

Intelligence Artificielle - Examen - session n° 1

Date : 09/02/2023**Durée** : 2h30**Documents autorisés** : 2 feuilles (4 pages manuscrites)**Barème** : 3 + 5 + 6 + 6

Les exercices sont indépendants. Vous pouvez les traiter dans l'ordre de votre choix.

Toute réponse doit être **justifiée**, sinon elle ne sera pas prise en compte.

Exercice 1 (Questions de cours)

Répondre en quelques lignes (5 maxi) aux questions suivantes :

1. Un CSP qui est 2-cohérent est-il nécessairement 3-cohérent ?
Un CSP qui est 3-cohérent est-il nécessairement 2-cohérent ?
Prouvez le ou donnez un contre exemple.
2. La cohérence de borne implique-t-elle la cohérence d'arc ? Justifiez votre réponse.
3. Expliquez pourquoi, lors de la résolution d'un CSP par un algorithme de recherche, choisir d'affecter en priorité la variable la plus contrainte avec la valeur la moins contrainte est une bonne heuristique.

Exercice 2 (Formalisation de problème)

Alain vient de terminer ses études et son premier poste est au service commercial d'une petite entreprise qui vend et installe des alarmes pour des maisons individuelles et locaux professionnels. On lui a demandé de préparer une brochure présentant 4 modèles de centrales d'alarmes que commercialise cette société. Elles ont toutes été commercialisées à des dates différentes, ont des niveaux sonores différents et des tarifs différents. Alain a glané des informations ici et là auprès de ses collègues mais comme il est nouveau, les choses sont encore un peu confuses pour lui.

Il a retenu que :

1. Le modèle SmartGuard a un niveau sonore inférieur de 5 décibels à celui du modèle commercialisé en 2019.
2. Le modèle le plus cher a été commercialisé avant 2019 et avant le modèle SmartGuard
3. Aucun des modèles dont le niveau sonore est à 100 décibels ou 105 décibels ne correspond à celui commercialisé à 440€.
4. L'AlertSecure 3 sonne plus fort que l'alarme qui coûte 789€ .
5. Le modèle à 645€ est 5 décibels au-dessus de celui commercialisé en 2017
6. Aucun des modèles entre celui dont le niveau est 115 décibels et celui commercialisé en 2015 ne correspond au CerbereT2.
7. L'alarme qui coûte 360€ sonne plus fort que le modèle commercialisé en 2021.
8. Entre le modèle EcoW-R15 and et celui dont le niveau est à 110 décibels, un des modèles a été commercialisé en 2019 et l'autre coûte 360€.

On souhaite aider Alain à retrouver les caractéristiques de chaque alarme pour qu'il puisse réaliser sa brochure.

1. Formaliser l'énoncé précédent sous forme d'un problème de satisfaction de contraintes. Vous prendrez soin de **de bien justifier vos choix pour les variables et domaines** et modéliser toutes les contraintes mentionnées dans l'énoncé.
Conseil : bien lire l'énoncé dans son intégralité afin de déterminer quelle est la modélisation la plus judicieuse. *NB* : on ne demande pas la solution à ce stade
2. Dessiner l'hyper-graphe de contraintes correspondant à votre modèle (faites attention à l'arité de vos contraintes).

Exercice 3 (Algorithmes Hybrides)

On considère le CSP $\mathcal{P} = (X, D, C)$ caractérisé par :

- $X = \{A, B, C, D, E, F\}$
- $D = \{d, d, d, d, d, d\}$, avec $d = 1..5$
- $C = \{ F \geq 2, E \neq D, D < C, C - 2 \leq B, A > B + 1, |A - E| \leq 3, F = B + 2, E \neq F, D \bmod 2 = 1, F \neq D, E \leq 3, D > B \}$

1. Dessinez le graphe de contraintes correspondant à \mathcal{P}
2. Construire l'arbre correspondant à l'exécution de l'algorithme de recherche avec backtrack (BT) pour trouver les solutions de ce problème mais uniquement dans le prolongement de la branche correspondant à l'affectation partielle $A = 4, B = 2$. Vous affecterez les variables restantes dans l'ordre F, D, C, E .
3. On souhaite à présent comparer le résultat précédent avec une méthode hybride. Après avoir traité les contraintes unaires de façon adéquate, appliquez l'algorithme de recherche hybride du *forward checking* (FC) pour trouver **toutes** les solutions de ce problème. Vous affecterez les variables dans l'ordre D, B, F, C, A, E

Exercice 4 (OPL - Gappy)

Le jeu Gappy consiste à noircir certaines cases d'une grille de taille $n * n$, de façon à respecter certaines propriétés. Chaque ligne et chaque colonne doit contenir exactement deux cases noires. Deux cases noires ne peuvent être voisines, i.e. ne peuvent se toucher, latéralement ou diagonalement. Pour chaque instance une série d'indices est communiqué sous la forme d'entiers figurant à gauche de chaque ligne (resp. au sommet de chaque colonne). Le nombre indiqué caractérise le nombre de cases blanches séparant les deux cases noires sur cette ligne (resp. colonne).

Par exemple, une solution possible pour l'instance décrite figure 1a correspond à la figure 1b.

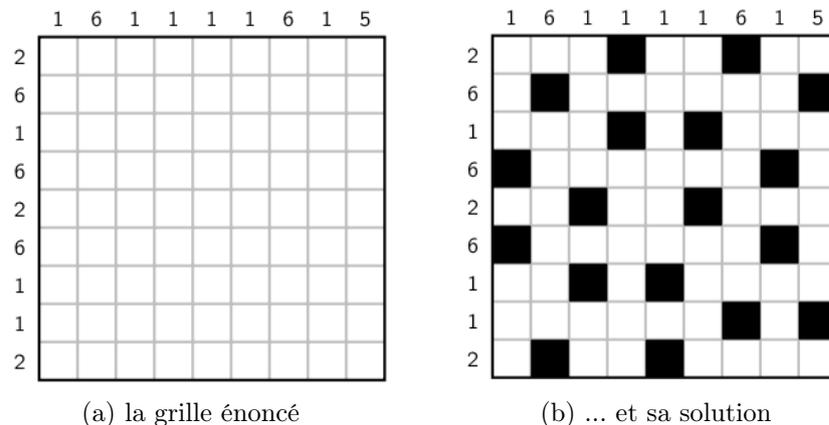


FIGURE 1 – Une instance du problème Gappy sur une grille 9×9 et sa solution

On souhaite écrire un modèle générique OPL permettant de résoudre n'importe quelle instance pour ce jeu.

1. Proposer un moyen de décrire dans le langage OPL les données caractérisant une instance d'un problème Gappy (l'illustrer avec le code décrivant l'instance de la figure 1a).
2. Écrire le code OPL permettant de modéliser ce problème (sauf les contraintes, traitées à la question suivante). Vous détaillerez la façon dont les données sont représentées (de façon externalisées), les pré-traitements éventuels, les variables de décision à déclarer et l'affichage de la solution.
3. Décrire le code modélisant les contraintes à respecter pour trouver une solution pour n'importe quelle instance. Bien expliquer la façon dont chaque contrainte de l'énoncé est modélisée.