Cálculo en paralelo de la homología cúbica con gridMathematica

Jónathan Heras, Luis Javier Hernández, María Teresa Rivas, Julio Rubio, Eduardo Sáenz de Cabezón

Departamento de Matemáticas y Computación
Universidad de La Rioja
Spain

III Congreso Mathematica en España

Índice

Índice

Nociones previas

Intervalo elemental:

$$I = [k, k+1] \qquad I = [k, k] \qquad k \in \mathbb{Z}$$

Nociones previas

Intervalo elemental:

$$I = [k, k+1] \qquad I = [k, k] \qquad k \in \mathbb{Z}$$

Cubo elemental:

$$Q = I_1 \times \ldots \times I_n$$

Nociones previas

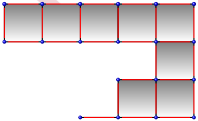
Intervalo elemental:

$$I = [k, k+1] \qquad I = [k, k] \qquad k \in \mathbb{Z}$$

Cubo elemental:

$$Q = I_1 \times \ldots \times I_n$$

Complejo Cúbico:



Nociones previas

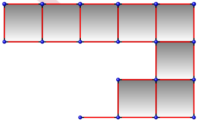
Intervalo elemental:

$$I = [k, k+1] \qquad I = [k, k] \qquad k \in \mathbb{Z}$$

Cubo elemental:

$$Q = I_1 \times \ldots \times I_n$$

Complejo Cúbico:



Dada la aplicación borde:

$$\partial_k : C_k \longrightarrow C_{k-1}$$

Calculamos su matriz de incidencia
Cálculo de homología:

Basado en diagonalización de matrices

Nociones previas

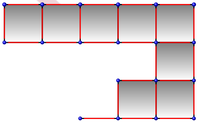
Intervalo elemental:

$$I = [k, k+1] \qquad I = [k, k] \qquad k \in \mathbb{Z}$$

Cubo elemental:

$$Q = I_1 \times \ldots \times I_n$$

Complejo Cúbico:



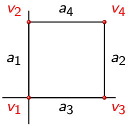
Dada la aplicación borde:

$$\partial_k : C_k \longrightarrow C_{k-1}$$

Calculamos su matriz de incidencia
Cálculo de homología:

Basado en diagonalización de matrices

Ejemplo:



Matriz Incidencia:

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} a_1 & a_2 & a_3 & a_4 \end{array} \\ \begin{pmatrix} -1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \end{array} = M_1$$

Nociones previas

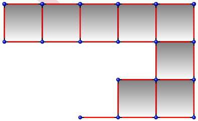
Intervalo elemental:

$$I = [k, k+1] \qquad I = [k, k] \qquad k \in \mathbb{Z}$$

Cubo elemental:

$$Q = I_1 \times \ldots \times I_n$$

Complejo Cúbico:



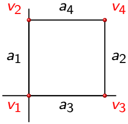
Dada la aplicación borde:

$$\partial_k : C_k \to C_{k-1}$$

Calculamos su matriz de incidencia
Cálculo de homología:

Basado en diagonalización de matrices

Ejemplo:



Matriz Incidencia:

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} a_1 & a_2 & a_3 & a_4 \end{array} \\ \begin{pmatrix} -1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \end{array} = M_1$$

Utilidad:

Minería de Datos

Índice

Algoritmos de cálculo de la homología cúbica

Algoritmos básicos:

- *RowExchange*
- *RowAdd*
- *RowMultiply*
- *ColumnExchange*
- *ColumnAdd*
- *ColumnMultiply*

Algoritmos de cálculo de la homología cúbica

Algoritmos básicos:

- *RowExchange*
- *RowAdd*
- *RowMultiply*
- *ColumnExchange*
- *ColumnAdd*
- *ColumnMultiply*

Algoritmos Importantes:

- *RowEchelon*
- *Kernel-Image*
- SmithForm
- *Solve*

RowEchelon

RowEchelon:

Calcula la forma escalada por filas de la matriz dada

- Matriz triangular superior
- Dadas dos filas consecutivas r_i y r_{i+1}:
  - Si $r_{i+1} \neq 0 \Rightarrow r_i \neq 0$
  - Posición pivote r_{i+1} > Posición pivote r_i

RowEchelon

RowEchelon:

