Rapport EI DEMAILLE COURNIL RABEUX REBOLA CAHITTE

Matéo Cahitte, Clément Cournil-Rabeux, Melkior Demaille, Clément Rebola

January 2026

1 Introduction

1.1 Pourquoi un algorythme génétique ?

En jouant sur 32 paramètres (les deux premiers angles pour chaque dinucléotide), on a un espace grand pour les configurations, trop pour une résolution analytique (d'autant que le calcul devrait prendre en compte les positions de miliers de bases). De plus, il y a potentiellement de nombreux minimums locaux qu'une descente de gradient risque de ne pas éviter efficacement. Pour finir, on se doute que les trajectoires sont tout de même des fonctions lisses (même si très sensibles) des angles reportés dans les tables, un algorythme génétique est donc bien adapté pour que génération après génération, on se rapproche des extrémums les plus prometteurs.

1.2 Quelle fonction de fitness ?

Le sujet repose sur les plasmides. La compréhension que nous en avons est qu'il faut arriver pour chaque plasmide à une table qui donne la trajectoire la plus proche possible d'une chaîne qui boucle, dans la limite des écarts-types donnés par le modèle. Réalistiquement, aux échelles des tailles des plasmides, des effets des forces intermoléculaires ou de la température doivent courber d'identiques dinucléotides différemment, un effet dont on ne pourra pas ici prendre compte. De même on pourraît vouloir entraîner une seule table "universelle" pour toute séquence ADN, qui donne un bon repliment pour ces séquences (ce que le modèle fournit pour celles qui sont assez courtes). Cependant il faudrait pour cela disposer de séquences de référence dont on connaît les positions dans l'espace pour chaque nucléotide (à l'instar des protéines sur lesquelles AlphaFold s'entraîne). Ici la seule information donnée sur les entrées est que ce sont des plasmides. La seule chose que nous pouvons mesurer est donc à quel point ils "bouclent", sans autre prérequis envisageable sur la forme précise qu'ils occupent dans l'espace.

La première idée qui vient est donc de mesurer la distance entre la première et la dernière base dans l'espace, comme fonction de fitness : plus les extrémités sont proches, plus il est convainquant que la table corresponde à celle modélisant

au mieux le chemin du plasmide. En réfléchissant un peu plus, le plasmide devant boucler, ce qui nous intéresse est la différence entre la position de la première base de la séquence (l'origine), et la position où elle se retrouverait en parcourant la boucle jusqu'au bout. On peut même adjoindre les k premières bases à la fin de notre séquence, et minimiser la distance entre la position de chacune d'entre elle en début et en bout de chaîne. Le fait de mesurer la distance sur au moins deux bases permet en outre de tenir compte de la table dans le rapprochement des plasmides : boucler de façon compatible avec la table de rotation est plus souhaitable que de retrouver nos extrémités certes superposées mais avec un angle d'insidence quelconque - ceci s'apparente à une condition de continuité Ck.

Nous avons eu une idée pour vérifier plus loin la compatibilité d'une table avec un plasmide, mais qui pointait vers les limites du modèle : si la table permet en partant d'une représentation linéique particulière du plasmide de le voir boucler sur lui-même, nous n'avons pour l'instant aucune garantie que si le client considère plutôt un autre gène comme le début du plasmide la table fonctionne toujours. Autrement dit, il faudrait tester la qualité du bouclage du chemin obtenu avec un "début/fin" du plasmide différent, d'autres coupes de sa séquence que celles données par le fichier. Ainsi nous avons ajouté en paramètre de notre fonction le nombre de coupes autres que celles fournie qu'il faudra tester (réparties de façon homogène). Évidemment, ceci multiplie le temps d'exécution de la fonction de fitness, et la rend bien plus exigente : nous ne sommes même pas certains qu'il puisse exister de solution acceptable en général, avec le modèle fourni. Après expérimentation, nous avons convenu qu'il vallait mieux laisser ce paramètre de côté et se permettre des populations plus larges pour aboutir à des résultats convenables, même si dépendants du point de départ.

1.3 Quelle sellection ?

La première génération est distribuée uniformément sur les intervales autorisés par le modèle, pour chaque coordonnée. Ceci permet une maximation de l'entropie, évitant d'instaurer des biais de selection. Pour passer d'une génération à la suivante, nous avons essayé plusieurs méthodes de sélection, pour les comparer et déterminer celle qui convergeait le mieux, le plus vite. Nous avons voulu laisser en argument la proportion des individus gardés d'une génération à l'autre.

