

Modélisation Hexaly assistée par LLM

Julien Darlay¹, Nicolas Violot¹

Hexaly

{jdarlay, nviolot}@hexaly.com

Mots-clés : *Modélisation, LLM*

1 Introduction

Les avancées dans le domaine du *machine learning* ont rendu son utilisation possible pour la résolution de problèmes d’optimisation [1, 2, 3]. Un axe de recherche complémentaire consiste à utiliser des grands modèles de langage (LLM) pour la génération de modèles mathématiques à partir d’une description en langage naturel [4]. Les travaux de la littérature se focalisent essentiellement sur des modélisations sous forme de programmes linéaires en nombres entiers.

Hexaly est un solveur proposant un formalisme de modélisation non linéaire et ensembliste. La modélisation de problèmes combinatoires classiques tels que les problèmes de *packing*, de tournées de véhicules ou encore d’ordonnancement nécessite d’utiliser des variables ensemblistes comme les listes ou les intervalles. Le langage de modélisation fourni par Hexaly est un langage spécifique (HXM) avec un corpus très réduit par rapport à d’autres langages de programmation comme Python, le rendant peu représenté dans le corpus d’entraînement des LLM.

Cette présentation s’intéressera à la capacité des LLM récents (GPT5.0, Grok4.5, Sonnet4.5¹) à générer un modèle Hexaly à partir d’une description en langage naturel.

2 Expérimentations

La littérature propose plusieurs benchmarks pour évaluer la performance des LLM pour générer des modèles mathématiques. Un des plus cités est celui fourni pour la compétition NL4OPT lors de la conférence NeurIPS2022 [5]. Ce jeu de données contient une description en langage naturel de 1101 problèmes d’optimisation ainsi que les informations permettant de reconstruire le modèle mathématique. Un exemple de description, adapté au contexte d’Hexaly est présenté ci-dessous :

A hotel employs cleaners and receptionists. Cleaners earn \$500 per week and receptionists earn \$350 per week. The hotel requires a minimum of 100 workers of whom at least 20 must be receptionists. To keep the hotel clean and running smoothly, the number of receptionists should be at least a third of the number of cleaners. Formulate an Hexaly model using Hexaly’s modeling language (HXM) to minimize the wage bill.

Ces problèmes difficiles à modéliser pour des LLM en 2022 sont aujourd’hui à la portée des derniers LLM. Nos premières expérimentations avec GPT o4-mini indiquent que dans 77% des cas le modèle Hexaly généré est correct : la syntaxe est juste et la solution optimale est identique à la solution optimale du modèle reconstruit à partir des données. Dans 20% des cas, le modèle généré contient des erreurs de syntaxe liées au langage HXM. Les erreurs restantes sont de réelles erreurs de modélisation. Les erreurs de syntaxe peuvent être corrigées en intégrant au LLM des outils d’analyse syntaxique non détaillés dans ce résumé. En revanche, ces benchmarks

1. respectivement chatgpt.com, grok.com, claude.ai

Problèmes	GPT5.0	Grok4	Sonnet4.5
<i>network design</i>	✓	✓*	✗
<i>vehicle routing</i>	✓*	✓*	✓*
<i>scheduling</i>	✓*	✓	✗

TAB. 1 – Correction des modèles générés. Un ✓ indique un modèle juste, ✓* indique des corrections syntaxiques mineures, ✗ indique un modèle faux.

ne permettent pas de mesurer la capacité des LLM à faire une bonne modélisation dans le langage d’Hexaly : les problèmes sont petits, linéaires et ne permettent pas d’exploiter la modélisation ensembliste.

Nous utiliserons pour la suite trois descriptions en langage naturel de problèmes de *network design*, *vehicle routing* et *scheduling* déjà utilisées pour comparer des formulations PLNE [6]. La synthèse des résultats obtenus sur ces trois problèmes et les trois LLM sont présentés Tab 1. Les modèles trouvés par Grok4 et GPT5.0 sont justes et ne nécessitent que des corrections syntaxiques mineures.

3 Conclusions et perspectives

Les dernières versions des LLM permettent de modéliser correctement la plupart des problèmes linéaires contenus dans les benchmarks de la littérature. Ils permettent également de fournir de bonnes modélisations en utilisant les opérateurs ensemblistes d’Hexaly. Des techniques comme des outils d’analyse syntaxique et la génération d’instructions supplémentaires en préambule de la requête utilisateur permettent de corriger les erreurs de syntaxe et d’obtenir des modèles corrects à partir d’une description en langage naturel. Ces méthodes sont en cours d’intégration dans Hexaly Studio, l’environnement de modélisation en ligne d’Hexaly.

Références

- [1] Y. Bengio, A. Lodi, A. Prouvost. *Machine learning for combinatorial optimization : a methodological tour d’horizon*. European Journal of Operational Research, 290(2), 405-421.
- [2] W. Chen, M. Tanneau, P. Van Hentenryck. *End-to-End Feasible Optimization Proxies for Large-Scale Economic Dispatch* IEEE Transactions on Power Systems, 2023
- [3] L. Baty, K. Jungel, P. S. Klein, A. Parmentier, M. Schiffer. *Combinatorial optimization-enriched machine learning to solve the dynamic vehicle routing problem with time windows*. Transportation Science, 58(4), 708-725.
- [4] T. Ahmed, S. Choudhury. *LM4OPT : Unveiling the potential of Large Language Models in formulating mathematical optimization problems*. INFOR : Information Systems and Operational Research, 62(4), 559-572.
- [5] R. Ramamonjison, T. Yu, et al. *NL4opt competition : Formulating optimization problems based on their natural language descriptions*. NeurIPS 2022 competition track. PMLR, 2023.
- [6] B. Schaefer *Which LLM Builds the Best Optimization Model ?* INFORMS 2025