Calcula la forma escalada por filas de la matriz dada

- Matriz triangular superior
- Dadas dos filas consecutivas r_i y r_{i+1}:
  - Si $r_{i+1} \neq 0 \Rightarrow r_i \neq 0$
  - Posición pivote $r_{i+1} >$ Posición pivote r_i

Ejemplo:

$$A = \begin{pmatrix} 3 & 2 & 1 & 4 \\ 2 & 3 & 1 & -1 \\ 4 & 4 & -2 & -2 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & -1 & 0 & 5 \\ 0 & -1 & -11 & -11 \\ 0 & 0 & 18 & 22 \end{pmatrix}$$

Kernel-Image

Kernel-Image:
Dada una matriz A calcula:

- Una base del Kernel de A
- Una base de la Imagen de A

Kernel-Image

Kernel-Image:

Dada una matriz A calcula:

- Una base del Kernel de A
- Una base de la Imagen de A

Ejemplo:

$$A = \begin{pmatrix} 0 & 2 & 2 \\ 1 & 0 & -1 \\ 3 & 4 & 1 \\ 5 & 3 & -2 \end{pmatrix}$$

Kernel: $\{\{1\}, \{-1\}, \{1\}\}$
Image: $\{\{2, 0\}, \{0, 1\}, \{4, 3\}, \{3, 5\}\}$

SmithForm

SmithForm:
Dada una matriz A calcula:

Matriz diagonal de la forma

$$\left(\begin{array}{ccc|c} B[1,1] & & 0 & \\ & \ddots & & 0 \\ 0 & & B[k,k] & \\ \hline & 0 & & B[k+1:m, k+1:n] \end{array}\right)$$

SmithForm

SmithForm:

Dada una matriz A calcula:

Matriz diagonal de la forma

$$\left(\begin{array}{ccc|c} B[1,1] & & 0 & \\ & \ddots & & 0 \\ 0 & & B[k,k] & \\ \hline & 0 & & B[k+1:m, k+1:n] \end{array}\right)$$

Ejemplo:

$$A = \begin{pmatrix} 3 & 2 & 3 \\ 0 & 2 & 0 \\ 2 & 2 & 2 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Solve

Solve:

Resuelve sistemas de ecuaciones:

- Componentes enteras
- Solución enteros

Solve

Solve:

Resuelve sistemas de ecuaciones:

- Componentes enteras
- Solución enteros

Ejemplo:

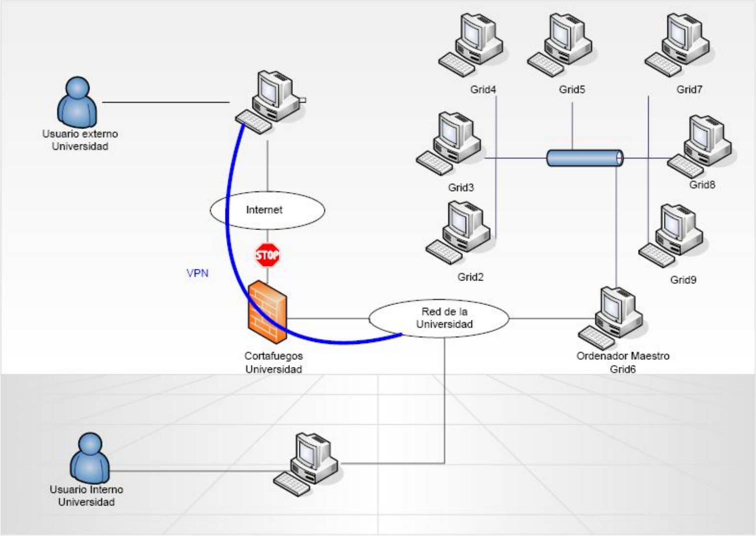
$$\left\{ \begin{array}{c} x + y + z = 6 \\ x + 2y + 3z = 14 \\ x + z = 4 \end{array} \right. \Rightarrow \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 6 \\ 14 \\ 4 \end{pmatrix}$$

Solución: $\{1, 2, 3\}$

Índice

El entorno de trabajo

Nuestro Grid:



El entorno de trabajo

Programas desarrollo:

- Mathematica v6.0
- gridMathematica v2.0
  - Linux
  - Mathematica v5.2
- Wolfram Workbench v1.1

Programas adicionales:

- Cygwin
- SSH Client

Índice

Aproximaciones para paralelizar el algoritmo

Ideas Fundamentales:

- Ejecución simultanea en vez de ejecución secuencial
- Independencia

Aproximaciones para paralelizar el algoritmo

Ideas Fundamentales:

