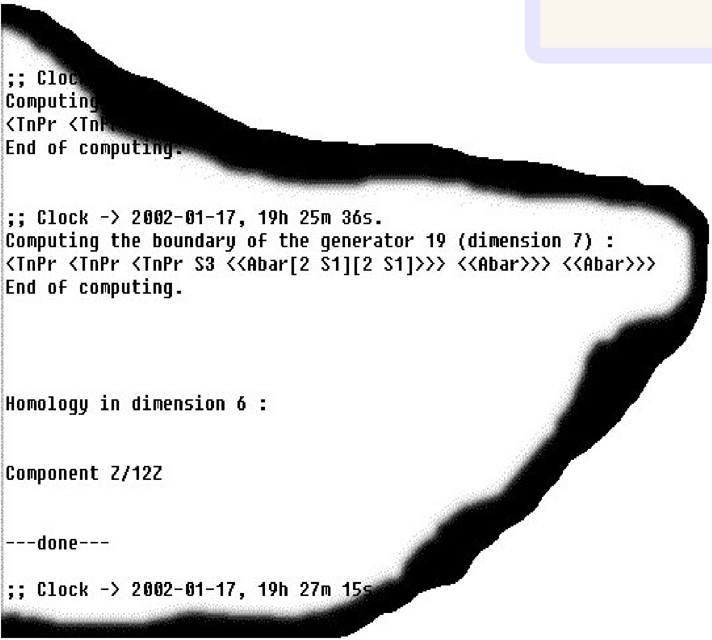
The Paradox of the Exact Couples



Francis Sergeraert, Institut Fourier, Grenoble
Department of Mathematics and Computer Science
University of La Rioja, January 14-15, 2009

Semantics of colours:

Blue = "Standard" Mathematics

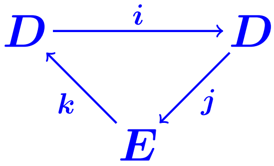
Red = Constructive, effective,

algorithm, machine object, . . .

Violet = Problem, difficulty, obstacle, disadvantage, . . .

Green = Solution, essential point, mathematicians, . . .

Definition: An exact couple is a diagram:



with:

- The components D and E are some $\mathbb{Z}$-modules.
- The components i, j and k are module morphisms.
- The circular sequence:

$$\{\cdots \to D \xrightarrow{i} D \xrightarrow{j} E \xrightarrow{k} D \xrightarrow{i} D \to \cdots\}$$

is exact.

Denoted by (D, E, i, j, k).

Definition: Let $\begin{array}{ccc} D & \xrightarrow{i} & D \\ & {}_{k}\nwarrow \quad \swarrow_{j} & \\ & E & \end{array}$ be an exact couple.

The derived exact couple $\begin{array}{ccc} D' & \xrightarrow{i'} & D' \\ & {}_{k'}\nwarrow \quad \swarrow_{j'} & \\ & E' & \end{array}$ is defined as follows:

- $D' := i(D) = \mathrm{im}(i) = \ker(j)$.
- $E' := H(E, jk)$ is the homology group of the differential module E provided with the differential jk.
- $i' := i|D'$.
- If $a \in D'$, this implies there exists some $b \in D$ satisfying $i(b) = a$; then $j'(a) := j(b)$.
- If $a \in E'$, the homology class a is represented by some cycle $b \in Z(E, jk)$; then $k'(a) := k(b)$.

