Herramientas de cálculo simbólico para la Topología Algebraica: homología efectiva

Jónathan Heras,
V. Pascual, J. Rubio y F. Sergeraert

Departamento de Matemáticas y Computación
Universidad de La Rioja
Spain

22 de Mayo de 2010

Índice

Índice

Algebra Homológica

Definición

Un complejo de cadenas C_* es un par de secuencias $C_* = (C_q, d_q)_{q\in\mathbb{Z}}$ donde:

- *para cada $q \in \mathbb{Z}$, la componente C_q es un R-modulo*
- *para cada $q \in \mathbb{Z}$, la componente d_q es un morfismo de módulos $d_q : C_q \to C_{q-1}$, aplicación diferencial*
- *Para cada $q \in \mathbb{Z}$, la composición $d_q d_{q+1} = 0$*

Algebra Homológica

Definición

Un complejo de cadenas C_* es un par de secuencias $C_* = (C_q, d_q)_{q\in\mathbb{Z}}$ donde:

- *para cada $q \in \mathbb{Z}$, la componente C_q es un R-modulo*
- *para cada $q \in \mathbb{Z}$, la componente d_q es un morfismo de módulos $d_q : C_q \to C_{q-1}$, aplicación diferencial*
- *Para cada $q \in \mathbb{Z}$, la composición $d_q d_{q+1} = 0$*

Definición

Sea $C_* = (C_n, d_n)_{n\in\mathbb{Z}}$ un complejo de cadenas de R-módulos. Para cada grado $n \in \mathbb{Z}$ se define el n-ésimo grupo de homología de C_*:

$$H_n(C_*) = \frac{Ker\ d_n}{Im\ d_{n+1}}$$

Algebra Homológica

Definición

Un complejo de cadenas C_* es un par de secuencias $C_* = (C_q, d_q)_{q\in\mathbb{Z}}$ donde:

- *para cada $q \in \mathbb{Z}$, la componente C_q es un R-modulo*
- *para cada $q \in \mathbb{Z}$, la componente d_q es un morfismo de módulos $d_q : C_q \to C_{q-1}$, aplicación diferencial*
- *Para cada $q \in \mathbb{Z}$, la composición $d_q d_{q+1} = 0$*

Definición

Sea $C_* = (C_n, d_n)_{n\in\mathbb{Z}}$ un complejo de cadenas de R-módulos. Para cada grado $n \in \mathbb{Z}$ se define el n-ésimo grupo de homología de C_*:

$$H_n(C_*) = \frac{Ker\ d_n}{Im\ d_{n+1}}$$

Definición

Sean $C_* = (C_n, d_n)_{n\in\mathbb{Z}}$ y $D_* = (D_n, \widehat{d}_n)_{n\in\mathbb{Z}}$ complejos de cadenas de R-módulos. Un morfismo de complejos de cadenas $f : C_* \to D_*$ es una colección de morfismos $f = \{f_q : C_q \to D_q\}_q$ satisfaciendo que $f_{q-1}d_q = \widehat{d}_q f_q$

Topología Simplicial

Definición

Un complejo simplicial C es un conjunto finito de símplices satisfaciendo:

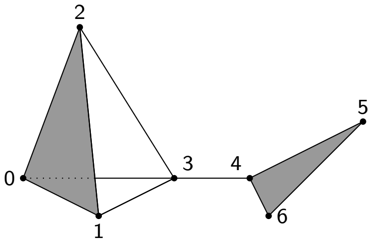
- *si σ_n es un simplex de C, y τ_p es una cara de σ_n, entonces τ_p está en C;*
- *si σ_n y τ_p son símplices de C, entonces $\sigma_n \cap \tau_p$ es una cara común de σ_n y τ_p.*

Topología Simplicial

Definición

Un complejo simplicial C es un conjunto finito de símplices satisfaciendo:

- *si σ_n es un simplex de C, y τ_p es una cara de σ_n, entonces τ_p está en C;*
- *si σ_n y τ_p son símplices de C, entonces $\sigma_n \cap \tau_p$ es una cara común de σ_n y τ_p.*



Topología Simplicial

Definición

Un conjunto simplicial K, es la unión $K = \bigcup_{q\geq 0} K^q$, donde los K^q son conjuntos disjuntos, junto a las funciones:

$$\begin{array}{lll} \partial_i^q : K^q \to K^{q-1}, & q > 0, & i = 0, \ldots, q, \\ \eta_i^q : K^q \to K^{q+1}, & q \geq 0, & i = 0, \ldots, q, \end{array}$$

sujetas a las relaciones:

$$\begin{array}{llclcl} (1) & \partial_i^{q-1}\partial_j^q & = & \partial_{j-1}^{q-1}\partial_i^q & \mathit{if} & i < j, \\ (2) & \eta_i^{q+1}\eta_j^q & = & \eta_j^{q+1}\eta_{i-1}^q & \mathit{if} & i > j, \\ (3) & \partial_i^{q+1}\eta_j^q & = & \eta_{j-1}^{q-1}\partial_i^q & \mathit{if} & i < j, \\ (4) & \partial_i^{q+1}\eta_i^q & = & \mathit{identity} & = & \partial_{i+1}^{q+1}\eta_i^q, \\ (5) & \partial_i^{q+1}\eta_j^q & = & \eta_j^{q-1}\partial_{i-1}^q & \mathit{if} & i > j+1, \end{array}$$

Topología Simplicial

Definición

Sea C un complejo simplicial. Entonces el conjunto simplicial $K(C)$ canónicamente asociado con C se define del siguiente modo. El conjunto $K^n(C)$ de n-símplices es el conjunto de símplices de cardinalidad $n+1$ de C. Además, dado un símplice $\{v_0, \ldots, v_q\}$ los operadores cara y degeneración se definen:

$$
\begin{aligned}
\partial_i(\{v_0, \ldots, v_i, \ldots, v_q\}) &= \{v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_q\} \\
\eta_i(\{v_0, \ldots, v_i, \ldots, v_q\}) &= \{v_0, \ldots, v_i, v_i, \ldots, v_q\}
\end{aligned}
$$