- Ejecución simultanea en vez de ejecución secuencial
- Independencia

Formas de Paralelizar:

- ParallelEvaluate
- Paralelización a bajo nivel
- MinNonZero
- MejorandoSmith
- Smith en listas

Paralelización a bajo nivel

En Local:

$$\begin{pmatrix} 1 & 2 & 3 \\ -3 & -4 & 10 \\ 12 & -4 & -8 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\searrow 1 \qquad \downarrow 2 \;\; (\textit{fin } 1) \qquad \swarrow 3 \;\; (\textit{fin } 2)$$

Paralelización a bajo nivel

En Local:

$$\begin{pmatrix} 1 & 2 & 3 \\ -3 & -4 & 10 \\ 12 & -4 & -8 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$\searrow 1$ $\quad \downarrow 2 \ \ (\mathit{fin}\ 1)$ $\quad \swarrow 3 \ \ (\mathit{fin}\ 2)$

En Paralelo:

$$\begin{pmatrix} 1 & 2 & 3 \\ -3 & -4 & 10 \\ 12 & -4 & -8 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$\downarrow \quad \downarrow \quad \downarrow$



$\searrow \quad \downarrow \quad \swarrow$

Resultado

MinNonZero

$$A = \begin{pmatrix} 8 & 7 & 3 & 6 \\ -3 & -4 & 10 & -2 \\ 12 & -4 & -8 & 9 \\ -4 & 11 & 5 & 3 \end{pmatrix}$$

MinNonZero

$$A = \begin{pmatrix} 8 & 7 & 3 & 6 \\ -3 & -4 & 10 & -2 \\ 12 & -4 & -8 & 9 \\ -4 & 11 & 5 & 3 \end{pmatrix}$$

Distribuimos las filas:

$$\begin{array}{cccc} (8 \;\; 7 \;\; 3 \;\; 6) & (-3 \;\; -4 \;\; 10 \;\; -2) & (12 \;\; -4 \;\; -8 \;\; 9) & (-4 \;\; 11 \;\; 5 \;\; 3) \\ \downarrow & \downarrow & \downarrow & \downarrow \end{array}$$

MinNonZero

$$A = \begin{pmatrix} 8 & 7 & 3 & 6 \\ -3 & -4 & 10 & -2 \\ 12 & -4 & -8 & 9 \\ -4 & 11 & 5 & 3 \end{pmatrix}$$

Distribuimos las filas:

$$(8 \;\; 7 \;\; 3 \;\; 6) \qquad (-3 \;\; -4 \;\; 10 \;\; -2) \qquad (12 \;\; -4 \;\; -8 \;\; 9) \qquad (-4 \;\; 11 \;\; 5 \;\; 3)$$

$$\downarrow \qquad\qquad \downarrow \qquad\qquad \downarrow \qquad\qquad \downarrow$$

Obtenemos $(3 \;\; -2 \;\; -4 \;\; 3)$

Mejorando Smith

$$A = \begin{pmatrix} \color{red}{1} & 7 & 8 & 6 \\ 3 & 4 & 24 & -2 \\ 2 & 4 & 8 & 9 \\ -4 & 2 & 1 & 3 \end{pmatrix}$$

Mejorando Smith

$$A = \begin{pmatrix} \color{red}{1} & 7 & 8 & 6 \\ 3 & 4 & 24 & -2 \\ 2 & 4 & 8 & 9 \\ -4 & 2 & 1 & 3 \end{pmatrix} \xrightarrow{PartSmithForm} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -17 & 0 & -20 \\ 0 & -10 & -8 & -3 \\ 0 & 30 & 33 & 27 \end{pmatrix}$$

Mejorando Smith

$$A = \begin{pmatrix} \color{red}{1} & 7 & 8 & 6 \\ 3 & 4 & 24 & -2 \\ 2 & 4 & 8 & 9 \\ -4 & 2 & 1 & 3 \end{pmatrix} \xrightarrow{\textit{PartSmithForm}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -17 & 0 & -20 \\ 0 & -10 & -8 & -3 \\ 0 & 30 & 33 & 27 \end{pmatrix}$$

La matriz se divide en

$$\begin{pmatrix} 1 & 7 & 8 & 6 \\ 3 & 4 & 24 & -2 \end{pmatrix} \qquad \begin{pmatrix} 1 & 7 & 8 & 6 \\ 2 & 4 & 8 & 9 \end{pmatrix} \qquad \begin{pmatrix} 1 & 7 & 8 & 6 \\ -4 & 2 & 1 & 3 \end{pmatrix}$$

