Observing Linear Hashing : Optimality and Applications

Encadrement. Le projet sera pour un binôme d'élèves (une ***Option trinôme*** est posible, *cf.* point 3 dans **Description détaillée du travail attendu**) et encadré par Ky NGUYEN (`ky.nguyen@lip6.fr`).

Mots clés. Fonctions de hachage linéaires ; Signatures ; Implémentation

Contexte et Résumé. L'utilisation de fonctions de hachage est omniprésente dans la conception et l'analyse d'algorithmes randomisés [MU05, Sect. 5]. Ce projet adopte une approche d'implémentation pour examiner les familles de fonctions de hachage et leurs comportements dans un contexte bien connu : *lancer n balles uniformément et indépendamment dans n urnes*. Plus spécifiquement, l'objectif du projet est d'examiner les progrès récents sur la façon dont la famille de fonctions de hachage choisie impacte le *nombre maximum de balles dans une urne donnée*, également appelé la *charge maximale (max-load)*. En particulier, nous étudions le comportement de la famille la plus simple possible de fonctions de hachage : les matrices aléatoires dans un corps de caractéristique 2, *i.e.* $\mathbb{F}_2$:

- Soit $\ell := \log n$ et nous organisons nos n urnes en un espace vectoriel de dimension ℓ sur $\mathbb{F}_2$, *i.e.* l'espace vectoriel $\mathbb{F}_2^\ell$.
- L'ensemble univers à hacher est l'espace vectoriel $\mathbb{F}_2^u$.
- Notre famille de fonctions de hachage contient des applications linéaires $h : \mathbb{F}_2^u \to \mathbb{F}_2^\ell$.

Dans divers régimes de paramètres, cette famille peut sans doute être l'une des plus simples à implémenter. Par exemple, en prenant $u = O(\ell)$, une fonction de la famille peut simplement effectuer des XOR bit à bit sur $O(\log n)$ mots, et nécessite $O(\log^2 n)$ bits pour être décrite. Pourtant, malgré sa simplicité, des travaux récents révolutionnaires montrent que le hachage linéaire atteint une *charge maximale optimale*, voir par exemple [JKZ25].

Au-delà des aspects théoriques, le hachage linéaire semble trouver des applications dans d'autres domaines de l'informatique théorique, notamment en *cryptographie*. Tout commence avec le travail fondateur sur la *cryptographie incrémentale (incremental cryptography)* par Bellare, Goldreich et Goldwasser [BBG95]. Au cœur de la cryptographie incrémentale se trouve l'observation que les données hachées consistent en éléments *qui se diffèrent par de légères modifications*. Idéalement, les futurs hachages peuvent alors être effectués en un temps proportionnel à la quantité de modification apportée au *premier élément haché*. C'est effectivement le cas pour le hachage linéaire, ce qui ouvre la voie à des applications dans le contexte des *signatures multi-utilisateurs*, comme démontré dans un travail récent de Zhu et Tessaro [TZ23].

Description détaillée du travail attendu. Le travail attendu pour le projet comprend :

1. Une implémentation de familles de fonctions de hachage linéaires sur $\mathbb{F}_2$, accompagnée d'un profilage, pour démontrer la charge maximale moyenne optimale théoriquement prouvée pour mapper m balles vers n urnes (conformément aux théorèmes 9, 10, 11 de [JKZ25]).

2. Un prototype de divers schémas de signature basés sur des fonctions de hachage linéaires, généralisées heuristiquement sur des corps de caractéristique autre que 2, comme suite du point 1, suivant [TZ23, Sect. 4.1,5.2].
3. (***Option trinôme***) Comme extension des points de travail ci-dessus, on peut compléter ce profilage des propriétés du hachage linéaire en examinant ses comportements en termes de normes au-delà de la ℓ_1-norme [1]. Nous notons que le cas de la ℓ_1-norme reflète la distance statistique de la distribution de hachage, qui est caractérisée dans [JKZ25] et a été classiquement démontrée par le *Leftover Hash Lemma* dans [ILL89] :

   Soit $S \subset \mathbb{F}_2^u$ et $h : \mathbb{F}_2^u \to \mathbb{F}_2^\ell$ choisie dans une famille de fonctions de hachage universelles satisfaisant $\ell \leqslant \log_2(|S|) - 2 \cdot \log_2(1/\epsilon)$ pour un certain $\epsilon > 0$. Alors $(h, h(U_S))$ est ϵ-proche de la distribution uniforme, en ℓ_1-norme, où U_S dénote une variable aléatoire uniformément distribuée sur S.