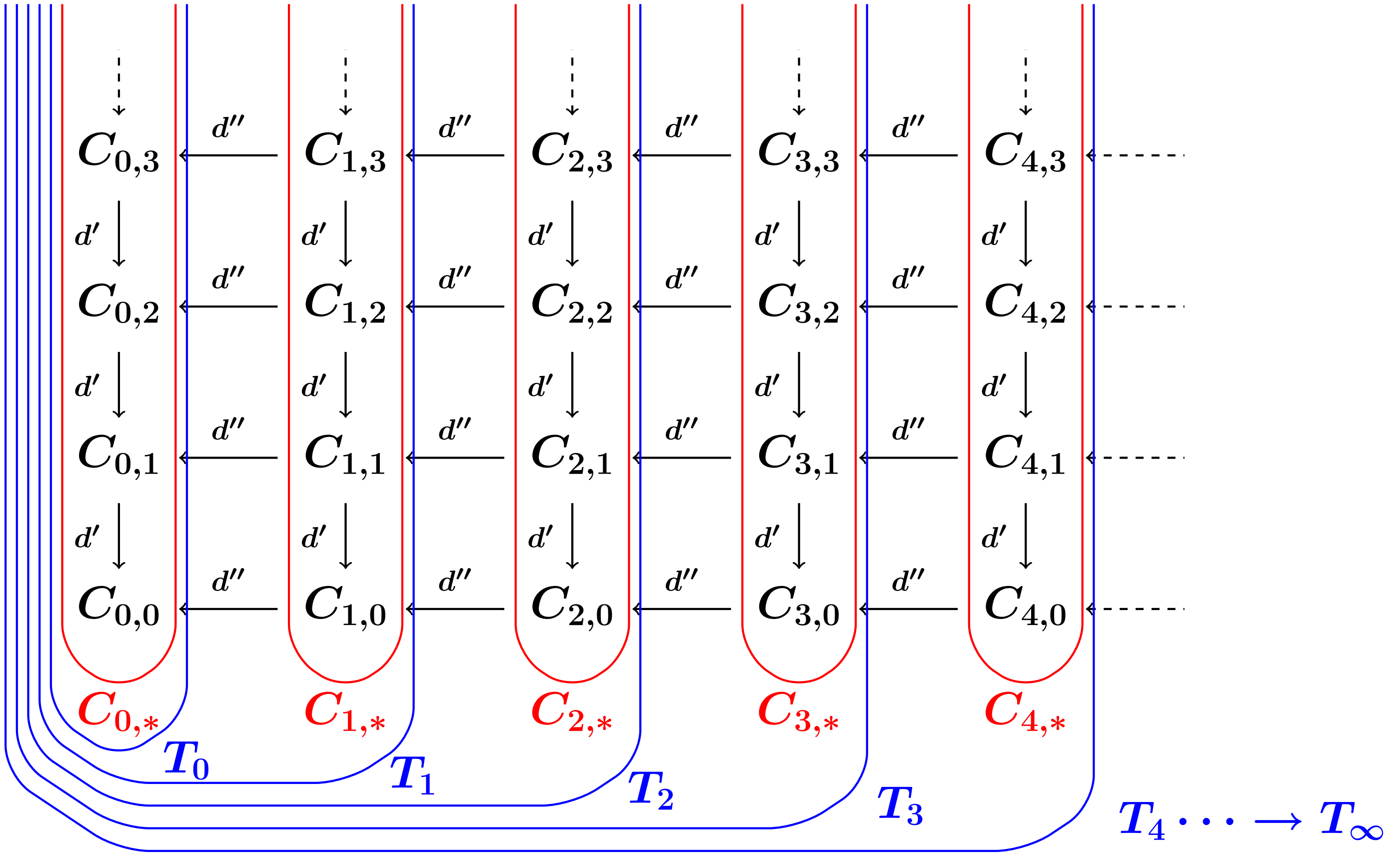
$\Rightarrow$ **Iteration of the derivation process:**

$$\begin{array}{c} D^{(1)} \xrightarrow{i} D^{(1)} \\ {\scriptstyle k}\nwarrow \quad \swarrow{\scriptstyle j} \\ E^{(1)} \end{array} \longmapsto \begin{array}{c} D^{(2)} \xrightarrow{i} D^{(2)} \\ {\scriptstyle k}\nwarrow \quad \swarrow{\scriptstyle j} \\ E^{(2)} \end{array} \longmapsto \cdots \longmapsto \begin{array}{c} D^{(r)} \xrightarrow{i} D^{(r)} \\ {\scriptstyle k}\nwarrow \quad \swarrow{\scriptstyle j} \\ E^{(r)} \end{array} \longmapsto \cdots$$

In many circumstances, $E^{(r)}$ = page r of a spectral sequence.

What about the computability of these exact couples ??