Topología Simplicial

Definición

Sea K un conjunto simplicial, definimos el complejo de cadenas, $C_*(K) = (C_n(K), d_n)_{n\in\mathbb{N}}$, asociado con K del siguiente modo:

- *$C_n(K) = \mathbb{Z}[K^n]$ ese el $\mathbb{Z}$-módulo libre generado por K^n*
- *la aplicación diferencial $d_n : C_n(K) \to C_{n-1}(K)$ viene dado por:*

$$d_n(x) = \sum_{i=0}^{n} (-1)^i \partial_i(X) \quad para\ x \in K^n$$

Topología Simplicial

Definición

Sea K un conjunto simplicial, definimos el complejo de cadenas, $C_*(K) = (C_n(K), d_n)_{n\in\mathbb{N}}$, asociado con K del siguiente modo:

- *$C_n(K) = \mathbb{Z}[K^n]$ ese el $\mathbb{Z}$-módulo libre generado por K^n*
- *la aplicación diferencial $d_n : C_n(K) \rightarrow C_{n-1}(K)$ viene dado por:*

$$d_n(x) = \sum_{i=0}^{n} (-1)^i \partial_i(X) \quad para\ x \in K^n$$

Definición

Sea K un conjunto simplicial, el n-ésimo grupo de homología de K, $H_n(K)$, es el n-ésimo grupo de homología del complejo de cadenas $C_*(K)$

Homología efectiva

Definición

Un complejo de cadenas efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n \in \mathbb{N}}$, donde cada grupo C_n es finitamente generado y

- *un algoritmo devuelve una $\mathbb{Z}$-base en cada grado n*
- *un algoritmo proporciona las diferenciales d_n*

Homología efectiva

Definición

Un complejo de cadenas efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n\in\mathbb{N}}$, donde cada grupo C_n es finitamente generado y

- *un algoritmo devuelve una $\mathbb{Z}$-base en cada grado n*
- *un algoritmo proporciona las diferenciales d_n*

Si un complejo de cadenas $C_* = (C_n, d_n)_{n\in\mathbb{N}}$ es efectivo:

Homología efectiva

Definición

Un complejo de cadenas efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n \in \mathbb{N}}$, donde cada grupo C_n es finitamente generado y

- *un algoritmo devuelve una $\mathbb{Z}$-base en cada grado n*
- *un algoritmo proporciona las diferenciales d_n*

Si un complejo de cadenas $C_* = (C_n, d_n)_{n \in \mathbb{N}}$ es efectivo:

- diferenciales $d_n : C_n \rightarrow C_{n-1}$ expresadas como matrices finitas de enteros

Homología efectiva

Definición

Un complejo de cadenas efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n \in \mathbb{N}}$, donde cada grupo C_n es finitamente generado y

- *un algoritmo devuelve una $\mathbb{Z}$-base en cada grado n*
- *un algoritmo proporciona las diferenciales d_n*

Si un complejo de cadenas $C_* = (C_n, d_n)_{n \in \mathbb{N}}$ es efectivo:

- diferenciales $d_n : C_n \rightarrow C_{n-1}$ expresadas como matrices finitas de enteros
- posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}

Homología efectiva

Definición

Un complejo de cadenas efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n \in \mathbb{N}}$, donde cada grupo C_n es finitamente generado y

- *un algoritmo devuelve una $\mathbb{Z}$-base en cada grado n*
- *un algoritmo proporciona las diferenciales d_n*

Si un complejo de cadenas $C_* = (C_n, d_n)_{n \in \mathbb{N}}$ es efectivo:

- diferenciales $d_n : C_n \longrightarrow C_{n-1}$ expresadas como matrices finitas de enteros
- posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}
- posible calcular los grupos de homología

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n \in \mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n\in\mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

- no tenemos información global

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$*-módulos,* $C_* = (C_n, d_n)_{n\in\mathbb{N}}$*, donde cada grupo* C_n *está formado por un número infinito de generadores*

- no tenemos información global
- no es posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n \in \mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

- no tenemos información global
- no es posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}
- posible realizar cálculos locales, caras de un elemento

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n\in\mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

- no tenemos información global
- no es posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}
- posible realizar cálculos locales, caras de un elemento

Definición

Un objeto localmente efectivo es aquel en el que sólo se pueden realizar cálculos locales, no hay información global

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n\in\mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

- no tenemos información global
- no es posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}
- posible realizar cálculos locales, caras de un elemento

Definición

Un objeto localmente efectivo es aquel en el que sólo se pueden realizar cálculos locales, no hay información global

Ejemplo:

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n\in\mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

- no tenemos información global
- no es posible calcular los subgrupos $Ker\ d_n$ y $Im\ d_{n+1}$
- posible realizar cálculos locales, caras de un elemento

Definición

Un objeto localmente efectivo es aquel en el que sólo se pueden realizar cálculos locales, no hay información global

Ejemplo:

- Conjunto simplicial localmente efectivo

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n \in \mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

- no tenemos información global
- no es posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}
- posible realizar cálculos locales, caras de un elemento

Definición

Un objeto localmente efectivo es aquel en el que sólo se pueden realizar cálculos locales, no hay información global

Ejemplo:

- Conjunto simplicial localmente efectivo
- conjunto de n-símplices no necesariamente finito

Homología efectiva

Definición

Un complejo de cadenas localmente efectivo es un complejo de cadenas libre de $\mathbb{Z}$-módulos, $C_* = (C_n, d_n)_{n\in\mathbb{N}}$, donde cada grupo C_n está formado por un número infinito de generadores

