long terme et intégration

Développer une théorie mathématique des systèmes aléatoires à approximation près, étendant les modèles et les résultats des systèmes aléatoires en y intégrant la notion d'approximation. Cette théorie va permettre de prendre en compte naturellement les erreurs de précision, d'assurer des propriétés de robustesse, et de résoudre des problèmes que l'on ne sait pas résoudre dans les modèles classiques.

Cet axe de recherche est aligné avec les thématiques de l'ANR STOCH-MC (dont je fais partie), qui rassemble en particulier Nathalie Bertrand, Blaise Genest et Ocan Sankur à l'IRISA, et Hugo Gimbert au LaBRI.

L'équipe SuMo à l'IRISA étudie les systèmes aléatoires, en rassemblant plusieurs points de vue : le point de vue de l'informatique fondamentale, mais également du traitement du signal et de l'ingénieurie en électronique. Cette combinaison de compétences permet d'élargir les perspectives, et intégrer cette équipe serait pour moi une excellente opportunité de développer cet axe de recherche.

L'équipe Méthodes Formelles au LaBRI étudie l'application des méthodes formelles dans les systèmes complexes, et en particulier les systèmes aléatoires. De par la diversité des modèles étudiés, mêlant concurrence, aspects temporisés et aléatoires, c'est un terreau idéal pour mener l'étude des systèmes aléatoires à observation partielle à approximation près.

L'équipe MoVe au LIF possède une expertise variée sur la modélisation et la vérification. En particulier, Benjamin Monmege et Pierre-Alain Reynier étudient diverses notions de robustesse pour les systèmes quantitatifs, en relation avec cet axe de recherche.

Seconde ligne directrice du projet : algorithmes en ligne

La question de Rabin

Cet axe de recherche est motivé par une question posée par Rabin, à propos de la complexité en ligne des automates probabilistes. Considérons un automate probabiliste, on souhaite construire un algorithme en ligne simulant cet automate ; l'algorithme lit les lettres l'une après l'autre, et doit à chaque instant être capable de déterminer la probabilité d'acceptation du mot lu.

Dans son article fondateur en 1963, Rabin définit les automates probabilistes [Rab63], et pose de nombreuses questions, qui ont eu d'importants échos et ont permis le développement de ce modèle de calcul. La dernière partie de cet article concerne la complexité en ligne ; le résultat principal de cette partie est d'exhiber une sous-classe d'automates probabilistes ayant une faible complexité en ligne. En d'autres termes, pour chaque automate dans cette sous-classe on peut construire un algorithme en ligne efficace pour le simuler. Dans la conclusion de cette partie, Rabin annonce que cette propriété n'est pas vraie en général, sans donner plus de détails, laissant cette question ouverte.

J'ai montré dans l'article [Fij16b] que l'intuition de Rabin est exacte en ce qui concerne les algorithmes en ligne déterministes : il existe un automate probabiliste dont la complexité spatiale en ligne déterministe est maximale. Ceci est une première étape vers la résolution de la question de Rabin, qui constitue un programme de recherche ambitieux et que l'on peut formuler ainsi :

- Peut-on simuler en ligne et de manière efficace un automate probabiliste, au moyen d'un algorithme déterministe, non-déterministe, alternant ?
- Peut-on décrire la classe des automates probabilistes que l'on peut efficacement simuler en ligne ?

Importance des algorithmes en ligne

À l'opposé d'un algorithme classique, un algorithme en ligne a un accès restreint à son entrée ; il ne peut lire l'entrée qu'une seule fois, de manière séquentielle.
Considérer des algorithmes ayant un accès restreint à leur entrée apparaît dans de nombreux scénarios ; la notion d'algorithme en ligne a été identifiée comme un objet d'étude fondamental depuis les années 80.