Very strange situation with respect to computability !!


d''
d'
$C_{0,3}$
$C_{1,3}$
$C_{2,3}$
$C_{3,3}$
$C_{4,3}$
$C_{0,2}$
$C_{1,2}$
$C_{2,2}$
$C_{3,2}$
$C_{4,2}$
$C_{0,1}$
$C_{1,1}$
$C_{2,1}$
$C_{3,1}$
$C_{4,1}$
$C_{0,0}$
$C_{1,0}$
$C_{2,0}$
$C_{3,0}$
$C_{4,0}$
$C_{0,*}$
$C_{1,*}$
$C_{2,*}$
$C_{3,*}$
$C_{4,*}$
T_0
T_1
T_2
T_3
$T_4 \cdots \longrightarrow T_\infty$

$$\begin{array}{ccc} D^{(1)} & \xrightarrow{i} & D^{(1)} \\ & {}_{k}\nwarrow \quad \swarrow_{j} & \\ & E^{(1)} & \end{array} \longmapsto \begin{array}{ccc} D^{(2)} & \xrightarrow{i} & D^{(2)} \\ & {}_{k}\nwarrow \quad \swarrow_{j} & \\ & E^{(2)} & \end{array} \longmapsto \cdots \longmapsto \begin{array}{ccc} D^{(r)} & \xrightarrow{i} & D^{(r)} \\ & {}_{k}\nwarrow \quad \swarrow_{j} & \\ & E^{(r)} & \end{array} \longmapsto \cdots$$

Simplest example of the bicomplex exact couple.

Let $(C_{p,q}, d'_{p,q}, d''_{p,q})$ be a first quadrant bicomplex.

$\Rightarrow$

$D^{(1)} := \bigoplus_p H_* T_p$ **with** $T_p =$ **totalization**

of the sub-bicomplex made of columns $0 \cdots p$.

$E^{(1)} = \bigoplus_p H_* C_{p,*} =$ **homology of columns.**

$[r \to \infty] \Rightarrow [E^{(r)} \to H_* T_\infty]$.

Paradox:

Paradox:

1. **You intend to compute H_*T_∞.**

Paradox:

1. **You intend to compute H_*T_∞.**

2. **You learn the exact couple method.**

Paradox:

1. You intend to compute H_*T_∞.

2. You learn the exact couple method.

3. $H_*T_\infty = \lim_{r\to\infty} E^{(r)}$.

Paradox:

1. **You intend to compute H_*T_∞.**

2. **You learn the exact couple method.**

3. $H_*T_\infty = \lim_{r\to\infty} E^{(r)}$.

4. **$\Rightarrow$ You must compute $(E^{(r)})_{1\leq r}$.**

Paradox:

1. You intend to compute H_*T_∞.

2. You learn the exact couple method.

3. $H_*T_\infty = \lim_{r\to\infty} E^{(r)}$.

4. $\Rightarrow$ You must compute $(E^{(r)})_{1\le r}$.

5. $\Rightarrow$ You must start with $E^{(1)}$ and $\boxed{D^{(1)}}$.

Paradox:

1. You intend to compute $H_* T_\infty$.

2. You learn the exact couple method.

3. $H_* T_\infty = \lim_{r \to \infty} E^{(r)}$.

4. $\Rightarrow$ You must compute $(E^{(r)})_{1 \leq r}$.

5. $\Rightarrow$ You must start with $E^{(1)}$ and $\boxed{D^{(1)}}$.

6. But $D^{(1)} := \bigoplus_p H_* T_p \Leftrightarrow H_* T_\infty$!!!

Paradox:

1. You intend to compute H_*T_∞.

2. You learn the exact couple method.

3. $H_*T_\infty = \lim_{r\to\infty} E^{(r)}$.

4. $\Rightarrow$ You must compute $(E^{(r)})_{1\leq r}$.

5. $\Rightarrow$ You must start with $E^{(1)}$ and $\boxed{D^{(1)}}$.

6. But $D^{(1)} := \bigoplus_p H_*T_p \Leftrightarrow H_*T_\infty$!!!

$\Rightarrow$ **Vicious Circle** !!!