- no tenemos información global
- no es posible calcular los subgrupos *Ker* d_n y *Im* d_{n+1}
- posible realizar cálculos locales, caras de un elemento

Definición

Un objeto localmente efectivo es aquel en el que sólo se pueden realizar cálculos locales, no hay información global

Ejemplo:

- Conjunto simplicial localmente efectivo
- conjunto de n-símplices no necesariamente finito
- podemos calcular las caras de los símplices

Homología efectiva

Definición

Una reducción ρ *entre dos complejos de cadenas* C_* *y* D_* *(denotada por* $\rho : C_* \Rightarrow D_*$*) es un triple* $\rho = (f, g, h)$

$$\begin{tikzcd} C_* \arrow[loop left, "h"] \arrow[r, bend left, "f"] & D_* \arrow[l, bend left, "g"] \end{tikzcd}$$

satisfaciendo las siguiente relaciones:

1) $fg = \mathrm{Id}_{D_*}$;
2) $d_C h + h d_C = \mathrm{Id}_{C_*} - gf$;
3) $fh = 0$; $\quad hg = 0$; $\quad hh = 0$.

Homología efectiva

Definición

Una reducción ρ entre dos complejos de cadenas C_* y D_* (denotada por $\rho : C_* \Rightarrow D_*$) es un triple $\rho = (f, g, h)$

$$\begin{tikzcd} C_* \arrow[loop left, "h"] \arrow[r, bend left, "f"] & D_* \arrow[l, bend left, "g"] \end{tikzcd}$$

satisfaciendo las siguiente relaciones:

1) $fg = \mathrm{Id}_{D_*}$;
2) $d_C h + h d_C = \mathrm{Id}_{C_*} - gf$;
3) $fh = 0$; $\quad hg = 0$; $\quad hh = 0$.

Si $C_* \Rightarrow D_*$, entonces $C_* \cong D_* \oplus A_*$, con A_* acíclico, por lo que $H_n(C_*) \cong H_n(D_*)$ para todo n.

Homología efectiva

Definición

Una reducción ρ entre dos complejos de cadenas C_* y D_* (denotada por $\rho : C_* \Rightarrow D_*$) es un triple $\rho = (f, g, h)$

$$\begin{tikzcd} C_* \arrow[loop left, "h"] \arrow[r, bend left, "f"] & D_* \arrow[l, bend left, "g"] \end{tikzcd}$$

satisfaciendo las siguiente relaciones:

1) $fg = \mathrm{Id}_{D_*}$;
2) $d_C h + h d_C = \mathrm{Id}_{C_*} - gf$;
3) $fh = 0$; $\quad hg = 0$; $\quad hh = 0$.

Si $C_* \Rightarrow D_*$, entonces $C_* \cong D_* \oplus A_*$, con A_* acíclico, por lo que $H_n(C_*) \cong H_n(D_*)$ para todo n.

Situación habitual:

- C_* localmente efectivo
- D_* efectivo
- reducción permite calcular homología de C_* a partir de la de D_*

Homología efectiva

Definición

Una equivalencia (fuerte de cadenas) ε *entre* C_* *y* D_**,* $\varepsilon : C_* \Lleftarrow\!\!\Rrightarrow D_*$*, es un triple* $\varepsilon = (B_*, \rho, \rho')$ *donde* B_* *es un complejo de cadenas,* $\rho : B_* \Rightarrow C_*$ *y* $\rho' : B_* \Rightarrow D_*$*.*

$$\begin{tikzcd} & B_* \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\ C_* & & D_* \end{tikzcd} \qquad \begin{tikzcd} & \frac{42}{30} \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\ \frac{14}{10} & & \frac{21}{15} \end{tikzcd}$$

Homología efectiva

Definición

Una equivalencia (fuerte de cadenas) ε *entre* C_* *y* D_*, $\varepsilon : C_* \Leftrightarrow D_*$, *es un triple* $\varepsilon = (B_*, \rho, \rho')$ *donde* B_* *es un complejo de cadenas,* $\rho : B_* \Rightarrow C_*$ *y* $\rho' : B_* \Rightarrow D_*$.

$$\begin{tikzcd} & B_* \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\ C_* & & D_* \end{tikzcd} \qquad \begin{tikzcd} & \frac{42}{30} \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\ \frac{14}{10} & & \frac{21}{15} \end{tikzcd}$$

Definición

Un objeto con homología efectiva es un cuádruple $(X, C_*(X), HC_*, \varepsilon)$ *donde:*

- X *es un objeto localmente efectivo*
- $C_*(X)$ *es un complejo de cadenas (localmente efectivo) canónicamente asociado con* X, *que permite estudiar la naturaleza homológica de* X
- HC_* *es un complejo de cadenas efectivo*
- ε *es una equivalencia* $\varepsilon : C_*(X) \Leftrightarrow HC_*$

Homología efectiva

Definición

Una equivalencia (fuerte de cadenas) ε *entre* C_* *y* D_*, $\varepsilon : C_* \Leftrightarrow D_*$, *es un triple* $\varepsilon = (B_*, \rho, \rho')$ *donde* B_* *es un complejo de cadenas,* $\rho : B_* \Rightarrow C_*$ *y* $\rho' : B_* \Rightarrow D_*$.

$$\begin{tikzcd} & B_* \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\ C_* & & D_* \end{tikzcd} \qquad \begin{tikzcd} & \frac{42}{30} \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\ \frac{14}{10} & & \frac{21}{15} \end{tikzcd}$$

Definición

Un objeto con homología efectiva es un cuádruple $(X, C_*(X), HC_*, \varepsilon)$ *donde:*

- X *es un objeto localmente efectivo*
- $C_*(X)$ *es un complejo de cadenas (localmente efectivo) canónicamente asociado con* X, *que permite estudiar la naturaleza homológica de* X
- HC_* *es un complejo de cadenas efectivo*
- ε *es una equivalencia* $\varepsilon : C_*(X) \Leftrightarrow HC_*$

Grupos de homología de X son isomorfos a los de HC_*

Homología Efectiva

Dado un complejo de cadenas C_*, ¿es posible determinar su homología efectiva?

