

## AMS : PROJET RECHERCHE BIBLIOGRAPHIQUE EN LABORATOIRE

### Langages de modélisation algébrique

Xavier GANDIBLEUX

## 1 Présentation

Les solveurs logiciel dédiés aux problèmes d'optimisation sont principalement composés de deux grandes parties : les langages de modélisation algébrique et les moteurs d'optimisation. GMP (GNU MathProg modeling language) et GLPK (GNU Linear Programming Kit) sont des illustrations pour respectivement ces deux entités (<https://www.gnu.org/software/glpk/>).

Les travaux de recherche menés en optimisation à Nantes utilisent abondamment des langages de modélisation algébrique et des contributions open-source sont menés en vue d'étendre le champs d'expression d'un langage donné. Par exemple, les travaux de recherche sur vOptSolver, qui est un écosystème pour la modélisation et la résolution de problèmes d'optimisation linéaires multi-objectif (<https://voptsolver.github.io/vOptSolver/>), se basent sur JuMP, Julia<sup>1</sup> for Mathematical Optimization (<https://jump.dev/JuMP.jl/stable/>).

## 2 Consignes et attendus du travail

Le projet de recherche bibliographique en laboratoire<sup>2</sup> en L2 CMI a pour objet d'initier aux pratiques de recherche et de traitement de l'information. Il est le premier lien avec le monde de la recherche et de l'innovation : thématique de recherche bibliographique proposée par le laboratoire, entrée des étudiants dans les laboratoires, nouveaux échanges avec leurs enseignants sur leurs activités de recherche, etc.

Les *langages de modélisation algébrique* est le sujet proposé dans le cadre de cette étude aux étudiants de la spécialité OPT/IM pour ce projet. Les étudiants auront à recenser les principaux langages existants permettant de modéliser des programmes linéaires en programmation mathématique. Ils auront à les présenter leur investigations d'une manière synthétique et uniformisée. Pour les alimenter dans cet exercice, ils auront à rencontrer des enseignants-chercheur du laboratoire (dont Anthony Przybylski, Evgeny Gurevsky et Xavier Gandibleux) qui soient soit utilisateurs de langages de modélisation algébrique, soit contributeurs à ces derniers, avec pour objectif d'entendre leur expertise sur le sujet et la verser dans la synthèse.

Le livrable final se présentera sous la forme d'un mémoire rapportant l'état de l'art sur le sujet. Il sera demandé d'illustrer de manière claire et pédagogique les langages rapportés. Il sera attendu de trouver dans le mémoire des moments de synthèse, sous forme tabloïde permettant au lecteur d'avoir une vue d'ensemble rapidement. Le mémoire sera produit avec L<sup>A</sup>T<sub>E</sub>X, en faisant usage de BibTeX. Les étudiants pourront s'inspirer d'études bibliographiques existantes, présentes dans les thèses du domaine, voir par exemple <https://www.theses.fr/178912662> ou dans des articles de synthèse, voir par exemple <https://link.springer.com/article/10.1007/s002910000046>.

Un point de rencontre régulier (fréquence à fixer) avec le promoteur du sujet sera à fixer pour les besoins de suivi de l'avancement des travaux.

---

1. Julia est un langage de programmation open-source ciblant spécialement différents besoins rencontrés dans le calcul scientifique (optimisation, statistique, équations différentielles, algèbre, physique, astronomie, machine learning, etc.) ; voir <https://julialang.org/> et <https://julialang.org/community/>.

2. <https://cminantes.wordpress.com/>