Paradox:

1. You intend to compute H_*T_∞.

2. You learn the exact couple method.

3. $H_*T_\infty = \lim_{r\to\infty} E^{(r)}$.

4. $\Rightarrow$ You must compute $(E^{(r)})_{1\le r}$.

5. $\Rightarrow$ You must start with $E^{(1)}$ and $\boxed{D^{(1)}}$.

6. But $D^{(1)} := \bigoplus_p H_*T_p \Leftrightarrow H_*T_\infty$!!!

$\Rightarrow$ **Vicious Circle** !!!

$\Rightarrow$ Serious Computability Problem !!!

John McCleary, 1985:

"User's Guide to Spectral Sequences"

It is worth repeating the caveat about differentials

mentioned in chapter 1:

Knowledge of $E_r^{*,*}$ and d_r determines $E_{r+1}^{*,*}$ but not d_{r+1}.

[. . .]

If some differential is unknown,

then some other (any other!) principle is needed to proceed.

[. . .]

In the non-trivial cases,

it is often a deep geometric idea

that is caught up in the knowledge of a differential.

Good news: It is in fact possible

to overcome MacCleary's obstacle !

Theorem:

It is possible to break off the vicious circle of exact couples

and also to solve the extension problems at abutment.

Key point: Notion of locally semi -effective object.

Notion of locally effective abelian group:

$$M = (\tau_M, \varepsilon_M, +_M, 1_M, -_M)$$

with :

- τ_M = **data type** = $\tau_M : \mathcal{U} \to \mathbb{B} = \{\bot, \top\}$.
- $\mathcal{M}$ = **{Objects coding M-elements}** = $\tau_M^{-1}(\top)$.
- ε_M = **M-comparator** = $\varepsilon_M : \mathcal{M} \times \mathcal{M} \to \mathbb{B}$.
- $+_M$ = **M-adder** = $+_M : \mathcal{M} \times \mathcal{M} \to \mathcal{M}$.
- $1_M \in \mathcal{M}$ = **neutral element**.
- $-_M$ = **M-opposite** : $-_M : \mathcal{M} \to \mathcal{M}$.

Notion of locally semi- effective abelian group:

$$M = (\tau_M, \varepsilon_M, +_M, 1_M, -_M)$$

with :

- τ_M = data type = $\tau_M : \mathcal{U} \to \mathbb{B} = \{\bot, \top\}$.
- $\mathcal{M}$ = {Objects coding M-elements} = $\tau_M^{-1}(\top)$.
- ~~ε_M = M-comparator = $\varepsilon_M : \mathcal{M} \times \mathcal{M} \to \mathbb{B}$.~~
- $+_M$ = M-adder = $+_M : \mathcal{M} \times \mathcal{M} \to \mathcal{M}$.
- $1_M \in \mathcal{M}$ = neutral element.
- $-_M$ = M-opposite : $-_M : \mathcal{M} \to \mathcal{M}$.

A locally semi- effective abelian group

is the same as

a locally effective abelian group

except the comparison operator ε_M is not available.

In particular it is no longer possible to decide

whether some $a \in \mathcal{M}$ is null or not.

It is so possible to overcome

the apparent paradox of the exact couples !!

Typical example of locally semi-effective object.

Let $M = (\tau_M, \varepsilon_M, +_M, 0_M, -_M, d_M)$ be

a locally effective differential abelian group.

M locally effective $\Rightarrow H(M, d_M)$ in general non-computable.

But $H(M, d_M)$ can be coded

as a locally semi- effective abelian group !!

Coding in this way the D-components of exact couples

makes the exact couple effective.

Standard locally effective coding:

$$\boldsymbol{M = (\tau_M, \varepsilon_M, +_M, 0_M, -_M, d_M)}$$

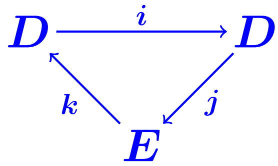
Then:

$$\boldsymbol{H(M, d_M) = (\tau_H, +_H, 0_H, -_H)}$$

with:

- $\boldsymbol{\tau_H(a) := \tau_M(a)}$ **and** $\boldsymbol{\varepsilon_M(d_M(a), 0_M)}$ ($\Leftrightarrow$ $\boldsymbol{a =}$ **cycle**)
- $\boldsymbol{+_H(a, b) := +_M(a, b)}$ ($\boldsymbol{+_H := +_M}$)
- $\boldsymbol{0_H := 0_M}$ ($\boldsymbol{0_H := 0_M}$)
- $\boldsymbol{-_H(a) := -_M(a)}$ ($\boldsymbol{-_H := -_M}$)

$\Rightarrow$ Notion of effective exact couple.



