

Étude de l'émergence de nouveaux algorithmes de recherche locale par neuro-évolution

Olivier Goudet, Mohamed Salim Amri Sakhri, Adrien Goëffon et Frédéric Saubion

LERIA, Université d'Angers - 2 Boulevard Lavoisier, Angers, 49045, France

{olivier.goudet,mohamedsalim.amrisakhri,adrien.goëffon,frederic.saubion}@univ-angers.fr

Mots-clés : *Neuro-évolution, Recherche locale, Optimisation boîte noire, Modèle NK*

1 Introduction

Comment faire émerger de nouveaux opérateurs de recherche locale bien adaptés à un problème combinatoire particulier ? Certains travaux proposent déjà une génération automatique d'opérateurs à partir de constituants de base d'heuristiques [1], ce qui limite cependant la possibilité de produire des opérateurs complètement nouveaux. La spécificité de ce travail, est de faire émerger de nouveaux algorithmes de recherche locale grâce aux techniques de neuro-évolution [3] et d'apprentissage par renforcement, sans présupposer l'existence de constituants prédéfinis, mais en faisant "jouer" et évoluer ces "agents de recherche locale" dans un monde ouvert constitué du paysage de fitness du problème à résoudre, à la manière de ce qui a été récemment fait pour l'étude de l'émergence de nouveaux moyens de locomotions par des intelligences artificielles dans des environnements complexes [4].

2 Apprentissage d'algorithmes de recherche locale pour résoudre des problèmes pseudo-booléens

Dans ce travail, nous considérons exclusivement des problèmes pseudo-booléens. Dans ce cas, l'espace de recherche est défini par $\mathcal{X} = \{0, 1\}^N$. La qualité de ces solutions est évaluée à l'aide d'une fonction objectif $f_{obj} : \mathcal{X} \rightarrow \mathbb{R}$. La relation de voisinage est définie par une fonction d'inversion de bit, notée $flip_i : \mathcal{X} \rightarrow \mathcal{X}$, $i \in \llbracket 1, N \rrbracket$ telle que $flip_i(x)$ est égal à x sauf sa $i^{\text{ème}}$ composante qui est inversée (passage de 0 à 1 et vice versa). Étant donné une solution $x \in \mathcal{X}$, on note ainsi $\mathcal{N}(x) = \{flip_i(x) \mid i \in \llbracket 1, N \rrbracket\}$ l'ensemble de ses voisins dans l'espace de recherche. On introduit aussi une fonction d'observation d'un état à valeur dans l'espace des observations Ω , correspondant dans notre cas aux variations de la fonction objectif f_{obj} pour chaque inversion possible d'un bit de x : $o(x) = (f_{obj}(flip_1(x)) - f(x), \dots, f_{obj}(flip_N(x)) - f(x)) \in \mathbb{R}^N$.

Nous utilisons le formalisme de l'apprentissage par renforcement pour définir une stratégie de recherche locale sous la forme d'une politique $\pi : \Omega \rightarrow \mathcal{A}$ avec $\mathcal{A} = \{flip_i \mid i \in \llbracket 1, N \rrbracket\}$, un ensemble d'actions. Une telle politique peut être paramétrée par un vecteur de paramètres θ . L'ensemble de toutes les politiques est $\Pi = \{\pi_\theta \mid \theta \in \Theta\}$ où Θ est l'espace des paramètres.

Étant donné une instance $I = (\mathcal{X}, f_{obj})$, une solution initiale $x_0 \in \mathcal{X}$, une politique π_θ , et un horizon H , une trajectoire $T(\pi_\theta, I, x_0, H)$ est une séquence $x_0, a_0, x_1, a_1, \dots, x_{H-1}, a_{H-1}, x_H$, telle que $\forall i \in \llbracket 0, H-1 \rrbracket, a_i = \pi_\theta(o(x_i))$ et $\forall i \in \llbracket 1, H \rrbracket, x_i = \delta(x_{i-1}, a_{i-1})$, avec $\delta : \mathcal{A} \times \mathcal{X} \rightarrow \mathcal{X}$ la fonction de transition de l'environnement.

