



Un cours « électronique » sur la Programmation par Contraintes

JFPLC 2003

Christine Solnon

LIRIS, Université Lyon 1



Contexte : la e-miage

La e-miage : une version « à distance » de la filière MIAGE

- ▶ Pour qui ?
 - ▶ étudiants « à distance » (handicapés, à l'étranger)
 - ▶ professionnels en formation continue
 - ▶ ... accessible aux étudiants « en formation initiale »
- ▶ Comment ?
 - ▶ depuis un ordinateur connecté à Internet
 - ▶ connexion sur une « plateforme d'apprentissage »
 - ▶ suivi pédagogique par un « tuteur »
- ▶ Par qui ?
 - ▶ les enseignants des IUP-MIAGEs
 - ▶ ... pas forcément spécialistes en EIAH !!!



Contexte : la e-miage

Programme pédagogique national des IUP-MIAGEs

- ▶ 50 modules de 40 heures
 - ... dont un module intitulé « Intelligence Artificielle »
- ▶ Responsable du module IA de la e-miage : A. Mille
- ▶ Découpage du module IA en 45 séances de 45 mn
 - ▶ 2 séances d'introduction à l'intelligence artificielle
 - ▶ 22 séances sur la logique et la prog. logique
 - ▶ 5 séances sur la représentation des connaissances
 - ▶ 2 séances sur la résolution de problèmes
 - ▶ 5 séances sur la programmation par contraintes
 - ▶ 4 séances sur l'apprentissage
 - ▶ 5 séances sur les systèmes experts



Objectifs et choix pédagogiques

- ▶ Les objectifs pédagogiques : savoir...
 - ▶ ...ce qu'est un CSP
 - ▶ ...modéliser un pb sous la forme d'un CSP
 - ▶ ...comment fonctionne un solveur de contraintes
 - ▶ ...utiliser un langage de PPC pour résoudre un CSP
- ▶ Choix du langage : Gnu-Prolog
 - ▶ Continuité avec le cours de Prolog
 - ▶ Gratuit et facile à installer sur tous systèmes/machines
 - ▶ Manuel utilisateur en ligne



Organisation du cours

▶ 5 séances de 45 mn :

1. Contraintes et CSPs
2. Algorithmes pour la résolution de CSPs
3. Réalisation de solveurs de contraintes en Prolog
4. Programmation par contraintes avec Gnu-Prolog
5. Exercices de programmation par contraintes

► 5 séances de 45 mn :

1. Contraintes et CSPs
2. Algorithmes pour la résolution de CSPs
3. Réalisation de solveurs de contraintes en Prolog
4. Programmation par contraintes avec Gnu-Prolog
5. Exercices de programmation par contraintes

► Séance 1 « Contraintes et CSPs »

- Qu'est-ce qu'une contrainte ?
- Qu'est ce qu'un CSP ?
- Un premier exemple : le problème des reines
 - ~> Différentes modélisations / discussion
- Un deuxième exemple : les mariages stables
- Exercices de modélisation de problèmes sous la forme de CSPs
 - ~> Retour de monnaie d'un distributeur automatique
 - ~> Sel et moutarde (Lewis Carroll)
 - ~> Coloriage d'une carte
 - ~> Crypto « SEND + MORE = MONEY »
 - ~> Le zèbre (Lewis Carroll)



Organisation du cours

► 5 séances de 45 mn :

1. Contraintes et CSPs
2. Algorithmes pour la résolution de CSPs
3. Réalisation de solveurs de contraintes en Prolog
4. Programmation par contraintes avec Gnu-Prolog
5. Exercices de programmation par contraintes

► Séance 2 « Algos pour la résolution de CSPs »

- Restriction aux algos complets pour les CSPs sur les domaines finis
- L'algorithme « génère et teste »
 - ↪ Critique et notion d'espace de recherche d'un CSP
- L'algorithme « Simple retour-arrière »
- L'algorithme « Anticipation »
 - ↪ Introduction des notions de consistance locale et filtrage
- Intégration d'heuristiques
 - ↪ Ordres sur les variables et les valeurs

► 5 séances de 45 mn :

1. Contraintes et CSPs
2. Algorithmes pour la résolution de CSPs
3. Réalisation de solveurs de contraintes en Prolog
4. Programmation par contraintes avec Gnu-Prolog
5. Exercices de programmation par contraintes

► Séance 3 « Réalisation de solveurs en Prolog »

- Objectif : mieux comprendre les algorithmes vus à la séance 2
- Restriction aux CSPs binaires
 - Description d'un CSP binaire en Prolog par 2 prédicats
 - *variables*(L) unifie L avec la liste des variables/domaines
 - *consistants*((X_i, V_i), (X_j, V_j)) réussit si $(V_i, V_j) \in C_{(X_i, X_j)}$
- Le code décrivant les CSPs des reines, du coloriage et des mariages est donné à l'étudiant
- L'étudiant programme les différents algorithmes...
- ... et les compare expérimentalement sur les CSPs des reines, le coloriage et les mariages



Organisation du cours

► 5 séances de 45 mn :

1. Contraintes et CSPs
2. Algorithmes pour la résolution de CSPs
3. Réalisation de solveurs de contraintes en Prolog
4. Programmation par contraintes avec Gnu-Prolog
5. Exercices de programmation par contraintes

► Séance 4 « la PPC avec Gnu-Prolog »

- Rapide panorama des différents langages de PPC
- Présentation des prédicats « PPC » principaux de Gnu-Prolog (... avec des liens sur le manuel en ligne)
 - Variables sur les domaines finis
 - Contraintes sur les domaines finis
 - Résolution de CSPs
- Un premier exemple : les reines
- Un deuxième exemple : les mariages stables



Organisation du cours

▶ 5 séances de 45 mn :

1. Contraintes et CSPs
2. Algorithmes pour la résolution de CSPs
3. Réalisation de solveurs de contraintes en Prolog
4. Programmation par contraintes avec Gnu-Prolog
5. Exercices de programmation par contraintes

▶ Séance 5 « Exercices de PPC »

- ▶ Travaux Pratiques : codage en Gnu-Prolog des CSPs modélisés pendant la séance 1
 - ~> Retour de monnaie d'un distributeur automatique
 - ~> Sel et moutarde (Lewis Carroll)
 - ~> Coloriage d'une carte
 - ~> Crypto « SEND + MORE = MONEY »
 - ~> Le zèbre (Lewis Carroll)



Conclusion

- ▶ Première « expérimentation » des cours du module IA
 - ▶ 1 groupe d'étudiants de Lyon 1 en cours du soir
 - ▶ Premiers « cours » en janvier 2003
 - ▶ ... pas encore de retour d'expérience !
- ▶ Evaluation du temps / difficultés rencontrées
 - ▶ Maîtrise suffisante du langage Prolog ???
- ▶ Un cours accessible à tous

<http://www710.univ-lyon1.fr/~csolnon/Site-PPC/e-miage-ppc-som.htm>