<u>Definition</u>: An exact couple (D, E, i, j, k) is effective if:

- **E is an effective $\mathbb{Z}$-module.**
- **D is a locally semi-effective $\mathbb{Z}$-module $D = Z/B$.**

  (Details later)

- **The corresponding circular sequence is effectively exact.**

  (Details later)

Required presentation

for the D-component of an effective exact couple:

$D = (Z, A, \alpha)$ with:

Z and A = locally effective $\mathbb{Z}$-modules.

$\alpha : A \to Z$ = morphism of $\mathbb{Z}$-modules.

$B := \alpha(A) \subset Z$.

Then $D := Z/B = Z/\alpha(A)$.

$$\begin{array}{ccc} & a \in & Z \\ & & \cup \\ A & \xrightarrow{\alpha} & B \end{array}$$

$$D = Z/B$$

Important : Let $a \in Z$.

Then A locally effective

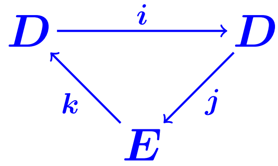
$\Rightarrow$ the membership relation $a \in B = \alpha(A)$ is undecidable.

Definition: Let $D = (Z, A, \alpha) = Z/B = Z/\alpha(A)$ be a locally semi-effective $\mathbb{Z}$-module implemented as before.

Let $\overline{a} \in D$ represented by (implemented as) an element $a \in Z$.

Then a certificate for the relation $\overline{a} = 0$ ($\Leftrightarrow a \in B = \alpha(A)$) for the element $\overline{a}$ so implemented by a is an element $a' \in A$ satisfying $\alpha(a') = a$.

Such certificates will be systematically required when defining the exactness property of a claimed effective exact couple.



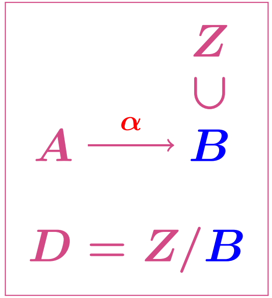
The module E is effective

and D is defined through $D = Z/\alpha(A)$.

The morphism i is implemented

as a morphism $i : Z \to Z$.

which must be effectively α-compatible.



A morphism $i' : A \to A$ must be given

satisfying $i \circ \alpha = \alpha \circ i'$.

The composition $Z \xrightarrow{i} Z \xrightarrow{j} E$ must be null.

What about the converse?

$$D \xrightarrow{\;i\;} D, \quad D \xrightarrow{\;j\;} E, \quad E \xrightarrow{\;k\;} D$$

Ordinary condition:

$a \in D$ **and** $j(a) = 0 \Rightarrow \boxed{\exists b \in D}$ <u>**st**</u> $i(b) = a$.

$$\begin{array}{ccc} & & Z \\ & & \cup \\ A & \xrightarrow{\;\alpha\;} & B \end{array}$$

$$D = Z/B$$

Effective form:

An algorithm $\beta_j : \ker(j) \to Z \times A$

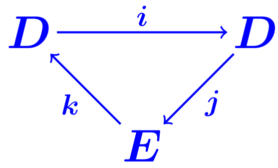
is given satisfying:

$$\beta_j(a) = (b, c) \;\Rightarrow\; a - i(b) = \alpha(c).$$

An i**-preimage** b **for** a **is computed**

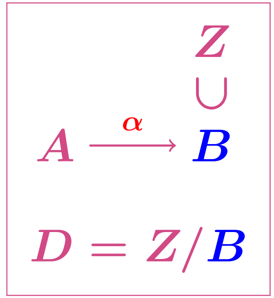
$\boxed{\text{with a certificate } c}$ **for the relation** $\boxed{i(b) \text{ “=” } a}$.