   Un travail récent [DD22, Thm. 1.2 & 2.1] considère h d'une famille de fonctions de hachage *linéaires* et montre que sous une condition plus relacheée

$$\log_2(|S|) - \ell = O\left(\log_2\left(\frac{\log_2(|S|)}{\tau\delta}\right)\right) \ ,$$

   au moins une fraction $(1-\delta)$ des choix de h aura $H(U_S)$ étant $(\tau 2^{-\ell})$-proche de la distribution uniforme, en ℓ_∞-norme [2]. Ce point de travail demande un profilage de ce résultat, qui caractérise le hachage linéaire de manière plus forte (ℓ_∞-norme) sous une hypothèse plus faible (autorisant une plus grande *perte d'entropie*, *i.e.* $\log_2(|S|) - \ell$).

Pour le point 1, le profilage doit illustrer graphiquement comment la charge maximale est liée aux différents choix de paramètres (m, n). Pour le point 2, un prototype de schéma de multi-signature s'exécutant sur plusieurs processus sur la même machine est suffisant, et il n'est pas nécessaire de faire une démonstration sur réseau. Pour les deux points, l'implémentation peut être réalisée dans n'importe quel langage de programmation au choix. Pour ***Option trinôme***, le profilage doit illustrer graphiquement comment, pour $y \in \mathbb{F}_2^\ell \subset \mathbb{F}_2^u$, la quantité

$$\left| \Pr_{h:\mathbb{F}_2^u \to \mathbb{F}_2^\ell}[h(U_S) = y] - 2^{-\ell} \right|$$

est liée à la borne $\tau 2^{-\ell}$, vis à vis des différents choix de paramètres (u, ℓ, τ, δ) et l'ensemble S.

Prérequis. Familiarité avec la programmation et les fonctions de hachage ; aucune connaissance requise en cryptographie mais être à l'aise avec l'algèbre et les probabilités est un avantage.

1. Intuitivement, c'est proportionnel à la somme des valeurs absolues des coordonnées d'un résultat hachage.
2. Intuitivement, c'est le max des valeurs absolues des coordonnées d'un résultat hachage.

Références

[MU05] Michael Mitzenmacher and Eli Upfal : *Probability and Computing : Randomized Algorithms and Probabilistic Analysis.* Cambridge University Press, Cambridge, 2005. https://doi.org/10.1017/CBO9780511813603

[JKZ25] Michael Jaber, Vinayak M. Kumar, and David Zuckerman : *Linear Hashing Is Optimal.* STOC '25 : Proceedings of the 57th Annual ACM Symposium on Theory of Computing, *pp. 245 - 255.* https://doi.org/10.1145/3717823.3718208

[ILL89] Russell Impagliazzo, Leonid Levin, and Michael Luby : *Pseudo-random generation from one-way functions.* STOC '89 : Proceedings of the 21st annual ACM symposium on Theory of computing, *pp. 12-24.* https://dl.acm.org/doi/pdf/10.1145/73007.73009

[BBG95] Mihir Bellare, Oded Goldreich, and Shafi Goldwasser : *Incremental cryptography and application to virus protection.* STOC '95 : Proceedings of the 27th annual ACM symposium on Theory of computing, *pp. 45-56.* https://dl.acm.org/doi/pdf/10.1145/225058.225080

[DD22] Manik Dhar and Zeev Dvir : *Linear Hashing with ℓ_∞-guarantees and two-sided Kakeya bounds,.* FOCS '22 : IEEE 63rd Annual Symposium on Foundations of Computer Science, Denver, CO, USA, 2022, pp. *419-428* https://doi.org/10.1109/FOCS54457.2022.00047

[TZ23] Stefano Tessaro, and Chenzhi Zhu : *Threshold and Multi-Signature Schemes from Linear Hash Functions.* Advances in Cryptology – EUROCRYPT 2023. Lecture Notes in Computer Science, vol 14008. Springer, Cham. https://doi.org/10.1007/978-3-031-30589-4_22