La sélection élitiste - sellectionnant les k individus au meilleur score parmi les n d'une génération - est un bon départ. Certes on abandonne une proportion de l'aléatoire qui fait l'efficacité des algorithmes génétiques, mais tant qu'une bonne façon de gérer l'aléatoire - bien adaptée à la sensibilité des trajectoires, et ne perdant pas son intérêt en considérant des mutations - n'a pas été trouvée, cette sélection reste un point de départ. C'est une bonne référence, un point de repère pour savoir si une autre méthode est intéressante ou non.

Pour adapté la sélection par tournoi à une proportion quelconque de survivants, nous avons choisi d'autoriser un individu à être tiré plusieurs fois. Ainsi ceux qui ont un bien meilleur score auront une meilleure espérence de

représentation. La motivation étant qu'un "outsider" aura toujours une chance d'être pris, et un très bon candidat aura moins de chance de se faire oublier.

Nous avons évidemment introduit la sélection roulette. Le problème de cette méthode est qu'elle est dépendante non-plus seulement de l'ordre des solutions, mais également de la valeur numérique de leur score. Ainsi, à priori, composer ce score avec toute fonction croissante nous donne une façon valide de faire tourner la roulette. Pour rester simples nous avons dans un premier temps directement utilisé le résultat de la fonction de fitness (distance euclidienne).

Pour raffiner la sélection roulette, nous nous sommes inspirés de la recuite simulée. En effet, on peut voir la pression évolutive comme un "ressort" qui doit pousser les extrémités du plasmide à se rejoindre, en ce sens un potentiel en x^2 (où x se trouve être la fonction de fitness) semble adapté. Pour le facteur de Boltzman il reste à fixer une température pour favoriser les bonnes solutions sans directement écraser celles qui s'en rapprochent. En s'inspirant du recuit, on peut vouloir diminuer la température avec les générations - tâchant de ne pas être trop brusque pour ne pas tomber directement sur un extrémum local trop pauvre - pour affiner la convergence.

Une autre forme de roulette est la sélection roulette-rang. Ici on oublie même la valeur de la fonction de fitness pour ne retenir que l'ordre qu'elle donne - on a cependant toujours la même problématique (composition avec une fonction croissante). On tire alors avec une probabilité proportionnelle au complémentaire du rang (rang 0 -¿ poids n).

Pour une variante de la sélection roulette-rang, on utilise une distribution géométrique des probabilités. Avec comme raison une pression de sélection 0¡q¡1, on ajuste le rapport des probabilités de choix du k-ième sur le k+1-ième. Ceci permet une répartition un peu plus contrôlée des "outsiders". Il semble cependant toujours dommage d'"oublier" une part de l'information donnée par fitness, mais de moins guider l'aléatoire peut le rendre plus puissant.

1.4 Quelle reproduction ?

La question qui se pose ici est de savoir si les individus mieux classés doivent "plus" se reproduire que les moins classés, y compris l'étape de sélection passée. En effet, tester plusieurs variations du meilleur candidat d'une génération à l'autre peut être intéressant. Nous pouvions jouer sur plusieurs paramètres : le nombre d'occasions qu'a un survivant de se reproduir, et le poids de son impact sur les gènes de ses descendants. Le premier paramètre étant en réalité déjà pris en compte dans certaines des fonctions de sélection, nous avons préféré nous concentrer sur le second : un descendant portera des gènes plus proches de son parents ayant reçu le meilleur score.

1.5 Quelle mutation ?

Savoir à quel rythme et l'amplitude des mutations est primordial. Des mutations trop fréquentes et on perd l'intérêt d'avoir un héritage. Trop faibles et on retrouve les écueils de la descente de gradient - surtout si les génomes sont

mélangés à chaque nouvel individu, avec une tendance à l'uniformisation. Nous avons choisi de répartir les mutations autour des valeurs initiales avec une loi normale (coupée pour éviter de dépasser les bornes du modèle) pour les amplitudes, et une probabilité égale pour chaque gène de muter. De plus, nous avons joué sur la largeur de la normale et la probabilité des mutations au fil des générations pour affiner les solutions, une fois un bon candidat trouvé.