Simplification de paramètres

Alexandre Temperville, doctorant
encadré par F.Lemaire et F.Boulier,
équipe CFHP, groupe CO2

CRIStAL, Université Lille

JNCF 2015, Cluny

Introduction

Objectif

Trouver des changements de variables permettant de diminuer les degrés des paramètres dans des systèmes d'équations pour les simplifier.

Intérêt

Lorsque l'on étudie des modèles (systèmes) avec des paramètres, les méthodes utilisées sont souvent coûteuses (calcul de points fixes, étude de bifurcations par exemple).
Diminuer les degrés (ou éliminer des paramètres) sur le modèle initial permet de simplifier leur étude.

On s'intéressera d'abord à simplifier des fractions rationnelles avant de généraliser à des systèmes d'équations en contenant.

Exemple qui a motivé le travail

Simplifier des équations dans des systèmes (BLLM08)

$$(S) : \begin{cases} G'(t) = 1 - G(t) - k_1 k_2 k_3 \alpha P_1(t)^4 G(t) \\ M'(t) = -\delta_M M(t) + G(t) + \rho_b - \rho_b G(t) \\ P_1'(t) = \dfrac{4 - 4G(t) + \beta M(t) - k_1 k_2 k_3 \alpha \delta_P P_1(t)^4 G(t)}{16 k_1 k_2 k_3 P_1(t)^3 + 9 k_1 k_2 P_1(t)^2 + 4 k_1 P_1(t) + 1} \end{cases}$$

On a envie d'effectuer manuellement les changements de variables $\overline{k_1} = k_1$, $\overline{k_2} = k_1 k_2$ et $\overline{k_3} = k_1 k_2 k_3$.

Problématique

Comment retrouver algorithmiquement ces changements de variables ?

Sommaire

Principe

Fraction rationnelle

Soit $q = \dfrac{a + b^2 y}{a + bx}$ une fraction rationnelle.

Principe

On cherche à faire un changement de variable sur a et b uniquement permettant de faire baisser les degrés sur a et b.

On cherche un changement de variable de la forme $\begin{cases} a = f(\bar{a}, \bar{b}) = \bar{a}^{\alpha} \bar{b}^{\beta} \\ b = g(\bar{a}, \bar{b}) = \bar{a}^{\gamma} \bar{b}^{\delta} \end{cases}$

avec $\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$ inversible **à coefficients rationnels**.

Changement de variable

Fraction rationnelle

Soit $q = \dfrac{a + b^2 y}{a + bx}$ une fraction rationnelle.

On a envie de poser $\begin{cases} \bar{a} = \frac{b^2}{a} \\ \bar{b} = \frac{b}{a} \end{cases}$.

Changement de variable

Si on réalise le changement de variables $\begin{cases} a = \frac{\bar{a}}{\bar{b}^2} \\ b = \frac{\bar{a}}{\bar{b}} \end{cases}$, on obtient

$$q = \frac{\bar{b}^{-1} + \bar{a}\bar{b}^{-1} y}{\bar{b}^{-1} + x} = \frac{1 + \bar{a} y}{1 + \bar{b} x}.$$

Représentation matricielle d'une fraction

On réécrit $q = \frac{a+b^2 y}{a+bx} = \frac{am_1+b^2 m_2}{am_3+bm_4}$ (avec $m_1 = m_3 = 1, m_2 = y, m_4 = x$).

Forme matricielle

On représente q sous forme de matrice dans laquelle les colonnes correspondent aux monômes et les lignes aux degrés des paramètres dans ces différents monômes :

$$N = \begin{array}{cc} \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{array}\right) & \begin{array}{c} a \\ b \end{array} \\ \begin{array}{cccc} m_1 & m_2 & m_3 & m_4 \end{array} & \end{array} .$$

Changement de variable sur N

Question

Comment agit notre changement de variable sur la matrice N ?

Sur notre exemple

Changement de variables $\begin{cases} a = \frac{\bar{a}}{\bar{b}^2} \\ b = \frac{\bar{a}}{\bar{b}} \end{cases} \implies C = \begin{array}{cc} & \begin{array}{cc} a & b \end{array} \\ \begin{array}{c} \bar{a} \\ \bar{b} \end{array} & \begin{pmatrix} 1 & 1 \\ -2 & -1 \end{pmatrix} \end{array}$.

Faire agir C à gauche sur N revient à effectuer le changement de variable.