Le premier exemple est celui des ensembles de données à très grande échelle, par exemple en bases de données, ou dans de très larges réseaux. Le volume de ces données est si important qu'il n'est pas possible de les stocker afin de les analyser à loisir. Il y a essentiellement deux solutions à ce problème : ne considérer qu'une partie des données, c'est le point de vue du test de propriété (Property Testing), ou traiter l'information en ligne, au fur et à mesure de son arrivée, c'est le point de vue du calcul en ligne.
Une réalisation concrète de cette problématique apparaît dans le traitement de fichiers XML,

qui est le langage de balise utilisé pour l'échange de fichiers entre systèmes d'informations hétérogènes, et sur Internet en particulier. Plusieurs approches (API) permettent de lire et traiter un document XML ; les méthodes DOM et SAX sont deux interfaces de programmation instanciant les deux approches les plus courantes. La méthode DOM charge l'intégralité d'un document XML dans une structure de données arborescente, ce qui peut s'avérer impossible lorsque le document est trop gros. À l'inverse, la méthode SAX apporte une alternative en parcourant le document XML élément par élément, c'est-à-dire au moyen d'un algorithme en ligne.

Le deuxième exemple est celui des systèmes temps réel, c'est-à-dire évoluant au cours du temps. Ceci inclut une grande variété de systèmes, qui apparaissent en robotique, dans les réseaux de télécommunications ou en développement matériel. Lorsque l'on souhaite vérifier ou contrôler l'évolution d'un tel système, les décisions doivent être prises immédiatement, on doit donc construire des algorithmes en ligne. Pour cette raison, les algorithmes en ligne sont étudiés en théorie du contrôle.

Modélisation

L'étude des algorithmes en ligne a donné lieu à de très nombreux développements ; je commence par discuter ici deux approches, avant d'introduire une troisième approche, la complexité spatiale en ligne.

Approches historiques. La première approche est celle des algorithmes dynamiques, introduits par Patnaik et Immerman [PI94], dans le but de maintenir une solution à un problème dont l'entrée est modifiée au cours du temps. La difficulté ici est de conserver suffisamment d'information pour transformer une solution pour une entrée en une solution pour l'entrée modifiée. La deuxième approche est celle des algorithmes de streaming, introduits par une série de papiers (Munro et Paterson [MP80], puis Flajolet et Martin [FM85], suivi du papier fondateur de Alon, Matias et Szegedy [AMS96]). Un algorithme de streaming est limité à la fois en espace et en temps de calcul, et tâche de calculer des statistiques sur l'entrée lue, par exemple sur les fréquences de distributions de lettres et de motifs.

Complexité spatiale en ligne. Une troisième approche, issue de la théorie des automates, a été étudiée dans les années 1960 ; il s'agit de la complexité spatiale en ligne. Le point de départ, proposé par Karp [Kar67], est la définition suivante : un algorithme en ligne est représenté par un automate, potentiellement infini. En effet, un automate est une machine disposant d'une mémoire (l'ensemble de ses états, potentiellement infini) et qui lit son entrée une fois et une seule de manière séquentielle. La taille d'un automate est la fonction $\mathbb{N} \to \mathbb{N}$ associant à un entier n le nombre d'états accessibles depuis l'état initial en lisant un mot de longueur au plus n. Étant donné un langage, sa complexité est la taille du plus petit automate le reconnaissant.

Spécificité de la complexité spatiale en ligne. L'idée de définir un algorithme en ligne comme étant un automate, proposée par Karp [Kar67], abstrait l'aspect calculatoire inhérent aux ma-

chines de Turing, qui sont le modèle utilisé pour les algorithmes dynamiques et les algorithmes de streaming.
Cette approche permet de capturer la quantité d'espace requise pour résoudre un problème en ligne, au travers d'un modèle simple.