Example of the bicomplex exact couple.



$$D = \bigoplus_p H_*(T_p) \qquad E = \bigoplus_p H_*(C_p)$$

Effective exactness between i and j ?



$$D = Z/B \qquad Z = \bigoplus_p Z_*(T_p, d)$$

$$B = \bigoplus_p B_*(T_p, d)$$

$$A = \bigoplus_p (T_p) \qquad \alpha = d : \bigoplus_p (T_p) \to \bigoplus_p (T_p)$$

$$E = \bigoplus_p H_*(C_p)$$

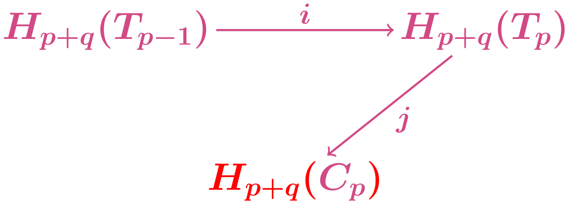
We assume the homological problem is solved for every C_p.

We intend to solve the homological problem for T_∞.

Solving the homological problem for a chain complex C_*

$\Leftrightarrow$ You must be able to:

1. **Determine the isomorphism class of $H_i(C_*)$ for arbitrary $i \in \mathbb{Z}$.**
2. **Produce a map $\rho : H_i(C_*) \to C_i$ giving a representant for every homology class.**
3. **Determine whether an arbitrary chain $c \in C_i$ is a cycle.**
4. **Compute, given an arbitrary cycle $z \in Z_i = \ker(d_i : C_i \to C_{i-1})$, its homology class $\overline{z} \in H_i(C_*)$.**
5. **Compute, given a cycle $z \in Z_i$ known as a boundary ($\overline{z} = 0$), a boundary-premimage $c \in C_{i+1}$ ($d_{i+1}(c) = z$).**



$z_{0,p+q} + \cdots + z_{p-1,q+1} + z_{p,q}$ "$\in$" $H_{p+q}(T_p)$

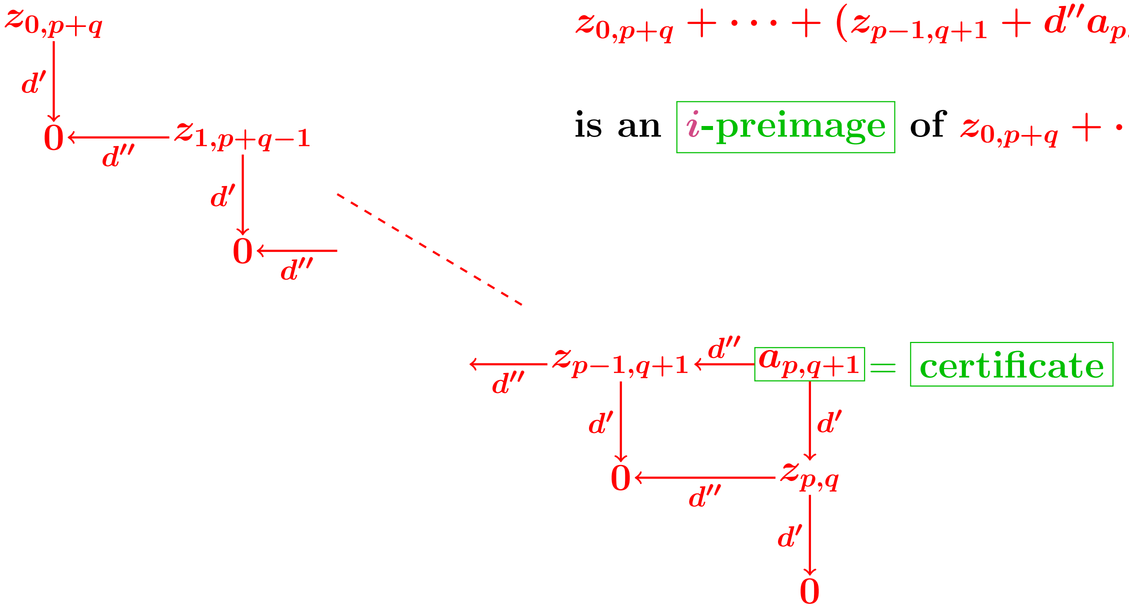
$j(z_{0,p+q} + \cdots + z_{p,q}) = 0 \;\Leftrightarrow\; z_{p,q} =$ bndr

