

Outils de combinatoire analytique en sage

Projet STL

Matthieu Dien Marguerite Zamansky

Université Pierre et Marie Curie

14 mai 2013

Objectifs

Sage

Combinatoire
analytique

Implémentation

- Fournir un outil de calcul symbolique pour des séries multivariées,
- de préférence libre :
- offrir une alternative à Mapple et porter les fonctionnalités de Gfun vers Sage.

Sommaire

1 Sage

2 Combinatoire analytique

3 Implémentation

Sage

Sage

Combinatoire analytique

Implémentation



- logiciel libre de calcul formel numérique et symbolique
- regroupe des outils déjà connus et éprouvés (GP/PARI, GAP, Singular, Maxima)
- et ses propres paquets (combinat, rings, matrix ...)
- le tout interfacé par un top-level Python

Sommaire

1 Sage

2 Combinatoire analytique

3 Implémentation

Définitions

Série génératrice multivariée

Une série génératrice A associée à une classe combinatoire \mathcal{A} :

$$A(X_1, \dots, X_k) = \sum_{i_1, \dots, i_k \geq 0}^{+\infty} a_{i_1, \dots, i_k} X_1^{i_1} \cdots X_k^{i_k}$$

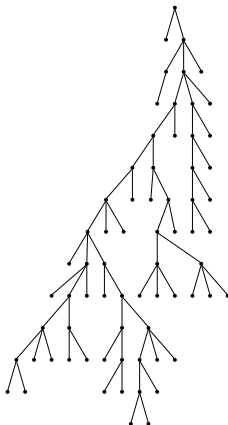
permet de compter le nombre d'élément de \mathcal{A} .

Séries Génératrices Multivariées

Exemple

Arbres binaires-ternaires :

$$\mathcal{ABT} = \{z\} \cdot \{w\} + \{z\} \cdot \{u\} \cdot \mathcal{ABT}^2 + \{z\} \cdot \{v\} \cdot \mathcal{ABT}^3$$



Exemple

$ABT(z, u, v, w) =$	
$z \cdot w +$	$z \cdot w$
$z^3 \cdot u \cdot w^2 +$	$ \begin{array}{c} z \cdot u \\ \swarrow \quad \searrow \\ z \cdot w \quad z \cdot w \end{array} $
$z^4 \cdot v \cdot w^3 +$	$ \begin{array}{c} z \cdot v \\ \swarrow \quad \quad \searrow \\ z \cdot w \quad z \cdot w \quad z \cdot w \end{array} $
$2 \cdot z^5 \cdot u^2 \cdot w^3 +$	$ \begin{array}{cc} \begin{array}{c} z \cdot u \\ \swarrow \quad \searrow \\ z \cdot u \quad z \cdot w \\ \swarrow \quad \searrow \\ z \cdot w \quad z \cdot w \end{array} & \begin{array}{c} z \cdot u \\ \swarrow \quad \searrow \\ z \cdot w \quad z \cdot u \\ \swarrow \quad \searrow \\ z \cdot w \quad z \cdot w \end{array} \end{array} $
\dots	\dots

Sommaire

1 Sage

2 Combinatoire analytique

3 Implémentation

Implémentation

Difficultés

- Représenter les séries formelles en mémoire
- S'intégrer à un projet de grande envergure

Implémentation

Difficultés

- Représenter les séries formelles en mémoire
- S'intégrer à un projet de grande envergure

Solutions

- Utilisation de streams (programmation paresseuse).
- Générateurs python

Générateurs Python

- Permet de créer des objets itérables
- `yield`

exemple : itérer sur les entiers

```
def integers_definition():  
    i = 0  
    while True :  
        yield i  
        i += 1  
  
for n in integers_definition():  
    if n % 2 == 0:  
        print("%d_est_pair"%n)  
    else :  
        print("%d_est_impair"%n)
```

Représentation

Un stream de listes de couples formés d'un entier et d'un n -uplet.

entrée	mémoire
F	[]
F.coefficients(2); F	[[] , [] , [(1, [0, 0, 1, 1])]]
F.coefficients(14); F	[[] , [] , [(1, [0, 0, 1, 1])] , [] , [] , [] , [(1, [1, 0, 2, 3])] , [] , [(2, [0, 1, 3, 4])] , [] , [(2, [2, 0, 3, 5])] , [] , [(5, [1, 1, 4, 6])] , [] , [(5, [3, 0, 4, 7])] , [(3, [0, 2, 5, 7])]]

Démonstration

Arbres binaires-ternaires

Ce qu'on écrit dans Sage,

```
R.<z,u,v,w> = FormalMultivariatePowerSeriesRing(QQ)
BTtree = R()
BTtree.define(z*w+z*u*BTtree^2+z*v*BTtree^3)
BTtree.coefficients(30)
BTtree
```

Démonstration

Arbres binaires-ternaires

Ce qu'on écrit dans Sage,

```
R.<z,u,v,w> = FormalMultivariatePowerSeriesRing(QQ)
BTtree = R()
BTtree.define(z*w+z*u*BTtree^2+z*v*BTtree^3)
BTtree.coefficients(30)
BTtree
```

le résultat qu'on obtient.

$$\begin{aligned} & z*w + z^3*u*w^2 + z^4*v*w^3 + 2*z^5*u^2*w^3 + 5*z^6*u*v*w^4 + \\ & 5*z^7*u^3*w^4 + 3*z^7*v^2*w^5 + 21*z^8*u^2*v*w^5 + 14*z^9*u^4*w^5 + \\ & 28*z^9*u*v^2*w^6 + 84*z^{10}*u^3*v*w^6 + 12*z^{10}*v^3*w^7 + 42*z^{11}*u^5*w^6 \\ & + 180*z^{11}*u^2*v^2*w^7 + 330*z^{12}*u^4*v*w^7 + 165*z^{12}*u*v^3*w^8 + \\ & 132*z^{13}*u^6*w^7 + 990*z^{13}*u^3*v^2*w^8 + 55*z^{13}*v^4*w^9 + \\ & 1287*z^{14}*u^5*v*w^8 + 1430*z^{14}*u^2*v^3*w^9 \end{aligned}$$

Conclusion

Opérateurs implémentés

- Addition
- Produit
- Séquence
- Dérivée
- Composition
- Cast en polynôme

Conclusion

Opérateurs implémentés

- Addition
- Produit
- Séquence
- Dérivée
- Composition
- Cast en polynôme

Contribution

- Patch bug dans Sage
- Proposition du package

Continuer l'implémentation pour avoir les fonctionnalités disponibles dans Gfun :

- comme les algorithmes *algfuntoalgeq*, *algeqtodiffeq*, *diffeqtorec*
- Sage days, du 17 au 21 juin