Direction 1 : classification des automates probabilistes

Les automates probabilistes habitent toutes les classes de complexité spatiale en ligne déterministe [Fij16b]. Ce résultat est une étape vers une question plus ambitieuse, qui classifie les automates probabilistes selon leur complexité. En d'autres termes, on souhaite décrire la classe des automates probabilistes pour lesquels il existe un algorithme en ligne efficace de simulation. Cette description peut être de différente nature, par exemple structurelle, algébrique, ou au moyen d'un formalisme logique.

Décrire précisément la classe des automates probabilistes de faible complexité spatiale en ligne peut s'avérer être une question très difficile. Dans un premier temps, je commencerais par sous approximer cette classe, c'est-à-dire par décrire des conditions suffisantes pour qu'un automate probabiliste ait une faible complexité spatiale en ligne. Par exemple, restreindre l'ambiguité d'un automate probabiliste peut impliquer des résultats positifs sur sa capacité à être simulé en ligne.

Obtenir des résultats de classification de la complexité spatiale en ligne des automates probabilistes signifie décrire de manière générique des algorithmes de simulation pour des classes d'automates. Ces techniques sont fondamentales pour développer l'utilisation des automates probabilistes dans les domaines où intervient la notion d'algorithme en ligne.

Direction 2 : complexité spatiale en ligne alternante

L'étude de la complexité spatiale en ligne s'est jusque là focalisée sur les algorithmes déterministes. Pour l'étude des automates probabilistes, ce modèle d'algorithme n'est pas suffisant, puisque les automates probabilistes ont une complexité spatiale en ligne déterministe maximale. Afin d'obtenir des résultats positifs de simulation, je propose ici de considérer des algorithmes alternants. Un algorithme alternant peut effectuer plusieurs choix simultanément, et accepte ou rejette son entrée en considérant toutes les réponses obtenues. Les algorithmes alternants forment une classe d'algorithmes à la fois puissants, robustes et efficaces. Son appropriation dans de nombreux domaines a été encouragée par le développement d'outils très efficaces pour les analyser, s'appuyant sur la satisfaisabilité de formules logiques (SAT solver).

La définition de la complexité spatiale en ligne alternante s'obtient directement à partir de la notion connue dans le cas déterministe. Ceci couvre en particulier le cas des algorithmes non-déterministes. La question principale est la simulation en ligne d'un automate probabiliste au moyen d'un algorithme alternant : est-ce toujours possible, et quel est l'espace nécessaire pour cela ? Ceci implique deux types de résultats : d'une part, des résultats positifs, en construisant des algorithmes de simulation, et d'autre part, des résultats négatifs, montrant des bornes inférieures sur l'espace nécessaire pour simuler un automate probabiliste.

Les algorithmes alternants offrent un bon compromis entre un riche pouvoir d'expression et la possibilité de les implanter en pratique. Pour cette raison, développer la notion d'algorithme alternant en ligne est un défi important, avec des applications pratiques.

Objectif à long terme et intégration

L'étude de la complexité spatiale en ligne apporte une compréhension théorique sur l'utilisation de l'espace dans les algorithmes en ligne. Elle exhibe des familiarités entre problèmes et montre des limites intrinsèques pour leur résolution. À terme, son développement permettra de décrire des algorithmes génériques, structurant notre compréhension des algorithmes en ligne.
De tels algorithmes ont une portée théorique et pratique ; par exemple, les automates probabilistes sont utilisés pour l'étude des langues naturelles. La simulation en ligne permettrait d'effectuer du traitement de langues naturelles en temps réel, ce qui constitue un des principaux enjeux des décennies à venir.

Cet axe de recherche est un projet personnel, dont je suis pour l'instant l'unique investigateur. Le modèle a été proposé et étudié auparavant, et les questions que je propose sont motivées par divers enjeux contemporains, en écho à des travaux majeurs réalisés dans des cadres connexes. Bien que les questions soient nouvelles, les outils mobilisés me sont familiers, et sont étudiés dans les équipes SuMo à l'IRISA, Méthodes Formelles du LaBRI et MoVe au LIF.