Homología Efectiva

Dado un complejo de cadenas C_*, ¿es posible determinar su homología efectiva?

- Si C_* es efectivo, entonces podemos elegir la homología efectiva trivial

$$C_* \Lleftarrow C_* \Rrightarrow C_*$$

Homología Efectiva

Dado un complejo de cadenas C_*, ¿es posible determinar su homología efectiva?

- Si C_* es efectivo, entonces podemos elegir la homología efectiva trivial

$$C_* \twoheadleftarrow\!\!\Leftarrow C_* \Rrightarrow C_*$$

- En algunos casos, resultados teóricos proporcionan una equivalencia entre el complejo de cadenas C_* y un complejo de cadenas efectivo. Ejemplo: $C_*(K(\mathbb{Z},1)) \Rrightarrow C_*(S^1)$

Homología Efectiva

Dado un complejo de cadenas C_*, ¿es posible determinar su homología efectiva?

- Si C_* es efectivo, entonces podemos elegir la homología efectiva trivial
$$C_* \Lleftarrow C_* \Rrightarrow C_*$$
- En algunos casos, resultados teóricos proporcionan una equivalencia entre el complejo de cadenas C_* y un complejo de cadenas efectivo. Ejemplo: $C_*(K(\mathbb{Z},1)) \Rrightarrow C_*(S^1)$
-

$$\begin{array}{c} X_1, X_2, \ldots, X_n \\ \Big\downarrow{\scriptstyle \varphi} \\ X \end{array}$$

Homología Efectiva

Dado un complejo de cadenas C_*, ¿es posible determinar su homología efectiva?

- Si C_* es efectivo, entonces podemos elegir la homología efectiva trivial

$$C_* \Lleftarrow C_* \Rrightarrow C_*$$

- En algunos casos, resultados teóricos proporcionan una equivalencia entre el complejo de cadenas C_* y un complejo de cadenas efectivo. Ejemplo: $C_*(K(\mathbb{Z},1)) \Rrightarrow C_*(S^1)$
-

$$\begin{tikzcd}
X_1, X_2, \ldots, X_n \arrow[d, "\varphi"] & & X_1^{EH}, X_2^{EH}, \ldots, X_n^{EH} \\
X & &
\end{tikzcd}$$

Homología Efectiva

Dado un complejo de cadenas C_*, ¿es posible determinar su homología efectiva?

- Si C_* es efectivo, entonces podemos elegir la homología efectiva trivial

$$C_* \Lleftarrow C_* \Rrightarrow C_*$$

- En algunos casos, resultados teóricos proporcionan una equivalencia entre el complejo de cadenas C_* y un complejo de cadenas efectivo. Ejemplo: $C_*(K(\mathbb{Z},1)) \Rrightarrow C_*(S^1)$
-

$$\begin{tikzcd}
X_1, X_2, \ldots, X_n \arrow[d, "\varphi"] & X_1^{EH}, X_2^{EH}, \ldots, X_n^{EH} \arrow[d, "\varphi^{EH}"] \\
X & X^{EH}
\end{tikzcd}$$

Idea general

Técnica de la homología efectiva:

- objeto localmente efectivo

Idea general

Técnica de la homología efectiva:

- objeto localmente efectivo
- complejo de cadenas efectivo

Idea general

Técnica de la homología efectiva:

- objeto localmente efectivo
- complejo de cadenas efectivo
- equivalencia de cadenas

Idea general

Técnica de la homología efectiva:

- objeto localmente efectivo
- complejo de cadenas efectivo
- equivalencia de cadenas

J. Rubio and F. Sergeraert. *Constructive Algebraic Topology*, Bulletin des Sciences Mathématiques. 2002 (126) 389-412.

J. Rubio and F. Sergeraert. *Constructive Homological Algebra and Applications*, Lecture Notes Summer School on Mathematics, Algorithms, and Proofs. University of Genova, 2006.

Índice

Kenzo

- Kenzo
  - Sistema de Cálculo Simbólico dedicado a la Topología Algebraica

Kenzo

- Kenzo
  - Sistema de Cálculo Simbólico dedicado a la Topología Algebraica
  - Grupos de Homología no calculables por otros medios

Kenzo

- Kenzo
  - Sistema de Cálculo Simbólico dedicado a la Topología Algebraica
  - Grupos de Homología no calculables por otros medios
  - Paquete Common Lisp

Kenzo

- Kenzo
  - Sistema de Cálculo Simbólico dedicado a la Topología Algebraica
  - Grupos de Homología no calculables por otros medios
  - Paquete Common Lisp
  - Trabaja con las principales estructuras matemáticas usadas en Topología Algebraica

Kenzo y la Homología Efectiva

- El sistema Kenzo utiliza la noción de objeto con homología efectiva para calcular los grupos de homología de algunos espacios complicados

Kenzo y la Homología Efectiva

- El sistema Kenzo utiliza la noción de objeto con homología efectiva para calcular los grupos de homología de algunos espacios complicados
  - Si el objeto es efectivo, entonces sus grupos de homología pueden ser calculados mediante operaciones elementales sobre matrices de enteros

Kenzo y la Homología Efectiva

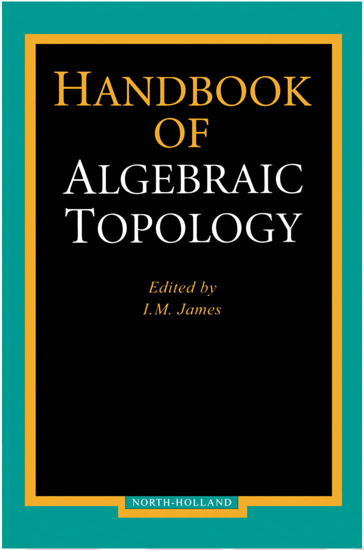
- El sistema Kenzo utiliza la noción de objeto con homología efectiva para calcular los grupos de homología de algunos espacios complicados
  - Si el objeto es efectivo, entonces sus grupos de homología pueden ser calculados mediante operaciones elementales sobre matrices de enteros
  - Si el objeto no es efectivo, el programa usa la homología efectiva