C_p with hom. pb. solved $\Rightarrow\; a_{p,q+1}$

$z_{0,p+q} + \cdots + (z_{p-1,q+1} + d'' a_{p,q+1})$ "$\in$" $H_{p+q}(T_{p-1})$

is an i-preimage of $z_{0,p+q} + \cdots + z_{p,q}$

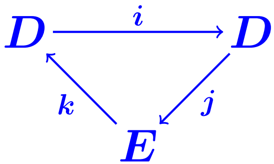
QED



Analogous interpretations

for the other components of effective exactness

for some exact couple:



<u>Fact</u>:

If the initial $E^{(1)}$ of some exact couple

corresponds to the homology groups of chain complexes

with the homological problem solved,

then the exact couple is effective.

Fundamental Theorem:

Let (D, E, i, j, k) be an effective exact couple.

Then an algorithm produces the derived exact couple
(D', E', i', j', k') which is also an effective exact couple.

Corollary: Same hypothesis.

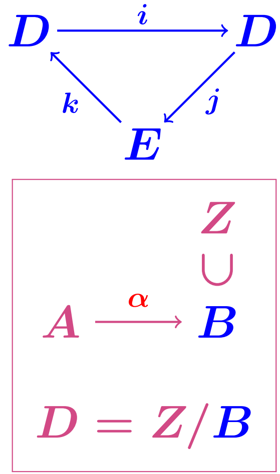
Then for every $n \geq 1$,
the iterated derivation $(D^{(n)}, E^{(n)}, i^{(n)}, j^{(n)}, k^{(n)})$
can be computed and is an effective exact couple.

Corollary: Same hypothesis.

Then the corresponding spectral sequence $(E^r)_{r \geq 1}$
is computable.

Proof.

$E' := H(E, jk)$ + E effective $\Rightarrow$ E' effective.

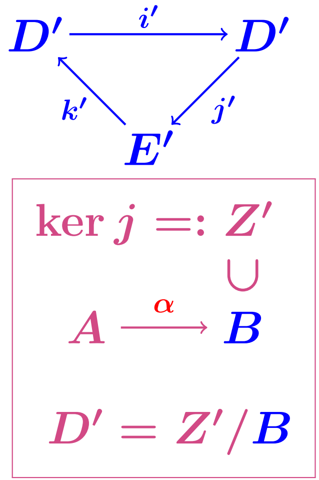


Semi-effective coding for $D' := i(D)$? ?

Note $i(D) = \ker j$

and j defined through $D = Z/B$

by $j : Z \to E$.

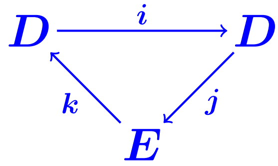


In particular $j(B) = 0$ and $B \subset \ker j$.

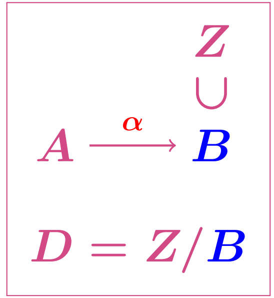
$\Rightarrow$ Solution $= D' = Z'/B$

with $Z' := \ker(j : Z \to E)$

and $\alpha : A \to B$ unchanged.



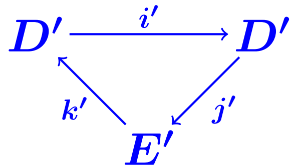
Computing j' ??



Standard definition:

$$a \in D' = \ker j \Rightarrow \exists b \in D \ \ \underline{\text{st}}\ i(b) = a.$$

Then $j'(a) := \overline{j(b)} \in H(E, jk)$.