Par ailleurs, la problématique des algorithmes en ligne apparaît en théorie du contrôle, étudié dans l'équipe SuMo, dans les travaux de Gabriele Puppis, Thomas Place et Anca Muscholl sur XML dans l'équipe Méthodes Formelles, ainsi que dans ceux de Pierre-Alain Reynier sur les transformations de transducteurs en flux (streamability). Je prévois d'étudier les liens entre la complexité spatiale en ligne et ces modèles et problématiques.

Réalisation du programme de recherche

Intégration dans les laboratoires

Mon projet de recherche s'inscrit dans les trois laboratoires suivants :

1. L'Institut de Recherche en Informatique et Systèmes Aléatoires (**IRISA**, Rennes), dans l'équipe SuMo, dirigée par Éric Fabre.

   Les trois directions de la première ligne directrice de mon projet, approximation, sont directement reliées aux travaux des membres de l'équipe SuMo. La direction 1, perturbations des transitions d'un automate probabiliste, rejoint les thématiques de robustesse et d'analyse de sensibilité d'Ocan Sankur. La direction 2, distances entre systèmes aléatoires, est influencée et motivée par les travaux récents d'Éric Fabre et Blaise Genest. La direction 3, approximer pour contourner le problème de Skolem, est déjà l'objet d'une collaboration avec Nathalie Bertrand, Blaise Genest et S. Akshay, commencée l'année dernière grâce au soutien de l'ANR STOCH-MC, dont je fais partie et dont Blaise Genest est responsable.

   Enfin, la théorie du contrôle, qui est un des thèmes développés dans l'équipe SuMo, entre en résonnance avec la seconde ligne directrice de mon projet, complexité spatiale en ligne, permettant de créer des ponts entre ces deux domaines.

2. Le Laboratoire Bordeaux de Recherche en Informatique (**LaBRI**, Bordeaux), dans l'équipe Méthodes Formelles, dirigée par Jérôme Leroux.

   Au sein de cette équipe, Hugo Gimbert se spécialise dans l'étude des systèmes aléatoires. J'ai commencé à collaborer avec lui avant même de commencer ma thèse, et la plupart de mes travaux sur les systèmes aléatoires sont issus de nos interactions ou de questions qu'il a posées. En particulier, la direction 1, perturbations des transitions d'un automate probabiliste, s'inscrit dans la continuité de travaux récents que nous avons réalisés ensemble. Les pistes de recherche qu'il étudie en ce moment avec son étudiant Edon Kelmendi partagent les mêmes motivations que celles décrites dans la première ligne directrice de mon projet, approximation.

   Le traitement de XML et les logiques sur les arbres, étudiés par Diego Figueira, Anca Muscholl et Gabriele Puppis, aborde des problématiques similaires à la seconde ligne directrice de mon projet, complexité spatiale en ligne. Ces connections permettent d'élargir le spectre d'applications de la complexité spatiale en ligne, et motivent de nouvelles questions.

   Enfin, Jérôme Leroux, Gabriele Puppis et Marc Zeitoun étudient des systèmes à compteurs avec des problématiques proches de celles que j'ai étudiées pendant ma thèse, ce qui pourrait amener à de futures collaborations.

3. Le Laboratoire d'Informatique Fondamentale (**LIF**, Marseille), dans l'équipe MOVE, dirigée par Pierre-Alain Reynier.

   L'étude des systèmes aléatoires est représentée dans cette équipe par Benjamin Monmege, avec qui je pourrais collaborer sur les trois directions de la première ligne de mon projet.

Pierre-Alain Reynier et Jean-Marc Talbot étudient des modèles de transducteurs, et la question de transformation en flux (streamability). Ils ont développé à ces fins des classes de complexité qui présentent des similarités avec celles étudiées dans la seconde ligne directrice de mon projet, complexité spatiale en ligne. Établir des connections précises entre ces deux questions ouvrent des perspectives intéressantes, reliant des modèles de natures différentes.

