

UNIVERSITÉ DE SHERBROOKE
Département d'informatique

IFT 615
Intelligence artificielle

Quiz 2
Hiver 2024

Publié le 4 avril 2024

À remettre le 11 avril à minuit par « turn-in Quiz2 »

PROFESSEUR

Froduald Kabanza

AUXILIAIRE D'ENSEIGNEMENT POUR LA CORRECTION

D'Jeff Kanda Nkashama

djeff.nkashama.kanda@usherbrooke.ca

INSTRUCTIONS

Ce quiz a 4 pages avec deux questions évaluées sur 6 points mais comptant pour 3 points de la note totale du cours.

Le quiz est individuel. Il ne peut être effectué en groupe.

Écrivez votre nom, prénom et matricule ci-dessous. Vous devez inscrire aussi votre nom et prénom en haut de chacune des autres pages à l'endroit indiqué.

NOM : _____

PRÉNOM : _____

MATRICULE : _____

Question 1 (3 points) – Recherche locale -- connaissances générales

Encerchez les affirmations correctes. Les réponses à cette question seront évaluées en faisant une somme des points obtenus sur les affirmations encadrées correctement et en soustrayant les points obtenus sur les affirmations encadrées fautivement. Une différence négative sera ramenée à 0. Une différence dépassant 3 sera ramenée à 3. Il n'est pas nécessaire de savoir combien de points affirmation correcte vaut ni le nombre d'affirmations correctes qu'il y a en tout.

1. Un algorithme génétique ne garantit pas de trouver une solution optimale.
2. Lorsque l'algorithme *recuit-simulé* (*simulated annealing*) est utilisé avec une température constante et positive à chaque itération, alors il a le même comportement que l'algorithme *Hill-Climbing*.
3. L'algorithme *Hill-Climbing* garantit de trouver une solution optimale lorsque l'heuristique utilisée est admissible – c.-à-d., lorsque l'heuristique utilisée ne surestime pas le coût du chemin vers l'état optimal.
4. L'algorithme *beam-search* n'est jamais utilisé en pratique. Il sert uniquement d'introduction aux algorithmes génétiques.
5. Un algorithme génétique pourrait être utilisé pour optimiser les hyperparamètres d'un réseau de neurones.
6. La complexité asymptotique de l'algorithme génétique est polynomiale.
7. La complexité asymptotique de *beam-search stochastique* est polynomiale.
8. Un algorithme génétique est un cas particulier de *stochastic beam search* où les états successeurs sont générés en combinant des parents plutôt que de modifier un seul état.

Question 2 (3 points) — Recherche locale par la satisfaction de contraintes

Supposons que vous devez préparer un souper composé des mets suivants : Soupe (S), Veau (V), Riz (R) et Haricots (H). Les mets S et V prennent chacun 2 heures à cuisiner, alors que les mets R et H prennent chacun 1 heure à cuisiner. La durée totale pour préparer tout le souper ne doit pas dépasser 4 heures. La préparation de chaque mets peut commencer seulement au début de l'heure, c.-à-d., à 1 heure, 2 heures, 3 heures et 4 heures. Les mets peuvent être cuisinés en parallèle, excepté pour S et R qui ne peuvent être cuisinés simultanément. De plus, la préparation de S doit être terminée avant celle de H , alors que celle de R doit être terminée avant celle de V . Le problème posé est celui de faire une cédule respectant les contraintes pour la préparation des quatre mets S , R , V et H .

Ce problème peut être formulé comme un problème de satisfaction de contraintes (**CSP**) dont les variables correspondent aux débuts de préparation de chaque mets. Notons ces variables **DS** (début de préparation de la soupe), **DR** (début du riz), **DV** (début du veau) et **DH** (début des haricots). Ainsi, le domaine des valeurs pour chacune de ces variables indique le début de préparation possible pour le mets correspondant, comme suit : $DS=DV=\{1, 2, 3\}$ et $DR=DH=\{1, 2, 3, 4\}$.

- a. (1 point) Sachant que nous avons affaire à des contraintes binaires, dessinez le graphe de contraintes pour ce problème. Mettez les contraintes de façon mathématique et précise et non des phrases en français.

- b. (1 point) Supposons que l'on assigne la valeur **2** à la variable **DS**, comme première étape de l'algorithme *backtracking-search* et qu'on applique ensuite l'algorithme *forward-checking*. Indiquez le domaine de valeurs de chaque variable à la fin de *forward-checking*. Pas besoin de montrer les itérations.

- c. (1 point) Supposons encore une fois que l'on assigne d'abord **2** à la variable **DS**, comme première étape de *backtracking-search* et qu'on applique ensuite *AC-3*. Indiquez le domaine de valeurs de chaque variable à la fin d'*AC-3*. Pas besoin de montrer les itérations.

FIN DU QUIZ