Mejorando Smith

$$A = \begin{pmatrix} \color{red}{1} & 7 & 8 & 6 \\ 3 & 4 & 24 & -2 \\ 2 & 4 & 8 & 9 \\ -4 & 2 & 1 & 3 \end{pmatrix} \xrightarrow{PartSmithForm} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -17 & 0 & -20 \\ 0 & -10 & -8 & -3 \\ 0 & 30 & 33 & 27 \end{pmatrix}$$

La matriz se divide en

$$\begin{pmatrix} 1 & 7 & 8 & 6 \\ 3 & 4 & 24 & -2 \end{pmatrix} \quad \begin{pmatrix} 1 & 7 & 8 & 6 \\ 2 & 4 & 8 & 9 \end{pmatrix} \quad \begin{pmatrix} 1 & 7 & 8 & 6 \\ -4 & 2 & 1 & 3 \end{pmatrix}$$

De cada una obtenemos

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -17 & 0 & -20 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -10 & -8 & -3 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 30 & 33 & 27 \end{pmatrix}$$

Smith en Listas

$$C_k(X) \xrightarrow{\partial_k} C_{k-1}(X) \xrightarrow{\partial_{k-1}} \ldots \to C_1(X) \xrightarrow{\partial_1} C_0(X) \to 0.$$

Smith en Listas

$$C_k(X) \xrightarrow{\partial_k} C_{k-1}(X) \xrightarrow{\partial_{k-1}} \ldots \rightarrow C_1(X) \xrightarrow{\partial_1} C_0(X) \rightarrow 0.$$

Local
Lista de matrices:

A_1	A_2	$\cdots$	A_k

Smith en Listas

$$C_k(X) \xrightarrow{\partial_k} C_{k-1}(X) \xrightarrow{\partial_{k-1}} \ldots \rightarrow C_1(X) \xrightarrow{\partial_1} C_0(X) \rightarrow 0.$$

Local
Lista de matrices:

A_1	A_2	$\cdots$	A_k

La dividimos en:

A_1	A_3	$\cdots$	A_{2k+1}

A_2	A_4	$\cdots$	A_{2k}

Smith en Listas

$$C_k(X) \xrightarrow{\partial_k} C_{k-1}(X) \xrightarrow{\partial_{k-1}} \ldots \to C_1(X) \xrightarrow{\partial_1} C_0(X) \to 0.$$

Local
Lista de matrices:

A_1	A_2	$\cdots$	A_k

La dividimos en:

A_1	A_3	$\cdots$	A_{2k+1}

A_2	A_4	$\cdots$	A_{2k}

Cálculo SmithForm secuencial

Smith en Listas

$$C_k(X) \xrightarrow{\partial_k} C_{k-1}(X) \xrightarrow{\partial_{k-1}} \ldots \to C_1(X) \xrightarrow{\partial_1} C_0(X) \to 0.$$

Local
Lista de matrices:

A_1	A_2	$\cdots$	A_k

La dividimos en:

A_1	A_3	$\cdots$	A_{2k+1}

A_2	A_4	$\cdots$	A_{2k}

Cálculo SmithForm secuencial

Paralelo:
Lista de matrices:

A_1	A_2	$\cdots$	A_n

Smith en Listas

$$C_k(X) \xrightarrow{\partial_k} C_{k-1}(X) \xrightarrow{\partial_{k-1}} \ldots \to C_1(X) \xrightarrow{\partial_1} C_0(X) \to 0.$$

Local
Lista de matrices:

A_1	A_2	$\cdots$	A_k

La dividimos en:

A_1	A_3	$\cdots$	A_{2k+1}

A_2	A_4	$\cdots$	A_{2k}

Cálculo SmithForm secuencial

Paralelo:
Lista de matrices:

A_1	A_2	$\cdots$	A_n

La dividimos en:

A_1	A_3	$\cdots$	A_{2k+1}

A_2	A_4	$\cdots$	A_{2k}

Smith en Listas

$$C_k(X) \xrightarrow{\partial_k} C_{k-1}(X) \xrightarrow{\partial_{k-1}} \ldots \rightarrow C_1(X) \xrightarrow{\partial_1} C_0(X) \rightarrow 0.$$

