

Introduction

L'algorithme présenté vise à résoudre le problème d'attribution de candidats à des ateliers, en utilisant une technique d'optimisation appelée Recuit Simulé. L'objectif est de trouver la meilleure répartition des candidats tout en maximisant une fonction objectif.

Principe de l'algorithme

Initialisation: L'algorithme démarre en générant aléatoirement un état initial représentant une répartition des candidats dans les ateliers.

Calcul de l'énergie: Une fonction d'énergie est définie pour évaluer la qualité de chaque état. Dans ce contexte, l'énergie est calculée comme l'opposé du score de l'état, qui mesure la performance de la répartition des candidats.

Boucle principale:

L'algorithme itère à travers les étapes suivantes jusqu'à un critère d'arrêt est satisfait:

Générer un état voisin aléatoire à partir de l'état courant.

Calculer l'énergie de cet état voisin.

Si l'énergie du voisin est meilleure que celle de l'état courant, ou selon une probabilité déterminée par la différence d'énergie et une fonction de température, l'algorithme accepte ce voisin comme nouvel état courant.

Mettre à jour l'état optimal si nécessaire.

Critère d'arrêt:

L'algorithme s'arrête lorsqu'un nombre maximum d'étapes est atteint ou lorsque l'énergie de l'état courant dépasse un certain seuil.

Paramètres clés de l'algorithme

Nombre d'étapes: Détermine la durée de l'exécution de l'algorithme.

Température: Paramètre crucial dans le calcul de la probabilité d'acceptation d'un état voisin moins favorable.

Seuil d'énergie maximale (eMax): Limite supérieure pour l'énergie de l'état courant.

Probabilité d'acceptation: Contrôle la probabilité d'acceptation d'un voisin moins favorable, favorisant l'exploration de l'espace des solutions.

Conclusion

L'algorithme de Recuit Simulé offre une approche flexible et efficace pour résoudre des problèmes d'optimisation, tels que l'attribution de candidats à des ateliers. En ajustant les paramètres et les stratégies de refroidissement, il est possible d'obtenir des solutions de haute qualité dans un temps raisonnable.

Notre programme convertit du JSON en classes java afin de traiter les données grâce à cet algo, avant de les retourner en JSON également.

