



CALCUL SYMBOLIQUE & SYMPY

NOMS ET MATRICULE DES MEMBRES :

BELL ARSÈNE KEVIN	22T2960
MELONG LETHYCIA	22W2147
DASSI MANDJO LEA JUSTINE	22W2164
AKONO NOAH CAROLE JESSICA	22T2892

Sous la supervision de Pr. MELATAGIA Paulin

Année Académique : 2025-2026

Structure de la présentation

① Introduction

② Calcul Symbolique

③ Sympy

④ Conclusion

Introduction

- Le **calcul symbolique**, ou *Computer Algebra*, consiste à manipuler des expressions mathématiques sous forme **exacte** plutôt que numérique.

Contrairement au calcul numérique, qui donne des approximations, il permet d'obtenir des résultats rigoureux : équations résolues, intégrales calculées, simplifications d'expressions.

- Ces méthodes sont essentielles dans de nombreux domaines : **mathématiques, physique, ingénierie, informatique, cryptographie**.
- Parmi les outils existants (Maple, Mathematica, Maxima), **SymPy** se distingue comme une bibliothèque Python **open source**, simple d'utilisation et facilement intégrable dans des programmes.

Le calcul symbolique : définition et objectifs

- Le calcul symbolique, ou calcul formel, permet de manipuler des expressions mathématiques **de manière exacte** en utilisant des symboles au lieu de valeurs numériques.
- Contrairement au calcul numérique, il permet de :
 - Simplifier des expressions
 - Dériver ou intégrer des fonctions
 - Résoudre des équations
 - Effectuer des substitutions et transformations algébriques
- Son utilisation est faite au travers des outils suivants MATLAB, Mathematica, SymPy (Python) pour faciliter l'expérimentation mathématique et la résolution de problèmes complexes.

Fonctionnement et techniques

- **Manipulation d'expressions** : traiter des expressions dans leur forme symbolique, ex. $x + y + 1$ au lieu de $3+4$.
- **Simplification automatique** : développer des produits, réduire au même dénominateur, factoriser.
- **Opérations différentielles et intégrales** : calcul formel de dérivées et intégrales, indéfinies ou définies.
- **Substitution** : remplacer des symboles par d'autres expressions ou des valeurs numériques.
- **Résolution d'équations** : équations linéaires, non-linéaires, et certaines équations différentielles.

Avantages et Limites du calcul symbolique

- **Avantages :**

- Exactitude et rigueur des résultats
- Automatisation de calculs complexes
- Aide à la démonstration et à l'analyse théorique

- **Limites :**

- Temps de calcul parfois élevé
- Complexité pour des expressions très volumineuses

Applications

- **Recherche mathématique :**

- Formulation de conjectures
- Validation d'hypothèses
- Aide à la preuve de théorèmes

- **Ingénierie et sciences :**

- Mécanique et physique appliquée
- Cryptographie
- Résolution de problèmes non-linéaires difficiles

- **Enseignement des mathématiques :**

- Les calculatrices et logiciels symboliques ont révolutionné l'apprentissage dans les années 1990.
- Permettent aux étudiants de se concentrer sur la compréhension conceptuelle plutôt que sur des calculs laborieux.
- Favorisent la visualisation et l'expérimentation interactive des concepts mathématiques.

Introduction à SymPy

- **SymPy** est une bibliothèque Python open source pour le calcul symbolique.
- Elle permet de manipuler des expressions mathématiques exactes dans un environnement Python.
- **Objectifs et points forts :**
 - Combiner rigueur mathématique et flexibilité de Python
 - Intégration avec l'écosystème scientifique (NumPy, matplotlib, Jupyter Notebook)
 - Permet le calcul symbolique et numérique
 - Gratuit et sans dépendances lourdes
- Applications : recherche scientifique, enseignement, prototypage rapide.

Exemple avec SymPy

Exemple de calcul symbolique

```
from sympy import symbols, diff, integrate, solve
```

```
x = symbols('x')  
expr = x**2 + 3*x + 2
```

```
print(diff(expr, x))  
print(integrate(expr, x))  
print(solve(expr, x))
```

la dérivée sera $2x + 3$

l'intégrale sera $x^3/3 + 3x^2/2 + 2x$

les solutions seront $[-2, -1]$

Avantages et limites

- **Avantages :**

- Gratuit et open source
- Facile à intégrer avec Python
- Large éventail de fonctionnalités symboliques

- **Limites :**

- Performance limitée pour des expressions très volumineuses
- Interface moins interactive que des logiciels commerciaux comme Mathematica

Comparaison des outils : SymPy vs Mathematica vs MATLAB

Critère	SymPy	Mathematica	MATLAB
Langage	Python	Wolfram	MATLAB
Type	Open source	Commercial	Commercial
Calcul symbolique	Oui	Oui	Oui*
Calcul numérique	Oui	Oui	Oui
Intégration	Excellente	Moyenne	Moyenne
Licence	Gratuit	Payant	Payant
Applications	Enseignement, recherche	Recherche, visualisation	Ingénierie, simulations
Interface	Console/Jupyter	Notebooks/GUI	GUI/Scripts

Conclusion

- Le **calcul symbolique** permet de manipuler des expressions mathématiques de manière exacte, offrant précision et rigueur dans les calculs.
- **SymPy** rend le calcul symbolique accessible dans Python, combinant théorie et programmation.
- Ses points forts : open source, gratuit, facile à intégrer avec Python, adapté à l'enseignement, à la recherche et au prototypage scientifique.
- Limites : performance limitée pour des expressions très volumineuses et interface moins interactive que des logiciels commerciaux comme Mathematica.
- En résumé : SymPy est un outil puissant et flexible pour l'apprentissage, l'expérimentation et l'automatisation des calculs mathématiques.

Merci pour votre attention !