

Proposition de stage de Master 2

Étude des automates à deque

1. Contexte et présentation du sujet

Les automates à deque (deque machine) constituent un modèle de calcul inspiré à la fois des automates finis et des machines de Turing. Ils manipulent un mot stocké dans une *liste doublement chaînée* (ou *deque*), sur laquelle il est possible d'ajouter ou de retirer des symboles aux deux extrémités (en temps constant). Ce modèle prolonge les automates à pile ou à file simple, en offrant une symétrie entre les deux extrémités du mot traité.

L'étude des automates à deque s'inscrit dans la recherche sur les **modèles d'énumération à mémoire bornée**, qui visent à énumérer des langages en limitant strictement la mémoire auxiliaire et le temps entre l'énumération de deux mots successifs. Les résultats récents ont montré que les automates à deque sont capables d'énumérer efficacement un grand nombre de langages, y compris certains qui ne sont pas réguliers ou contextuels. Ils permettent en particulier de modéliser des processus d'énumération *à délai constant*, c'est-à-dire produisant chaque mot successif en un temps borné indépendamment de la taille du mot.

Ce cadre théorique se situe au croisement de plusieurs domaines :

- la **théorie des langages formels**, en particulier la **théorie des automates**, via la comparaison entre langages réguliers, contextuels et rékursifs ;
- la **complexité**, en particulier celle des problèmes d'énumération (**complexité énumérative**) où l'on cherche à conserver une mémoire constante (**complexité en mémoire**).
- la **combinatoire algorithmique**, à travers en particulier les codes de Gray ; en effet, les automates à deque ont été introduit dans [1] afin d'implémenter un code de Gray.

2. Objectifs du stage

De nombreuses questions restent ouvertes quant au **pouvoir expressif** et aux **limites computationnelles** des automates à deque. Les axes de recherche suivants constituent des pistes intéressantes pour des travaux de Master ou de doctorat.

- Existe-t-il un **langage régulier** qui soit énumérable en délai constant mais non énumérable par un automate à deque ? Une telle construction permettrait de mieux cerner la frontière entre la régularité et la capacité d'énumération.
- Peut-on **classer précisément** les langages non contextuels mais énumérables par un automate à deque ? Certaines familles, comme $\{a^n b^n \mid n \in \mathbb{N}\}$, sont déjà connues pour

être énumérables, mais les conditions générales restent floues.

- Quels sont les liens exacts entre les langages énumérables par automates à deque et ceux définis dans d'autres cadres non uniformes, tels que les automates *push-pop* étudiés dans [2] ?
- Quel sous-ensemble des **langages réguliers** peut-on énumérer par un automate à deque.
- Peut-on trouver des modèles de calcul équivalents aux automates à deque ? Par exemple en modifiant les opérations qu'on peut réaliser sur la deque, sans en modifier le pouvoir d'expression.

Ces questions s'inscrivent dans une démarche plus large de **classification des langages selon leur complexité d'énumération**. Elles ouvrent la voie à une compréhension plus fine des capacités computationnelles minimales nécessaires à la génération ou à la reconnaissance de structures combinatoires. Il est éventuellement possible de continuer sur une thèse dans cette thématique.

3. Compétences requises

Pour mener à bien ce stage, des compétences et surtout un goût pour la complexité et la théorie des langages sont nécessaires.

4. Encadrement

- Le stagiaire sera encadré par Victor Marsault (LIGM) et Yann Strozecki (UVSQ) et collaborera également avec Claire David (LIGM), Nadime Francis (LIGM).
- Le stage aura lieu au Laboratoire d'informatique Gaspard Monge, à l'université Gustave Eiffel (proche de Paris, station Noisy-Champs sur le RER A). Possibilité de travailler aussi au laboratoire DAVID de l'université de Versailles Saint-Quentin-en-Yvelines (UVSQ).
- Les encadrants travailleront avec le stagiaire plusieurs fois par semaine.

Références

- [1] Antoine Amarilli, Claire David, Madime Francis, Victor Marsault, Mikaël Monet, and Yann Strozecki. Gray codes with constant delay and constant auxiliary space. Preprint available at : http://victor.marsault.xyz/resources/articles/ConstantDelaySpaceGrayCode_v0.pdf, 2025.
- [2] Antoine Amarilli and Mikaël Monet. Enumerating regular languages with bounded delay. In *STACS*, 2023. Preprint available at : <https://arxiv.org/abs/2209.14878>.