

ÉLÉMENT DE PORTFOLIO 02



Publication

1 DÉFINITION DE CET ÉLÉMENT

Titre de l'élément : *A quantitative study of fork-join processes with non-deterministic choice : application to the statistical exploration of the state-space.* Antoine Genitrini, Martin Pépin, Frédéric Peschanski. In Theoretical Computer Science, 2021, vol. 912.

URL de l'élément : <https://hal.science/hal-03201618/>

2 MOTIVATIONS DU CHOIX DE CET ÉLÉMENT

Nous avons choisi cet article car il représente un jalon important dans notre projet ParCo (Parallélisme & Combinatoire) dont l'objet concerne l'étude des systèmes concurrents sous l'angle de la combinatoire énumérative et/ou analytique. Pour la première fois, nous pouvons appliquer les outils quantitatifs et les algorithmes que nous avons développés pendant plusieurs années dans le cadre d'un langage concurrent assez expressif. L'article est d'ailleurs accompagné d'un artefact logiciel permettant de reproduire les expérimentations présentées.¹ Ce projet, en lui-même, est important pour l'équipe puisqu'il est un thème fédérateur entre des considérations liées à la programmation (concurrente) d'un côté, et aux études algorithmiques de l'autre. De plus, cet article est publié dans la revue TCS à fort impact, ce qui donne de la visibilité à notre projet. Cet article illustre également notre investissement dans la formation par la recherche, puisqu'il s'agit d'une des contributions importantes de la thèse de Martin Pépin, dirigée par Antoine Genitrini et co-encadrée par Frédéric Peschanski.

2.1 Contexte scientifique : le projet ParCo

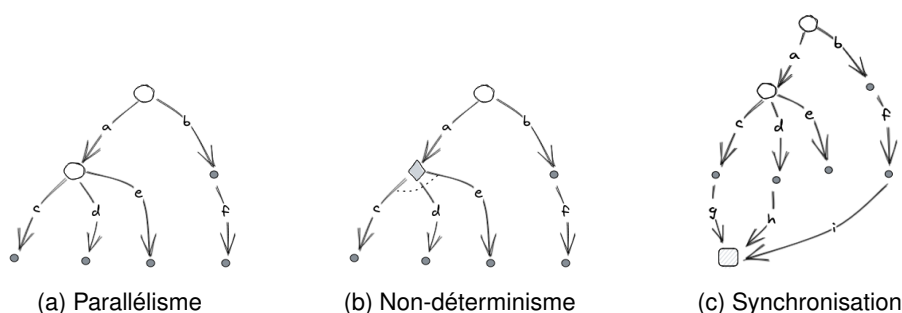


FIGURE 1 – Les principaux phénomènes de concurrence.

Dans le projet ParCo, nous étudions les programmes parallèles vus comme des objets combinatoires, dans une démarche originale de rapprochement entre la sémantique des langages de programmation et la combinatoire. Nous nous basons sur des interprétations combinatoires des principaux phénomènes de concurrence : parallélisme, non-déterminisme et synchronisation (cf. Figure 1). D'un point de vue quantitatif, nous étudions en particulier le phénomène d'explosion combinatoire, au cœur de la problématique de concurrence. D'un point de vue pratique, nos travaux ouvrent de nouvelles voies dans le domaine de la génération aléatoire uniforme : génération de structures et génération de chemins d'exécutions. Sur ces fondements algorithmiques, nous développons des prototypes logiciels dans les domaines de la génération automatique de tests ainsi que de la vérification statistique de modèles. Ce projet a conduit à la publication de nombreux articles tant dans le domaine de la concurrence [2, 5, 6] que de la combinatoire et l'analyse d'algorithmes [1, 3, 4].

1. cf. <https://gitlab.com/ParComb/libnfvj>

2.2 Présentation de l'article et de ses contributions

Le point de départ de l'article consiste en la présentation d'un modèle combinatoire permettant de combiner les interprétations que nous avons proposées dans nos précédents articles, en intégrant notamment :

- ▶ le parallélisme vu comme l'étiquetage croissant de structures arborescentes ;
- ▶ le non-déterminisme vu comme l'étiquetage croissant partiel de la structure ;
- ▶ la synchronisation de processus vue comme de l'étiquetage croissant sur-contraint.

En complément de cette interprétation combinatoire unifiée, l'article introduit également la possibilité de décrire des boucles dans les programmes, permettant les itérations. En terme d'expressivité, la proposition d'une interprétation combinatoire unique représente un grand pas en avant pour le projet ParCo, notamment pour ce qui concerne l'application en pratique de nos résultats.

Études quantitatives

Du point de vue analytique, la représentation de la sémantique de programmes concurrents sous forme de classes combinatoires nous permet d'étudier, en moyenne, des aspects quantitatifs concernant leur espace d'états. En guise d'exemple, l'article propose une étude quantitative précise du nombre moyen de chemins d'exécution induits par l'opérateur de choix non-déterministe. On peut noter, par exemple, que cet opérateur est bien moins « explosif » que l'opérateur de parallélisme, et qu'il peut donc être parfois intéressant de « déplier » les choix non-déterministes lorsque l'on désire effectuer des explorations statistiques de l'espace d'états. Un autre exemple d'étude quantitative proposée dans l'article concerne ce que l'on nomme « la forme typique » (*typical shape*) de l'espace d'états, étude basée sur l'estimation précise du nombre moyen de préfixes d'exécutions (plutôt que les exécutions complètes).

