Des méthodes et outils originaux pour enseigner la modélisation en programmation linéaire : l'expérience de caseine

Nadia Brauner, Hadrien Cambazard, Nicolas Catusse, Pierre Lemaire

Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, F-38000 Grenoble, France
{prenom.nom}@grenoble-inp.fr

Mots-clés : *Programmation linéaire, évaluation automatique, formation.*

Nous présentons une méthode pour tester automatiquement la validité d'un modèle de programme linéaire (PL) proposé par un étudiant à partir de la description en langage naturel d'un problème pratique et d'un modèle de référence qui correspond à la correction attendue par l'enseignant. Cette méthode a été implémentée dans un outil proposé sur la plateforme pédagogique caseine.org. Elle est utilisée sur différents types d'exercices partagés par les enseignants. Nous présentons la méthode, l'outil qui instancie cette méthode sur des exercices pratiques et les données d'utilisation sur les exercices faits par quelques centaines d'étudiants issus d'une dizaine d'universités françaises et internationales. En conclusion, nous expliquons comment rejoindre le projet, et proposer éventuellement ces exercices à vos étudiants.

1 Comment évaluer automatiquement un modèle de PL

Une modélisation en programmation linéaire est considérée comme valide si (1) c'est un programme linéaire (fonction objectif et contraintes linéaires) et (2) il produit les mêmes solutions qu'un modèle de référence. Le premier point est facile à vérifier et correspond à l'analyse statique du modèle de l'étudiant. Pour le second point, on ne souhaite pas imposer de décisions de modélisation comme les variables, leur nom, l'ordre des contraintes, le nombre de variables ou contraintes ; donc la vérification du polyèdre est plus délicate. L'idée de base est de faire varier les coefficients de la fonction-objectif afin que chaque point extrême du polyèdre de référence soit optimal pour au moins l'une des fonctions-objectifs de test. On peut alors vérifier que la valeur de chaque fonction-objectif dans chaque point extrême du polyèdre de référence est la même pour le modèle de l'étudiant et pour le modèle de référence. On vérifie aussi toutes les fonctions-objectifs parallèles aux contraintes pour s'assurer que le polyèdre de l'étudiant ne possède pas de points en plus.

2 Implémentation et usage

L'outil de vérification automatique qui implémente cette méthode est intégré à un système classique de gestion de formation (LMS) Moodle. La vue d'un exercice du point de vue étudiant est donnée Figure 1. Le vecteur c imposé dans l'exercice permet de faire varier la fonction-objectif. Plus généralement, n'importe quelle donnée du problème peut être ainsi définie par un paramètre que l'on peut faire varier pour valider la modélisation.

La plateforme contient une collection d'exercices partagés par une communauté d'enseignants qui peuvent les intégrer dans leurs cours. Chaque enseignant peut contribuer en choisissant de partager ses propres exercices.

Les exercices peuvent avoir différents types de difficultés qui vont de modèles très simples à des problèmes où la complexité peut venir aussi bien des techniques de modélisation à mettre en œuvre que de la taille de la description du problème ; par exemple pour des problématiques

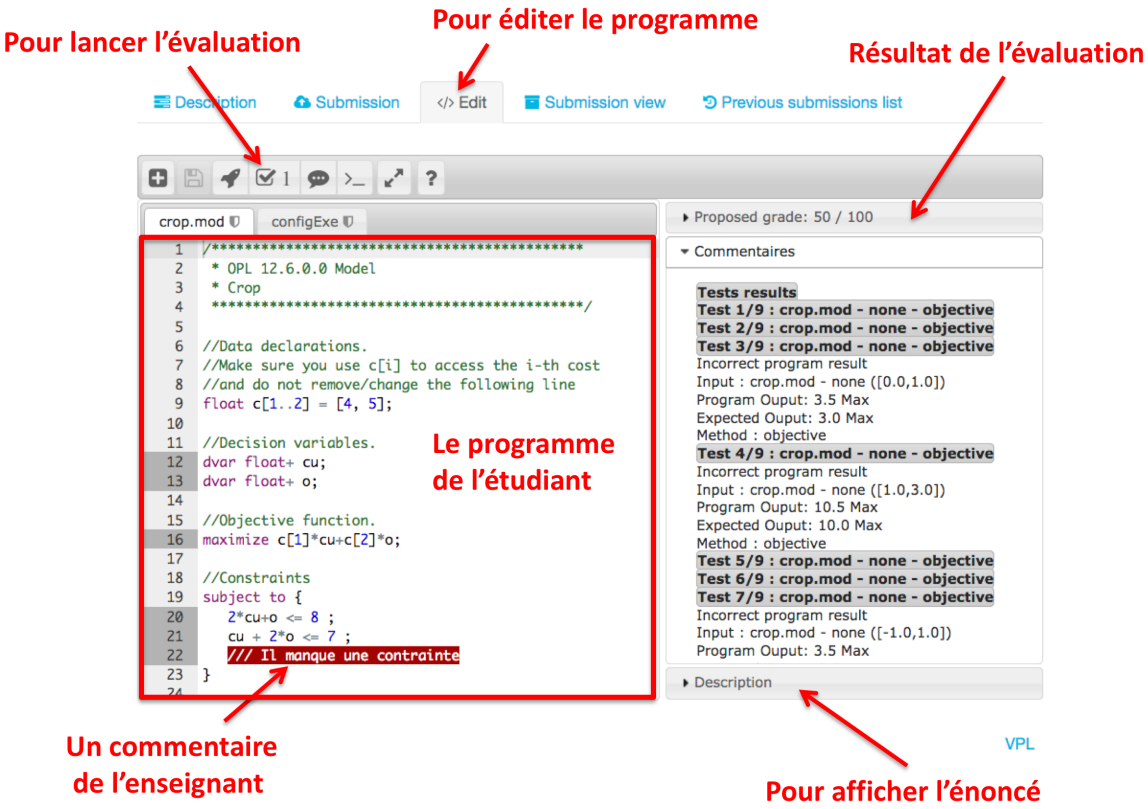

FIG. 1 – Vue étudiante d'un exercice de modélisation avec correction automatique

industrielles avec des données externes et un grand nombre de types de contraintes. La collection contient également des exercices de modélisation des problèmes classiques d'optimisation combinatoire (sac-à-dos, bin packing...) et de problèmes de graphes (stable, coloration...). Nous présenterons l'usage de ces exercices à partir des données issues de la plateforme.

3 Conclusion

Pour accompagner ces exercices de modélisation, nous présenterons des outils additionnels qui permettent par exemple de donner des indications à l'étudiant sur ses fautes, ou d'utiliser un environnement de développement adapté (OPL Studio par exemple). Actuellement, le système fonctionne avec le langage OPL ou avec l'API CPLEX. L'un des objectifs est d'intégrer d'autres langages de modélisation. Nous montrerons aussi quelques outils pour accompagner la compréhension de l'étudiant des méthodes de résolution de PL, de la dualité ou de l'analyse de sensibilité. Vous pouvez tester les exercices proposés ici :
`https://caseine.org/course/view.php?name=LPOpen`.