$N' = C.N = \left(\begin{array}{cc|cc} 1 & 2 & 1 & 1 \\ -2 & -2 & -2 & -1 \end{array}\right) \begin{array}{c} \bar{a} \\ \bar{b} \end{array}$ représente la fraction suivante :

$$q = \frac{\frac{\bar{a}}{\bar{b}^2} + \frac{\bar{a}^2}{\bar{b}^2} y}{\frac{\bar{a}}{\bar{b}^2} + \frac{\bar{a}}{\bar{b}} x} = \ldots = \frac{1 + \bar{a} y}{1 + \bar{b} x}.$$

Non-unicité de la représentation matricielle

Non-unicité

La représentation matricielle d'une fraction n'est pas unique.

Différentes représentations

On peut, par exemple, représenter $\dfrac{1+\bar{a}y}{1+\bar{b}x}$ avec

$$N' = \begin{array}{c} \left(\begin{array}{cc|cc} 1 & 2 & 1 & 1 \\ -2 & -2 & -2 & -1 \end{array}\right) \end{array} \begin{array}{c} \bar{a} \\ \bar{b} \end{array} \quad \textbf{ou} \quad N' = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \begin{array}{c} \bar{a} \\ \bar{b} \end{array} .$$

Remarque

Ajouter des multiples de $\mathbb{1} = \left(\begin{array}{cc|cc} 1 & 1 & 1 & 1 \end{array}\right)$ aux lignes de N' ne change pas la fraction.

Comment trouver un changement de variables ?

Problématique : représentation creuse de q

On veut trouver un changement de variable C tel que $C.N$ soit “simple”, avec beaucoup de 0 par exemple.

Comment obtenir $C = \begin{array}{cc} a & b \\ \end{array}$ $\begin{pmatrix} 1 & 1 \\ -2 & -1 \end{pmatrix}\begin{array}{c} \bar{a} \\ \bar{b} \end{array}$ et comment trouver $q = \dfrac{1+\bar{a}y}{1+\bar{b}x}$

avec une représentation $N' = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right)\begin{array}{c} \bar{a} \\ \bar{b} \end{array}$ creuse ?

Idée : utiliser l'algorithme CSB

L'**algorithme** *CSB* [Lemaire, Temperville, “On Defining and Computing Good Conservation Laws”, CMSB 2014] est un outil qui fait apparaître des zéros dans une base.

On veut l'utiliser sur la matrice $\begin{pmatrix} N \\ \mathbb{1} \end{pmatrix}$ s'il s'agit d'une **base par ligne**.

Principe de l'algorithme CSB

Toutes les bases de cet exposé sont stockées **par ligne** dans des matrices.

CSB (Compute Sparsest Basis)

- calcule une base la plus creuse possible à partir d'une base donnée,
- produit **généralement** une base de norme 1 plus petite que celle de la base initiale.

Exemple

On applique CSB sur la base $B = \begin{pmatrix} 1 & 1 & 2 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}$.
CSB parcourt un arbre binaire et trouve une combinaison linéaire à effectuer $B_1 \leftarrow B_1 - B_2 - B_3$ pour obtenir : $B = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}$.
Sur cette nouvelle base B, aucune ligne ne peut être rendue plus creuse.
B est donc une base la plus creuse.

Sommaire

Pré-traitement de N (1)

Les matrices N ne sont pas toujours des bases.

Exemple

Pour $q = \dfrac{1+abx}{1+aby}$, on a $N = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{array}\right) \begin{array}{c} a \\ b \end{array}$. On peut facilement la réduire sous la forme $N' = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{array}\right) \begin{array}{c} \bar{a} \\ \bar{b} \end{array}$ d'où $q = \dfrac{1+\bar{a}x}{1+\bar{a}y}$.

Élimination de paramètres

Si N n'est pas de rang plein, alors un pré-traitement par symétries de type dilatation dans q permet de supposer N de rang plein.
Cela revient ici à manipuler $N = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 1 \end{array}\right)$.

Pré-traitement de N (2)

On suppose maintenant N de rang plein.

La matrice $\begin{pmatrix} N \\ \mathbb{1} \end{pmatrix}$ n'est pas nécessairement une base.

Exemple

Pour $q = \dfrac{a + ax}{b + by}$, on a $M = \begin{pmatrix} N \\ \mathbb{1} \end{pmatrix} = \begin{array}{c} \left(\begin{array}{cc|cc} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{array}\right) \end{array}\begin{array}{c} a \\ b \\ \end{array}$. On peut la réduire sous la forme $M' = \left(\begin{array}{cc|cc} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{array}\right)\begin{array}{c} \bar{a} \\ \bar{b} \\ \end{array}$ d'où $q = \dfrac{\bar{a} + \bar{a}x}{1 + y}$.