Contrairement au cadre d'apprentissage classique de l'apprentissage par renforcement, où une récompense peut être attribuée après chaque action, dans notre contexte, la récompense est calculée globalement pour chaque trajectoire donnée : $R(\pi_\theta, I, x_0, H) = \max_{x \in T(\pi_\theta, I, x_0, H)} f_{obj}(x)$.

Dans ce contexte, trouver un algorithme efficace de recherche locale pour résoudre un ensemble d'instances revient à construire une politique optimale.

3 Apprentissage par neuro-évolution d'une stratégie de recherche locale

Nous proposons d'utiliser un réseau de neurones g_θ , paramétré par un vecteur de nombres réels θ , pour définir une nouvelle politique de recherche locale déterministe $\pi_\theta : \mathbb{R}^N \rightarrow \mathcal{A}$, appelée Neuro-LS. Ce réseau de neurones prend en entrée le vecteur d'observation $o(x)$ défini plus haut et produit en sortie un vecteur $g_\theta(o)$ de taille N dont la composante $g_\theta(o)_i$ correspond à un score de préférence associé à chaque observation o_i . Ensuite, l'action a_i correspondant au score le plus élevé $g_\theta(o)_i$ est sélectionnée. g_θ présente deux propriétés importantes qui le rendent consistant pour résoudre tout type d'instance : (1) il est équivariant relativement à des permutations du vecteur des observations qu'il reçoit en entrée, et (2) son nombre de paramètres ne dépend pas de la taille N du problème considéré.

Suivant le principe de la neuro-évolution, l'ensemble de paramètres θ de g_θ est optimisé afin de maximiser les récompenses $R(\pi_\theta, I, x_0, H)$ estimées sur des ensembles d'instances. Il s'agit d'un problème d'optimisation stochastique boîte noire dans un espace de recherche à valeurs réelles $\mathbb{R}^{|\theta|}$ et nous utilisons la stratégie évolutionnaire à estimation de distribution CMA-ES [2] pour le résoudre.

4 Expérimentations

Pour évaluer l'efficacité de notre approche, nous définissons un environnement de *jeu* basé sur des problèmes pseudo-bouliens de type modèle NK, qui nous permettent de décrire un paysage de fitness dont la taille du problème et la rugosité (déterminant le nombre de minima locaux) peuvent être ajustées facilement.

Les résultats montrent que différentes politiques émergent grâce à la neuro-évolution, chacune étant adaptée à la résolution d'un type particulier de distribution d'instances du modèle NK (paysages lisses ou rugueux). Notre algorithme est compétitif par rapport aux procédures de recherche locale déterministe de base pour tous les types de modèle NK considérés dans ce travail. En particulier pour les paysages rugueux, il peut obtenir des résultats nettement meilleurs avec une stratégie émergente originale, qui met en œuvre notamment ponctuellement un choix d'action qui dégrade beaucoup le score sur le court terme, mais qui s'avère très efficace à long terme pour échapper aux pièges des optima locaux.

Références

- [1] Jorge M Cruz-Duarte, José C Ortiz-Bayliss, Ivan Amaya, and Nelishia Pillay. Global optimisation through hyper-heuristics : Unfolding population-based metaheuristics. *Applied Sciences*, 11(12) :5620, 2021.
- [2] Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2) :159–195, 2001.
- [3] Matthew Hausknecht, Joel Lehman, Risto Miikkulainen, and Peter Stone. A neuroevolution approach to general atari game playing. *IEEE Transactions on Computational Intelligence and AI in Games*, 6(4) :355–366, 2014.
- [4] Nicolas Heess, Dhruva TB, Srinivasan Sriram, Jay Lemmon, Josh Merel, Greg Wayne, Yuval Tassa, Tom Erez, Ziyu Wang, SM Eslami, et al. Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv :1707.02286*, 2017.