L'ordre donné ci-dessus est alphabétique et ne reflète pas mes préférences personnelles. Je m'intégrerais dans ces trois laboratoires avec le même enthousiasme.

Collaborations nationales et internationales

Au niveau national, en dehors des trois laboratoires cités ci-dessus, j'ai développé des collaborations fructueuses avec les chercheurs suivants :

— Youssouf Oualhadj de l'équipe Vérification au LACL (Créteil),
— Thomas Colcombet, Florian Horn, Jean-Éric Pin et Olivier Serre de l'équipe Automates de l'IRIF (Paris),
— Matteo Mio de l'équipe PLUME de l'ENS-LIP (Lyon).

Au niveau international, les problèmes que je me propose d'aborder sont également étudiés dans plusieurs équipes de recherche. Je mentionne ici quelques chercheurs et équipes avec lesquels j'ai travaillé par le passé, et prévois de collaborer dans le futur :

— avec l'équipe Automates de l'Université de Varsovie, dirigé par Damian Niwiński,
— avec l'équipe Vérification de l'Université d'Oxford, dirigée par Marta Kwiatkowska,
— avec Cristian Riveros de l'Université de Santiago du Chili,
— avec le groupe de Jean-François Raskin de l'Université libre de Bruxelles,
— avec le groupe de Krishnendu Chatterjee de l'Université de Vienne,
— avec le groupe de Christof Löding de RWTH Aachen,
— avec Martin Zimmermann de l'Université de Sarrebrücken.

Réalisations logicielles

L'essentiel de ce programme de recherche est orienté vers une compréhension théorique. La finalité de ces travaux étant la construction d'algorithmes, je prévois également d'implanter les algorithmes obtenus.
C'est le chemin que j'ai pris lors de l'étude des automates probabilistes et à compteurs pendant ma thèse ; j'ai implanté, en collaboration avec Denis Kuperberg, une partie des algorithmes que nous avons construits [FK14]. Ce développement logiciel nous a permis d'avancer dans notre compréhension des problèmes. Ceci a permis d'obtenir la première implantation efficace d'un algorithme calculant la hauteur d'étoile d'un langage régulier, un problème réputé irréalisable en pratique.
Des développements logiciels sont envisagés suivant l'évolution de mon programme de recherche.

Références bibliographiques personnelles (liste partielle)

[Fij16a] Nathanaël Fijalkow. "Characterisation of an Algebraic Algorithm for Probabilistic Automata". In: *STACS*. (to appear). 2016.

[Fij16b] Nathanaël Fijalkow. "Online Space Complexity of Probabilistic Automata". In: *LFCS*. 2016, pp. 106–116.

[FGHO14] Nathanaël Fijalkow, Hugo Gimbert, Florian Horn, and Youssouf Oualhadj. "Two Recursively Inseparable Problems for Probabilistic Automata". In: *MFCS*. 2014, pp. 267–278.

[FGKO15] Nathanaël Fijalkow, Hugo Gimbert, Edon Kelmendi, and Youssouf Oualhadj. "Deciding the value 1 Problem for Probabilistic Leaktight Automata". In: *Logical Methods in Computer Science* 11.1 (2015).

[FGO12] Nathanaël Fijalkow, Hugo Gimbert, and Youssouf Oualhadj. "Deciding the Value 1 Problem for Probabilistic Leaktight Automata". In: *LICS*. 2012, pp. 295–304.

[FKS16] Nathanaël Fijalkow, Stefan Kiefer, and Mahsa Shirmohammadi. "Trace Refinement in Labelled Markov Decision Processes". In: *FoSSaCS*. (to appear). 2016.

[FK14] Nathanaël Fijalkow and Denis Kuperberg. "ACME : Automata with Counters, Monoids and Equivalence". In: *ATVA*. 2014, pp. 163–167.

