

a) La taille de l'espace de recherche est de 4^{88} car nous avons 88 règles libres en plus des règles fixes.

La taille du voisinage est de $85 * 3$ soit 255.

b) Concernant la métaheuristique Hill Climber First Improvement, j'ai décidé d'utiliser 2 paramètres : un nombre de répétitions (maxOpti) ainsi que l'automate (automata)

Je commence par initialiser l'espace de recherche ainsi que les règles.

Dans ma boucle « while » qui s'exécute selon la condition, je génère une règle aléatoire et compare si la fitness générée aléatoirement est meilleure que l'actuelle. Si c'est le cas on la garde sinon, on relance la boucle.

Remarque personnelle : pour avoir une meilleure fitness, il aurait fallu que j'enregistre les règles déjà rencontrées afin de gagner en vitesse de traitement.

c) Pour l'algo ILS, il s'agit d'une succession de HCFI, les paramètres sont les mêmes que pour la question précédente.

Après un HCFI sur une première règle, dans la boucle « while », on alterne 3 règles. Ensuite, on compare le résultat avec une seconde règle. Si la seconde est meilleure que la première, la première récupère cette valeur, sinon on répète l'opération. Et cela tant que le maximum de répétitions n'est pas atteint.

e) Pour le troisième algo, j'ai choisi de faire une random search.

C'est à dire que pour chaque itération, que le voisin soit meilleur ou non, la fitness gardée est celle du voisin.

Remarque personnelle : Cet algo ne nous donne pas forcément une fitness supérieure à la précédente.