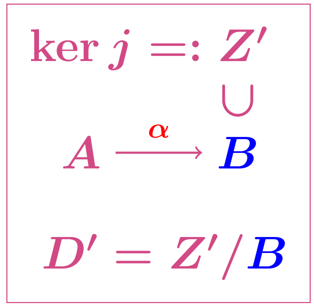


In our context: $a \in Z' = \ker j$

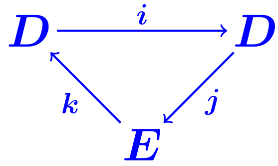
$\quad\quad + $ **effective exactness between** i **and** j

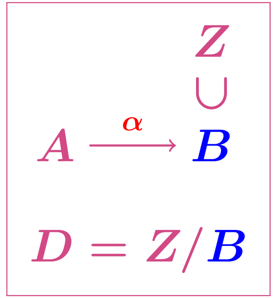
$\Rightarrow b \in Z$ **and** $c \in A$ **with** $i(b) = a + \alpha(c)$.

$\Rightarrow$ **Solution** $j'(a) = \overline{j(b)}$.



Effective exactness between i' and j' ??





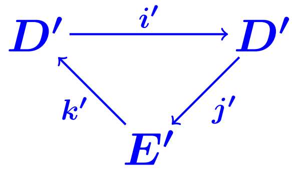
$a \in Z'$, $b \in Z$, $c \in A$, $ib = a + \alpha c$, $j'a = \overline{jb}$.

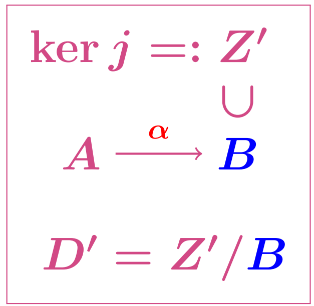
$j'a = 0 \Leftrightarrow \overline{jb} = 0 \Leftrightarrow jb = jkd$

$\Rightarrow j(b - kd) = 0$

$\Rightarrow (e \in Z) + (f \in A)$ st $ie = b - kd + \alpha f$

$\Rightarrow i(ie) = ib - ikd + i\alpha f = a + \alpha c - ikd + i\alpha f$





1) $ie \in Z'$

2) ik effectively null $\Rightarrow ikd = \alpha g$

3) i effectively α-compatible $\Rightarrow i\alpha f = \alpha f'$

$\Rightarrow i(ie) = a + \alpha(c - g + f')$

$\Rightarrow$ OK !

Analogous obvious arguments $\Rightarrow$ QED.

-o-o-o-o-o-o-

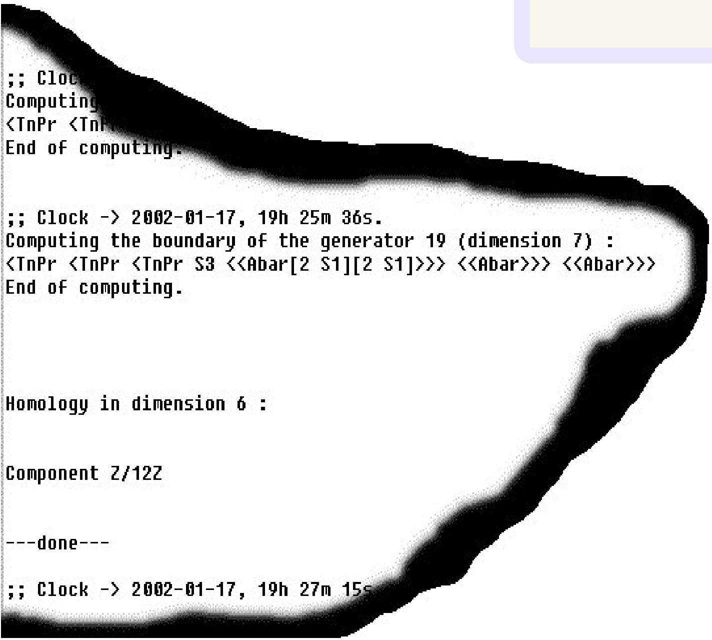
Next work:

Analysis of extension problems at abutment.

Programming.

Formal proof.

The END



Francis Sergeraert, Institut Fourier, Grenoble
Department of Mathematics and Computer Science
University of La Rioja, January 14-15, 2009