Références bibliographiques

[AAGT15] Manindra Agrawal, S. Akshay, Blaise Genest, and P. S. Thiagarajan. "Approximate Verification of the Symbolic Dynamics of Markov Chains". In: *J. ACM* 62.1 (2015).

[AAOW15] S. Akshay, Timos Antonopoulos, Joël Ouaknine, and James Worrell. "Reachability problems for Markov chains". In: *Inf. Process. Lett.* 115.2 (2015).

[AMS96] Noga Alon, Yossi Matias, and Mario Szegedy. "The Space Complexity of Approximating the Frequency Moments". In: *STOC*. 1996.

[BFS12] Nathalie Bertrand, John Fearnley, and Sven Schewe. "Bounded Satisfiability for PCTL". In: *CSL*. 2012.

[BC06] Mikołaj Bojańczyk and Thomas Colcombet. "Bounds in w-Regularity". In: *LICS*. 2006.

[CK14] Taolue Chen and Stefan Kiefer. "On the total variation distance of labelled Markov chains". In: *CSL-LICS*. 2014.

[Col11] Thomas Colcombet. "Green's Relations and Their Use in Automata Theory". In: *LATA*. 2011.

[Col09] Thomas Colcombet. "The Theory of Stabilisation Monoids and Regular Cost Functions". In: *ICALP*. 2009.

[CL89] Anne Condon and Richard J. Lipton. "On the Complexity of Space Bounded Interactive Proofs (Extended Abstract)". In: *FOCS*. 1989.

[FM85] Philippe Flajolet and G. Nigel Martin. "Probabilistic Counting Algorithms for Data Base Applications". In: *J. Comput. Syst. Sci.* 31.2 (1985).

[GGP10] Mai Gehrke, Serge Grigorieff, and Jean-Éric Pin. "A Topological Approach to Recognition". In: *ICALP*. 2010.

[GO10] Hugo Gimbert and Youssouf Oualhadj. "Probabilistic Automata on Finite Words : Decidable and Undecidable Problems". In: *ICALP*. 2010, pp. 527–538.

[HS06] Thomas A. Henzinger and Joseph Sifakis. "The Embedded Systems Design Challenge". In: *FM*. 2006.

[Kar67] Richard M. Karp. "Some Bounds on the Storage Requirements of Sequential Machines and Turing Machines". In: *J. ACM* 14.3 (1967).

[KNP07] M. Kwiatkowska, G. Norman, and D. Parker. "Stochastic Model Checking". In: *SFM*. Ed. by M. Bernardo and J. Hillston. Vol. 4486. LNCS (Tutorial Volume). Springer, 2007.

[MR95] Rajeev Motwani and Prabhakar Raghavan. *Randomized algorithms*. Cambridge University Press, 1995.

[MP80] J. Ian Munro and Mike Paterson. "Selection and Sorting with Limited Storage". In: *Theor. Comput. Sci.* 12 (1980).

[ORS14] Youssouf Oualhadj, Pierre-Alain Reynier, and Ocan Sankur. "Probabilistic Robust Timed Games". In: *CONCUR*. 2014.

[PI94] Sushant Patnaik and Neil Immerman. "Dyn-FO : A Parallel, Dynamic Complexity Class". In: *PODS*. 1994.

[Paz71] Azaria Paz. *Introduction to Probabilistic Automata*. Academic Press, 1971.

[Pin09] Jean-Éric Pin. "Profinite Methods in Automata Theory". In: *STACS*. 2009.

[Rab63] Michael O. Rabin. "Probabilistic Automata". In: *Information and Control* 6.3 (1963), pp. 230–245.

[San13] Ocan Sankur. "Robustness in Timed Automata : Analysis, Synthesis, Implementation". PhD thesis. Laboratoire Spécification et Vérification, ENS Cachan, France, 2013.

[Sim90] Imre Simon. "Factorization Forests of Finite Height". In: *Theor. Comput. Sci.* 72.1 (1990).