Kenzo y la Homología Efectiva

- El sistema Kenzo utiliza la noción de objeto con homología efectiva para calcular los grupos de homología de algunos espacios complicados
  - Si el objeto es efectivo, entonces sus grupos de homología pueden ser calculados mediante operaciones elementales sobre matrices de enteros
  - Si el objeto no es efectivo, el programa usa la homología efectiva
  - Demo

Cálculos complejos

Ejemplo típico extraído de:



Cálculos complejos

Capítulo 13:
Stable Homotopy and Iterated Loop Spaces
Gunnar Carlsson y James Milgram

CHAPTER 13

Stable Homotopy and Iterated Loop Spaces

Gunnar Carlsson
Department of Mathematics, Stanford University, Stanford, CA, USA
e-mail: gunnar@gauss.stanford.edu

R. James Milgram
Department of Mathematics, Stanford University, Stanford, CA, USA
e-mail: milgram@gauss.stanford.edu

Contents

Both authors were partially supported by grants from the N.S.F.

HANDBOOK OF ALGEBRAIC TOPOLOGY
Edited by I.M. James

505

Cálculos complejos

6. The structure of second loop spaces

In Section 5 we showed that for a connected CW complex with no one cells one may produce a CW complex, with cell complex given as the free monoid on generating cells, each in one dimension less than the corresponding cell of X, which is homotopy equivalent to ΩX. To go further one should study similar models for double loop spaces, and more generally for iterated loop spaces.

In principle this is direct. Assume X has no i-cells for $1 \leqslant i \leqslant n$ then we can iterate the Adams–Hilton construction of Section 5 and obtain a cell complex which represents $\Omega^n X$. However, the question of determining the boundaries of the cells is very difficult as we already saw with Adams' solution of the problem in the special case that X is a simplicial complex with $sk_1(X)$ collapsed to a point. It is possible to extend Adams' analysis to $\Omega^2 X$, but as we will see there will be severe difficulties with extending it to higher loop spaces except in the case where $X = \Sigma^n Y$.

Cálculos complejos

6. The structure of second loop spaces

In Section 5 we showed that for a connected CW complex with no one cells one may produce a CW complex, with cell complex given as the free monoid on generating cells, each in one dimension less than the corresponding cell of X, which is homotopy equivalent to ΩX. To go further one should study similar models for double loop spaces, and more generally for iterated loop spaces.

In principle this is direct. Assume X has no i-cells for $1 \leqslant i \leqslant n$ then we can iterate the Adams–Hilton construction of Section 5 and obtain a cell complex which represents $\Omega^n X$. However, the question of determining the boundaries of the cells is very difficult as we already saw with Adams' solution of the problem in the special case that X is a simplicial complex with $sk_1(X)$ collapsed to a point. It is possible to extend Adams' analysis to $\Omega^2 X$, but as we will see there will be severe difficulties with extending it to higher loop spaces except in the case where $X = \Sigma^n Y$.

Usando estos métodos no existe un algoritmo para calcular $H_*(\Omega^n X)$ para $n \geq 3$ excepto cuando X es una n-ésima suspension $X = \sum^n Y$

Cálculos complejos

6. The structure of second loop spaces

In Section 5 we showed that for a connected CW complex with no one cells one may produce a CW complex, with cell complex given as the free monoid on generating cells, each in one dimension less than the corresponding cell of X, which is homotopy equivalent to ΩX. To go further one should study similar models for double loop spaces, and more generally for iterated loop spaces.

In principle this is direct. Assume X has no i-cells for $1 \leqslant i \leqslant n$ then we can iterate the Adams–Hilton construction of Section 5 and obtain a cell complex which represents $\Omega^n X$. However, the question of determining the boundaries of the cells is very difficult as we already saw with Adams' solution of the problem in the special case that X is a simplicial complex with $sk_1(X)$ collapsed to a point. It is possible to extend Adams' analysis to $\Omega^2 X$, but as we will see there will be severe difficulties with extending it to higher loop spaces except in the case where $X = \Sigma^n Y$.

Usando estos métodos no existe un algoritmo para calcular $H_*(\Omega^n X)$ para $n \geq 3$ excepto cuando X es una n-ésima suspension $X = \sum^n Y$

Ejemplo típico: $H_*(\Omega^3(P^\infty\mathbb{R}/P^3\mathbb{R}))$

Demo

Nuevas Herramientas para Kenzo

Kenzo continua creciendo:

- Sucesiones espectrales

A. Romero, J. Rubio, and F. Sergeraert. *Computing spectral sequences*. Journal of Symbolic Computation, 41(10), 1059-1079, 2006.

Nuevas Herramientas para Kenzo

Kenzo continua creciendo:

- Sucesiones espectrales

A. Romero, J. Rubio, and F. Sergeraert. *Computing spectral sequences*. Journal of Symbolic Computation, 41(10), 1059-1079, 2006.

- Resoluciones

A. Romero, G. Ellis, and J. Rubio. *Interoperating between computer algebra systems: computing homology of groups with Kenzo and GAP*. 303–310, 2009.