Élimination de paramètres

Si M n'est pas de rang plein, alors un pré-traitement dans q permet de supposer M de rang plein, revenant ici à manipuler $N = \left(\begin{array}{cc|cc} 1 & 1 & 0 & 0 \end{array}\right)$.

Sommaire

Idée

On suppose $M = \begin{pmatrix} N \\ \mathbb{1} \end{pmatrix}$ de rang plein (on a éliminé des paramètres).

Exemple

Pour $q = \dfrac{a + b^2 y}{a + bx}$, on applique CSB sur $M = \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{array}\right) \begin{array}{c} \bar{a} \\ \bar{b} \\ \end{array}$ et on obtient $M' = D.M = \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right)$ avec $D = \begin{pmatrix} 1 & 1 & -1 \\ -2 & -1 & 2 \\ 1 & 0 & 0 \end{pmatrix}$.

Principe

On veut maintenant éliminer une ligne artificielle de M' correspondant à $\mathbb{1}$ pour obtenir N'.
Pour obtenir le changement de variable C, il faut donc également extraire dans D (3×3) une sous-matrice C (2×2) inversible.

Algorithme de recherche de C et N'

L'algorithme suivant permet de réaliser cela.

Algorithm 1: getNewRepresentation(N)

Input: une base N de $\mathbb{Z}^{m \times n}$ (avec $m \leq n$) telle que $\begin{pmatrix} N \\ \mathbb{1} \end{pmatrix}$ soit une base

Output: matrice de changement de variable C et représentation fractionnaire N'

begin

$M = \begin{pmatrix} N \\ \mathbb{1} \end{pmatrix}$;

$M' \leftarrow CSB(M)$;

Trier les lignes de M' par nombre de non-zéros croissants ;

Calculer D inversible telle que $M' = D.M$ $(D \in \mathbb{Q}^{(m+1)\times(m+1)})$;

Extraire une sous-matrice C inversible dans les m premières colonnes de D $(C \in \mathbb{Q}^{m \times m})$;

Soit i l'indice de la ligne de D qui n'est pas utilisée pour déterminer C ;

N' est la sous-matrice de M' privée de sa i-ième ligne ;

Renvoyer C, N' ;

end

Algorithme de recherche sur un exemple (1)

Exemple

Pour $q = \dfrac{a+b^2y}{a+bx}$, on a $N = \begin{array}{c} \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{array}\right) \end{array} \begin{array}{c} a \\ b \end{array}$.

Avec l'algorithme getNewRepresentation

On calcule $M' = CSB\begin{pmatrix} N \\ \mathbb{1} \end{pmatrix}$. Après tri, on obtient $M' = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{array}\right)$.

On calcule $D = \begin{pmatrix} 1 & 1 & -1 \\ -2 & -1 & 2 \\ 1 & 0 & 0 \end{pmatrix}$ et on extrait $C = \begin{array}{cc} a & b \\ \multicolumn{2}{c}{\begin{pmatrix} 1 & 1 \\ -2 & -1 \end{pmatrix}} \end{array} \begin{array}{c} \bar{a} \\ \bar{b} \end{array}$ inversible.

La 3ème ligne de D n'a pas été utilisée pour déterminer C, on a alors $N' = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \begin{array}{c} \bar{a} \\ \bar{b} \end{array}$.

Algorithme de recherche sur un exemple (2)

Exemple

Pour $q = \dfrac{a + b^2 y}{a + bx}$, on a $N = \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{array}\right) \begin{array}{c} a \\ b \end{array}$.

Changements de variables et représentation simplifiée de q

- C fournit les changements de variables $a = \dfrac{\bar{a}}{\bar{b}^2}$ et $b = \dfrac{\bar{a}}{\bar{b}}$,
- N' fournit une représentation simplifiée de $q = \dfrac{1 + \bar{a}y}{1 + \bar{b}x}$.

Sommaire

Simplification d'une somme de fractions rationnelles

Généralisation de l'algorithme sur une somme de fractions rationnelles

On peut généraliser notre algorithme à une somme de fractions rationnelles, que l'on ne veut pas réduire au même dénominateur.

Raisons de de pas mettre au même dénominateur les fractions

- Grossissement des données après mise au même dénominateur,
- on veut conserver exactement les mêmes monômes en sortie qu'en entrée pour la somme de fractions.

Représentation matricielle d'une somme de fractions

Exemple

Soit (E) : $\dfrac{a + aby}{a + bx} + abcy + \dfrac{abc + ay}{ax} = 0.$

Représentation matricielle de (E)

On représente (E) par la matrice

$$N = \begin{array}{cc} \left(\begin{array}{cc|cc||c|c||cc|c} 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{array}\right) & \begin{array}{c} a \\ b \\ c \end{array} \\ \begin{array}{ccccccccc} 1 & y & 1 & x & y & 1 & 1 & y & x \end{array} & \end{array}$$

Simplification d'une somme de fractions

Gestion de la non-unicité de plusieurs fractions

Chaque fraction rationnelle est inchangée modulo les lignes en magenta ajoutées ci-dessous.