Local
Lista de matrices:

A_1	A_2	$\cdots$	A_k

La dividimos en:

A_1	A_3	$\cdots$	A_{2k+1}

A_2	A_4	$\cdots$	A_{2k}

Cálculo SmithForm secuencial

Paralelo:
Lista de matrices:

A_1	A_2	$\cdots$	A_n

La dividimos en:

A_1	A_3	$\cdots$	A_{2k+1}

A_2	A_4	$\cdots$	A_{2k}

Cálculo SmithForm se distribuye por el grid

Índice

Resultados experimentales: ParallelEvaluate

algoritmo \ matriz	5×5	10×10	15×15	30×30	50×50
SmithForm con 6	0,3568	3,4022	11,2505	146,0604	358,3147
SmithForm con 4	0,4854	3,3742	11,6476	190,4567	597,1856
SmithForm con 2	0,4170	2,9982	11,0818	195,7012	698,5674
SmithForm local	0,1715	0,8581	2,8241	29,3964	228,3113

Tabla: ParallelEvaluate aplicado a SmithForm

Empeora mas de un 50 %

Resultados experimentales: Paralelización a bajo nivel

matriz / Versión	5×5	10×10	15×15	30×30	50×50
Versión Original	1,6006	5,3456	17,7105	240,4157	531,4841
Versión Paralelo	1,1600	4,3157	13,5878	180,2474	451,4861

Tabla: Send-Receive en *SmithForm*

Mejora aproximadamente un 30 %

Resultados experimentales: MinNonZero

matriz / Ejecución	10×10	25×25	50×50	100×100	200×200
Local	0,0116	0,0525	0,1460	0,4849	1,7776
Paralelo	0,0204	0,0682	0,2586	1,4369	12,8041

Tabla: Resultados Paralelizando *MinNonZero*

Empeora mas de un 70 %

Resultados experimentales: Mejorando Smith

modo \ matriz	5×5	10×10	15×15	20×20	30×30	50×50
Local	0,186	0,940	2,509	5,142	21,49	107,44
Paralelo	0,306	0,973	1,959	3,799	18,24	89,45

Tabla: *SmithParalelo* vs *Smith*

En matrices pequeñas: empeora 20 %

En matrices grandes: mejora 30 %

Resultados experimentales: Smith en listas

Ejecución \ matriz	5 × 5	10 × 10	15 × 15	30 × 30	50 × 50
Local	0,8481	4,5626	14,9649	110,3033	$*$
Paralelo	0,1743	0,7611	2,3116	35,5071	212,1252

Tabla: Smith en listas de tamaño 5

Ejecución \ matriz	5 × 5	10 × 10	15 × 15	30 × 30
Local	1,803	7,799	23,261	193,3125
Paralelo	1,4549	4,4061	8,6070	78,7754

Tabla: Smith en listas de tamaño 10

Ejecución \ matriz	5 × 5	10 × 10	15 × 15	30 × 30
Local	3,9532	18,2269	50,8457	318,8695
Paralelo	2,4304	12,8908	34,8918	247,6956

Tabla: Smith en listas de tamaño 15

Mejora 40-80 %

Índice

Trabajo Adicional: GUIKit

GUIKit:

¿Qué es?

- Paquete para crear interfaces gráficas con Mathematica

¿Para qué?

- Construir pequeña interfaz

¿Por qué?

- Evita utilizar J/Link

Trabajo Adicional: GUIKit

GUIKit:

¿Qué es?

- Paquete para crear interfaces gráficas con Mathematica

¿Para qué?

- Construir pequeña interfaz

¿Por qué?

- Evita utilizar J/Link

Nuestra Interfaz:

¿Qué es?

- Interfaz gráfica con funciones más relevantes

¿Por qué?

- Transparencia
- Desarrollo de algo gráfico

Índice

Conclusiones y Trabajo Futuro

Conclusiones:

Paralelizar:

- Potente herramienta
- No siempre positiva

Trabajo Futuro:

- Tecnología Wolfram: webMathematica, J/Link
- Matrices dispersas
- Nuevas formas de paralelizar
- Colapsos

Cálculo en paralelo de la homología cúbica con gridMathematica

Jónathan Heras, Luis Javier Hernández, María Teresa Rivas, Julio Rubio, Eduardo Sáenz de Cabezón

Departamento de Matemáticas y Computación
Universidad de La Rioja
Spain

III Congreso Mathematica en España