Nuevas Herramientas para Kenzo

Kenzo continua creciendo:

- Sucesiones espectrales
  - A. Romero, J. Rubio, and F. Sergeraert. *Computing spectral sequences*. Journal of Symbolic Computation, 41(10), 1059-1079, 2006.
- Resoluciones
  - A. Romero, G. Ellis, and J. Rubio. *Interoperating between computer algebra systems: computing homology of groups with Kenzo and GAP*. 303–310, 2009.
- Complejos de Koszul
  - J. Rubio and F. Sergeraert. *Constructive Homological Algebra and Applications*, Lecture Notes Summer School on Mathematics, Algorithms, and Proofs. University of Genova, 2006.

Nuevas Herramientas para Kenzo

Kenzo continua creciendo:

- Sucesiones espectrales
  - A. Romero, J. Rubio, and F. Sergeraert. *Computing spectral sequences*. Journal of Symbolic Computation, 41(10), 1059-1079, 2006.
- Resoluciones
  - A. Romero, G. Ellis, and J. Rubio. *Interoperating between computer algebra systems: computing homology of groups with Kenzo and GAP*. 303–310, 2009.
- Complejos de Koszul
  - J. Rubio and F. Sergeraert. *Constructive Homological Algebra and Applications*, Lecture Notes Summer School on Mathematics, Algorithms, and Proofs. University of Genova, 2006.
- ...

Verificación del sistema Kenzo

Incrementar la fiabilidad del sistema:

- Isabelle/HOL

J. Aransay, C. Ballarin, J. Rubio, *A mechanized proof of the Basic Perturbation Lemma*. Journal of Automated Reasoning 40 (2008) 271-293

Verificación del sistema Kenzo

Incrementar la fiabilidad del sistema:

- Isabelle/HOL
  - J. Aransay, C. Ballarin, J. Rubio, *A mechanized proof of the Basic Perturbation Lemma*. Journal of Automated Reasoning 40 (2008) 271-293
- Coq
  - C. Domínguez, J. Rubio, *The effective homology of bicomplexes, formalized in Coq*. Preprint

Verificación del sistema Kenzo

Incrementar la fiabilidad del sistema:

- Isabelle/HOL
  - J. Aransay, C. Ballarin, J. Rubio, *A mechanized proof of the Basic Perturbation Lemma*. Journal of Automated Reasoning 40 (2008) 271-293
- Coq
  - C. Domínguez, J. Rubio, *The effective homology of bicomplexes, formalized in Coq*. Preprint
- ACL2
  - L. Lambán, F. J. Martín-Mateos, J. Rubio, J. L. Ruiz Reina, *When first order is enough: the case of Simplicial Topology*. Preprint

Índice

fKenzo

Kenzo es un sistema Common Lisp:

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad
- fKenzo

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad
- fKenzo
  - Una interfaz de usuario extensible:

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad
- fKenzo
  - Una interfaz de usuario extensible:
    - proporciona un acceso amigable al sistema Kenzo (**f**riendly **Kenzo**)

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad
- fKenzo
  - Una interfaz de usuario extensible:
    - proporciona un acceso amigable al sistema Kenzo (**f**riendly **Kenzo**)
    - permite conectar distintos sistemas de cálculo simbólico

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad
- fKenzo
  - Una interfaz de usuario extensible:
    - proporciona un acceso amigable al sistema Kenzo (**f**riendly **Kenzo**)
    - permite conectar distintos sistemas de cálculo simbólico
    - ...

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad
- fKenzo
  - Una interfaz de usuario extensible:
    - proporciona un acceso amigable al sistema Kenzo (**f**riendly **Kenzo**)
    - permite conectar distintos sistemas de cálculo simbólico
    - ...

fKenzo

Kenzo es un sistema Common Lisp:

- Problema:
  - lenguaje no muy extendido
  - usabilidad y accesibilidad
- fKenzo
  - Una interfaz de usuario extensible:
    - proporciona un acceso amigable al sistema Kenzo (**f**riendly **Kenzo**)
    - permite conectar distintos sistemas de cálculo simbólico
    - ...
- Demo

Kenzo vs fKenzo

- Ventajas de usar Kenzo:

Kenzo vs fKenzo

- Ventajas de usar Kenzo:
    - toda la funcionalidad disponible

Kenzo vs fKenzo

- Ventajas de usar Kenzo:
  - toda la funcionalidad disponible
  - proporciona más información

Kenzo vs fKenzo

- Ventajas de usar Kenzo:
  - toda la funcionalidad disponible
  - proporciona más información
- Ventajas de usar *fKenzo*:

Kenzo vs fKenzo

- Ventajas de usar Kenzo:
  - toda la funcionalidad disponible
  - proporciona más información
- Ventajas de usar *fKenzo*:
  - no es necesario conocer Common Lisp

Kenzo vs fKenzo

- Ventajas de usar Kenzo:
  - toda la funcionalidad disponible
  - proporciona más información
- Ventajas de usar *fKenzo*:
  - no es necesario conocer Common Lisp
  - usabilidad, accesibilidad

Kenzo vs fKenzo

- Ventajas de usar Kenzo:
  - toda la funcionalidad disponible
  - proporciona más información
- Ventajas de usar *fKenzo*:
  - no es necesario conocer Common Lisp
  - usabilidad, accesibilidad
  - guía al usuario en la interacción con el sistema

Kenzo vs fKenzo

- Ventajas de usar Kenzo:
  - toda la funcionalidad disponible
  - proporciona más información
- Ventajas de usar *fKenzo*:
  - no es necesario conocer Common Lisp
  - usabilidad, accesibilidad
  - guía al usuario en la interacción con el sistema
  - proporciona algunas mejoras al sistema Kenzo (Demo)

Índice

Motivación

Imagen digital

Motivación

$$\begin{tikzcd} \text{Imagen digital} \arrow[d, "\text{triangulación}"] \\ \text{Complejo Simplicial} \end{tikzcd}$$

Motivación

$$
\begin{array}{c}
\text{Imagen digital} \\
\downarrow {\color{red}\text{triangulación}} \\
\text{Complejo Simplicial} \\
\downarrow {\color{red}\text{estructura graduada}} \\
\text{Conjunto Simplicial}
\end{array}
$$

