



Université de Yaoundé I

Département d'Informatique

Groupe 8: Projet Informatique

Calcul Symbolique et la Bibliothèque SymPy

Nom et Prénom	Matricule
Nangmo Feulfack Annick Duplesse	21S2530
Njiagupmun Njankouo Awa Rahma	22U2114
Nguefack Tangomo Chris Arthur	—

October 2, 2025

Superviseur : Pr. Paulin Melatagia

Plan

- 1 Introduction
- 2 Qu'est-ce que le Calcul Symbolique ?
- 3 Différence avec le Calcul Numérique
- 4 Présentation de SymPy
- 5 Caractéristiques de SymPy
- 6 Exemple de Code
- 7 Applications
- 8 Avantages et Limites
- 9 Comparaison avec d'autres Outils
- 10 Conclusion

- Le calcul symbolique est une branche des mathématiques qui manipule des objets mathématiques sous forme symbolique.
- Contrairement au calcul numérique, il fournit des résultats exacts.
- La bibliothèque SymPy en Python permet d'effectuer ces calculs.

Définition et Objectifs

- Manipulation d'expressions algébriques exactes.
- Objectifs : simplification, dérivation, intégration, résolution d'équations.
- Exemple : factorisation de $x^2 - 4$ en $(x - 2)(x + 2)$.

Calcul Symbolique vs Calcul Numérique

- Calcul numérique \rightarrow approximations (ex. $\pi \approx 3.14$).
- Calcul symbolique \rightarrow exact (ex. π reste π).
- Complémentarité des deux approches.

Qu'est-ce que SymPy ?

- Bibliothèque Python open-source.
- Fournit des outils pour : algèbre, calcul différentiel et intégral, équations différentielles.
- Conçu pour être léger et facilement intégrable.

Fonctionnalités principales

- Simplification d'expressions.
- Résolution d'équations.
- Dérivation et intégration symbolique.
- Calcul matriciel et manipulation de polynômes.

Exemple avec SymPy

```
from sympy import symbols, expand, factor

x = symbols('x')
expr = (x + 2) * (x - 2)
print(expand(expr))      #  $x^2 - 4$ 
print(factor(x**2-4))    #  $(x - 2)(x + 2)$ 
```


Applications du Calcul Symbolique

- Recherche scientifique.
- Ingénierie et physique.
- Éducation (aide à l'apprentissage des mathématiques).
- Développement de logiciels scientifiques.

- Résultats exacts.
- Automatisation de tâches complexes.
- Polyvalence et accessibilité avec Python.

- Peut être plus lent que le calcul numérique.
- Difficultés avec des expressions trop complexes.
- Consommation mémoire élevée pour certains calculs.

SymPy vs Autres Logiciels

- **SymPy** : open-source, léger, Python.
- **Mathematica / Maple** : très puissants mais propriétaires.
- **SageMath** : intègre SymPy et d'autres outils.

Conclusion

- Le calcul symbolique joue un rôle essentiel en mathématiques et en informatique.
- SymPy démocratise son utilisation grâce à Python.
- C'est un outil pratique pour étudiants, chercheurs et ingénieurs.