Simplification de N

Pour simplifier N, on manipulera dans notre algorithme la matrice :

$$M = \begin{pmatrix} N \\ L_1 \\ L_2 \\ L_3 \end{pmatrix} = \begin{array}{cc} \left(\begin{array}{cc|cc||c|c||cc|c} 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{array}\right) & \begin{array}{c} a \\ b \\ c \\ \\ \\ \\ \end{array} \\ \begin{array}{ccccccccc} 1 & y & 1 & x & y & 1 & 1 & y & x \end{array} & \end{array}$$

Application de l'algorithme sur une somme de fractions

Exemple

Soit (E) : $\dfrac{a + aby}{a + bx} + abcy + \dfrac{abc + ay}{ax} = 0.$
Nombre de paramètres dans (E) = **13**.

Après usage de l'algorithme

On trouve les changements de variables $a = \bar{a}$, $b = \frac{\bar{a}}{\bar{b}}$ et $c = \frac{\bar{b}\bar{c}}{\bar{a}}$.
Notre somme de fractions rationnelles devient :

$$(E') : \frac{\bar{a} + \bar{a}y}{\bar{b} + x} + \bar{a}\bar{c}y + \frac{\bar{c} + y}{x} = 0.$$

Nombre de paramètres dans (E') = **6**.

Généralisation à des systèmes d'équations

Généralisation

On procède de la même façon pour un système que pour une somme de fractions.
On peut séparer les équations par 3 | dans la représentation matricielle d'un système.

MABSys

L'algorithme étendu pour des systèmes d'équations contenant des fractions rationnelles peut être incorporé dans le logiciel MABSys (développé dans l'équipe) [Lemaire, Ürgüplü – ANB2010].

Application sur le système de BLLM08 (1)

Le système suivant est tiré de [Boulier, Lefranc, Lemaire, Morant – Algebraic Biology 2008].

Système d'équations obtenu avec MABSys dans BLLM08

$$(S) : \begin{cases} G'(t) = 1 - G(t) - k_1 k_2 k_3 \alpha P_1(t)^4 G(t) \\ M'(t) = -\delta_M M(t) + G(t) + \rho_b - \rho_b G(t) \\ P_1'(t) = \dfrac{4 - 4G(t) + \beta M(t) - k_1 k_2 k_3 \alpha \delta_P P_1(t)^4 G(t)}{16 k_1 k_2 k_3 P_1(t)^3 + 9 k_1 k_2 P_1(t)^2 + 4 k_1 P_1(t) + 1} \end{cases}$$

On a envie d'effectuer manuellement les changements de variables $\bar{k_1} = k_1$, $\bar{k_2} = k_1 k_2$ et $\bar{k_3} = k_1 k_2 k_3$.

Application sur le système de BLLM08 (2)

Simplification automatique avec notre algorithme (BLLM08)

$$(S) : \begin{cases} G'(t) = 1 - G(t) - k_1 k_2 k_3 \alpha P_1(t)^4 G(t) \\ M'(t) = -\delta_M M(t) + G(t) + \rho_b - \rho_b G(t) \\ P_1'(t) = \dfrac{4 - 4G(t) + \beta M(t) - k_1 k_2 k_3 \alpha \delta_P P_1(t)^4 G(t)}{16 k_1 k_2 k_3 P_1(t)^3 + 9 k_1 k_2 P_1(t)^2 + 4 k_1 P_1(t) + 1} \end{cases}$$

est simplifié grâce à notre algorithme en :

$$(S') : \begin{cases} G'(t) = 1 - G(t) - \bar{k}_3 \alpha P_1(t)^4 G(t) \\ M'(t) = -\delta_M M(t) + G(t) + \rho_b - \rho_b G(t) \\ P_1'(t) = \dfrac{4 - 4G(t) + \beta M(t) - \bar{k}_3 \alpha \delta_P P_1(t)^4 G(t)}{16 \bar{k}_3 P_1(t)^3 + 9 \bar{k}_2 P_1(t)^2 + 4 \bar{k}_1 P_1(t) + 1} \end{cases}$$

Conclusion

Conclusion

Notre algorithme permet d'automatiser une simplification de système d'équations qui était faite à la main, et qui n'est pas faisable par étude de symétries de type dilatation (car il n'y en a pas dans (S)).

Question time