Motivación

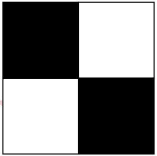
$$\begin{tikzcd}
\text{Imagen digital} \arrow[d, "\text{triangulación}"] \\
\text{Complejo Simplicial} \arrow[d, "\text{estructura graduada}"] \\
\text{Conjunto Simplicial} \arrow[d, "\text{cálculo}"] \\
\text{Homología}
\end{tikzcd}$$

Motivación

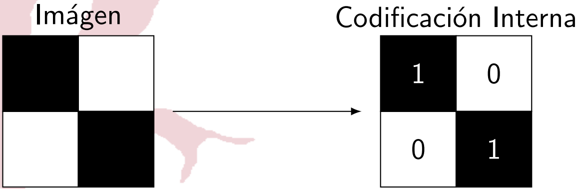
$$
\begin{tikzcd}
\text{Imagen digital} \arrow[d, "\text{triangulación}"] \\
\text{Complejo Simplicial} \arrow[d, "\text{estructura graduada}"] \\
\text{Conjunto Simplicial} \arrow[d, "\text{cálculo}"] \\
\text{Homología} \arrow[uuu, bend left=60, "\text{propiedades}"]
\end{tikzcd}
$$

Ejemplo

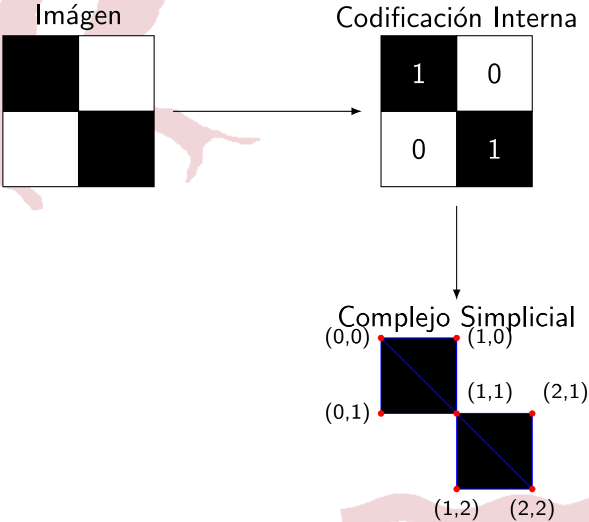
Imágen



Ejemplo

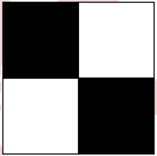


Ejemplo

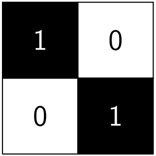

Imágen
Codificación Interna
1
0
0
1
Complejo Simplicial
(0,0)
(1,0)
(1,1)
(2,1)
(0,1)
(1,2)
(2,2)

Ejemplo

Imágen



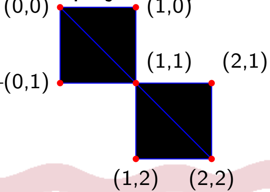
Codificación Interna



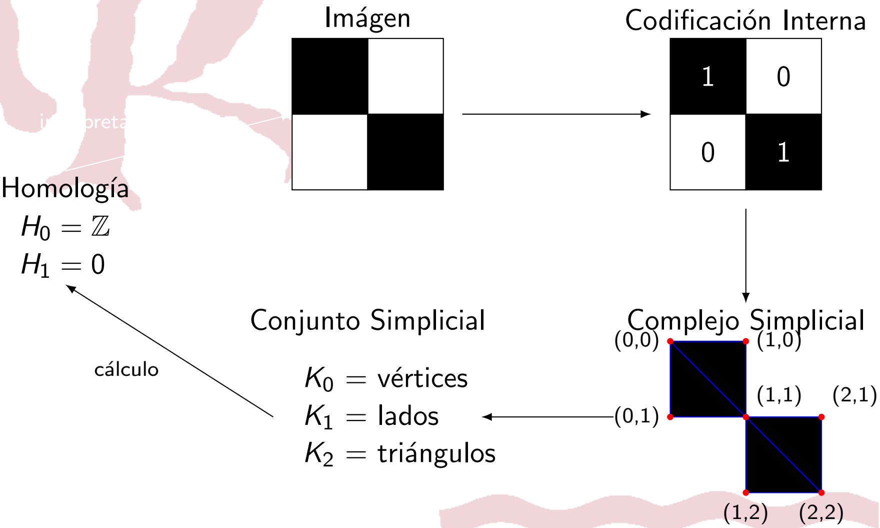
Conjunto Simplicial

$$K_0 = \text{vértices}$$
$$K_1 = \text{lados}$$
$$K_2 = \text{triángulos}$$

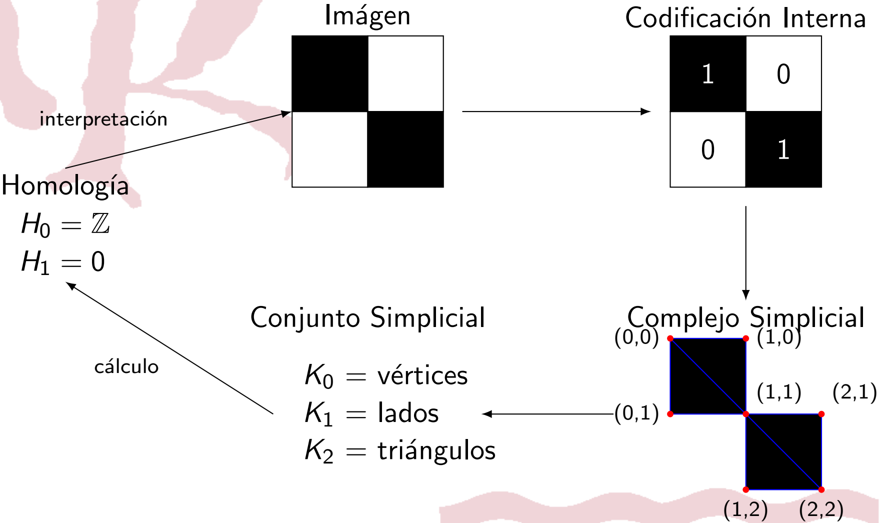
Complejo Simplicial



Ejemplo


Imágen
Codificación Interna
1 0
0 1
Complejo Simplicial
(0,0) (1,0)
(1,1) (2,1)
(0,1)
(1,2) (2,2)
Conjunto Simplicial
$K_0 = $ vértices
$K_1 = $ lados
$K_2 = $ triángulos
cálculo
Homología
$H_0 = \mathbb{Z}$
$H_1 = 0$