Algorithmique


D'un point de vue plus pratique, des algorithmes intéressants découlent naturellement de nos études quantitatives. Le premier algorithme présenté et expérimenté dans l'article concerne le comptage du nombre de préfixes d'exécution d'une longueur donnée. Bien que sans usage pratique immédiat, le comptage est la brique de base sur laquelle repose la plupart de nos autres algorithmes. Nous montrons dans [2] que si le modèle de synchronisation n'est pas contraint d'une façon ou d'une autre, le problème du comptage est difficile (techniquement, $\sharp P$ -complet). Le nombre intrinsèquement exponentiel de choix non-déterministes « locaux » représente une seconde difficulté pour le comptage. Dans l'article, nous adoptons un modèle de synchronisation de type *fork-join* qui, d'une part, simplifie le problème du comptage, et d'autre part reste suffisamment intéressant du point de vue de l'expressivité. De plus, nous proposons un encodage de l'espace d'états sous la forme de fonctions génératrices, permettant de définir une notion de *choix global* ne nécessitant pas le dépliage des choix locaux.

Analyses de programme

Nous exploitons notre brique algorithmique de comptage pour développer deux analyses complémentaires pour les systèmes concurrents. Tout d'abord, nous développons un générateur aléatoire uniforme permettant de générer, pour un programme spécifique, des chemins d'exécution (ou ordonnancements) admis par ce programme. La contrainte d'uniformité permet de garantir la meilleure couverture possible en l'absence de connaissance préalable sur la forme de l'espace d'états. Cela représente donc une bonne stratégie par défaut pour l'exploration statistique de cet espace d'états. Nous proposons dans l'article une seconde approche complémentaire qui consiste à générer, de façon uniforme, des *préfixes* de chemins d'exécution plutôt que des chemins complets. Arrivés à un certain point de programme, par la voie uniforme, il est alors possible d'introduire un biais dans l'exploration en sélectionnant telle ou telle branche, avant de progresser – toujours de manière uniforme – dans l'espace d'états. Il est ainsi par exemple possible de choisir explicitement des branches « moins probables » durant l'exploration, et de fournir ainsi un moyen de contrôler finement cette exploration. Ces stratégies d'exploration contrôlable de l'espace d'états ouvrent des perspectives intéressantes pour le développement d'outils d'analyse de programmes concurrents dans le domaine du test automatisé et de la vérification statistique de modèle (*Monte-Carlo model checking*).

3 RÉFÉRENCES BIBLIOGRAPHIQUES

- [1] Olivier Bodini, Matthieu Dien, Antoine Genitrini, and Frédéric Peschanski. The Ordered and Colored Products in Analytic Combinatorics : Application to the Quantitative Study of Synchronizations in Concurrent Processes. In *Analytic Algorithmics and Combinatorics (ANALCO17)*, pages 16 – 30, Barcelone, Spain, January 2017.

- 
- [2] Olivier Bodini, Matthieu Dien, Antoine Genitrini, and Frédéric Peschanski. The Combinatorics of Barrier Synchronization. In *International Conference on Applications and Theory of Petri Nets and Concurrency*, pages 386–405. Springer, 2019.
 - [3] Olivier Bodini, Matthieu Dien, Antoine Genitrini, and Frédéric Peschanski. Quantitative and Algorithmic aspects of Barrier Synchronization in Concurrency. *Discrete Mathematics and Theoretical Computer Science*, vol. 22 no. 3(3), 2021.
 - [4] Olivier Bodini, Antoine Genitrini, and Frédéric Peschanski. Enumeration and Random Generation of Concurrent Computations. In Nicolas Broutin and Luc Devroye, editors, *23rd International Meeting on Probabilistic, Combinatorial, and Asymptotic Methods in the Analysis of Algorithms (AofA'12)*, volume DMTCS Proceedings vol. AQ, 23rd Intern. Meeting on Probabilistic, Combinatorial, and Asymptotic Methods for the Analysis of Algorithms (AofA'12) of *DMTCS Proceedings*, pages 83–96, Montreal, Canada, June 2012. Discrete Mathematics and Theoretical Computer Science.
 - [5] Olivier Bodini, Antoine Genitrini, and Frédéric Peschanski. The Combinatorics of Non-determinism. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2013)*, volume 24 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 425–436, Guwahati, India, 2013.
 - [6] Matthieu Dien, Antoine Genitrini, and Frédéric Peschanski. A Combinatorial Study of Async/Await Processes. In *The 19th International Colloquium on Theoretical Aspects of Computing*, volume 13572 of *Lecture Notes in Computer Science*, pages 170–187, Tbilisi, Georgia, September 2022. Springer.