Ejemplo


Imágen
Codificación Interna
1 0
0 1
interpretación
Homología
$H_0 = \mathbb{Z}$
$H_1 = 0$
Conjunto Simplicial
Complejo Simplicial
(0,0) (1,0)
(1,1) (2,1)
(0,1)
(1,2) (2,2)
cálculo
K_0 = vértices
K_1 = lados
K_2 = triángulos

Integración del ejemplo en Kenzo/fKenzo

$$
\begin{tikzcd}
\text{Imagen digital} \arrow[d, "\text{triangulación}"] \\
\text{Complejo Simplicial} \arrow[d, "\text{estructura graduada}"] \\
\text{Conjunto Simplicial} \arrow[d, "\text{cálculo}"] \\
\text{Homología}
\end{tikzcd}
$$

Integración del ejemplo en Kenzo/fKenzo

$$
\begin{tikzcd}
\text{Imagen digital} \arrow[d, "\text{triangulación}"] \\
\text{Complejo Simplicial} \arrow[d, "\text{estructura graduada}"] \\
\text{Conjunto Simplicial} \arrow[d, "\text{cálculo}"] \\
\text{Homología}
\end{tikzcd}
$$

Integración del ejemplo en Kenzo/fKenzo

$$
\begin{tikzcd}
\text{Imagen digital} \arrow[d, "\text{triangulación}"] \\
\text{Complejo Simplicial} \arrow[d, "\text{estructura graduada}"] \\
\text{Conjunto Simplicial} \arrow[d, "\text{cálculo}"] \\
\text{Homología}
\end{tikzcd}
$$

Certificación de la corrección de los programas.

J. Heras, V. Pascual and J. Rubio, *An ACL2 certified module of Simplicial Complexes for the Kenzo system*. Preprint

Demo

Demo

Índice

Conclusiones

- Homología Efectiva:

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología
- Kenzo:

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología
- Kenzo:
  - Sistema de cálculo simbólico para la homología efectiva

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología
- Kenzo:
  - Sistema de cálculo simbólico para la homología efectiva
  - usado para calcular sobre espacios de lazos iterados, espacios clasificantes, ...

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología
- Kenzo:
  - Sistema de cálculo simbólico para la homología efectiva
  - usado para calcular sobre espacios de lazos iterados, espacios clasificantes, . . .
- *fKenzo*:

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología
- Kenzo:
  - Sistema de cálculo simbólico para la homología efectiva
  - usado para calcular sobre espacios de lazos iterados, espacios clasificantes, ...
- *fKenzo*:
  - interfaz agradable para el uso de Kenzo

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología
- Kenzo:
  - Sistema de cálculo simbólico para la homología efectiva
  - usado para calcular sobre espacios de lazos iterados, espacios clasificantes, ...
- *fKenzo*:
  - interfaz agradable para el uso de Kenzo
  - restringe la funcionalidad de Kenzo

Conclusiones

- Homología Efectiva:
  - Proporciona algoritmos reales para el cálculo de grupos de homología
- Kenzo:
  - Sistema de cálculo simbólico para la homología efectiva
  - usado para calcular sobre espacios de lazos iterados, espacios clasificantes, . . .
- *fKenzo*:
  - interfaz agradable para el uso de Kenzo
  - restringe la funcionalidad de Kenzo
  - proporciona un acceso mediado

Conclusiones

Imágenes digitales:

Conclusiones

Imágenes digitales:

- Pequeño caso de estudio

Conclusiones

Imágenes digitales:

- Pequeño caso de estudio
- Kenzo:

Conclusiones

Imágenes digitales:

- Pequeño caso de estudio
- Kenzo:
  - No pensado para procesar grandes matrices

Conclusiones

Imágenes digitales:

- Pequeño caso de estudio
- Kenzo:
  - No pensado para procesar grandes matrices
  - Paradoja:

Conclusiones

Imágenes digitales:

- Pequeño caso de estudio
- Kenzo:
  - No pensado para procesar grandes matrices
  - Paradoja:
    - Pensado para procesar matrices infinitas

Conclusiones

Imágenes digitales:

- Pequeño caso de estudio
- Kenzo:
  - No pensado para procesar grandes matrices
  - Paradoja:
    - Pensado para procesar matrices infinitas
    - Proceso de reducción produce matrices pequeñas

Conclusiones

Imágenes digitales:

- Pequeño caso de estudio
- Kenzo:
  - No pensado para procesar grandes matrices
  - Paradoja:
    - Pensado para procesar matrices infinitas
    - Proceso de reducción produce matrices pequeñas
- W-reductions of digital images:

F. Sergeraert, *Discrete Vector Fields and Fundamental Algebraic Topology*. Preprint, www-available: http://www-fourier.ujf-grenoble.fr/~sergerar/Papers/

Herramientas de cálculo simbólico para la Topología Algebraica: homología efectiva

Jónathan Heras,
V. Pascual, J. Rubio y F. Sergeraert

Departamento de Matemáticas y Computación
Universidad de La Rioja
Spain

22 de Mayo de 2010