



The EXEMPLAR BREAKPOINT DISTANCE is not approximable for genomes with duplicated genes

Guillaume Blin¹ Guillaume Fertin² Florian Sikora¹
 Stéphane Vialette¹

¹ Université Paris-Est, LIGM - UMR CNRS 8049 - France
 {gblin,sikora,vialette}@univ-mlv.fr

² Université de Nantes, LINA - UMR CNRS 6241 - France
 guillaume.fertin@univ-nantes.fr

WALCOM February 2009



Outline

Motivations and notations

The EXEMPLAR BREAKPOINT DISTANCE (EBD) Problem

The ZERO EXEMPLAR BREAKPOINT DISTANCE (ZEBD) Problem

Outline

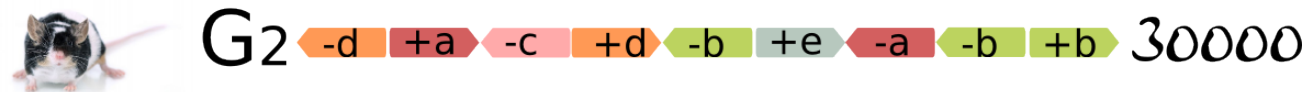
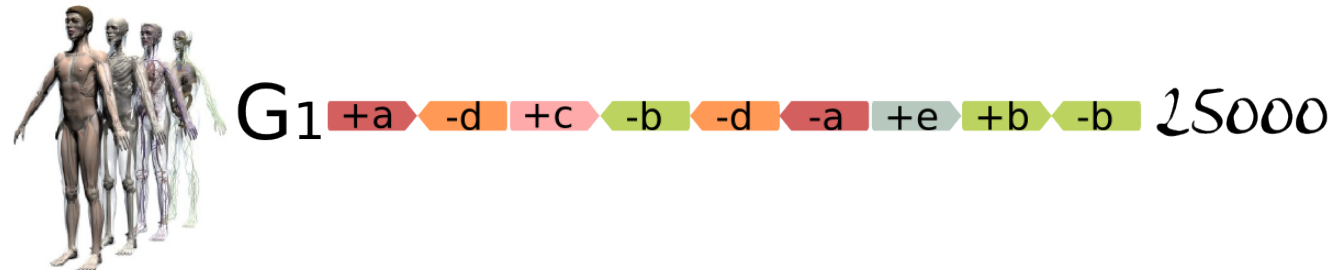
Motivations and notations

The EXEMPLAR BREAKPOINT DISTANCE (EBD) Problem

The ZERO EXEMPLAR BREAKPOINT DISTANCE (ZEBD) Problem

Motivations

- ▶ Comparing two genomes (set of signed genes) species



- ▶ Look for conserved set of genes in the same order



Look for a conserved set of genes

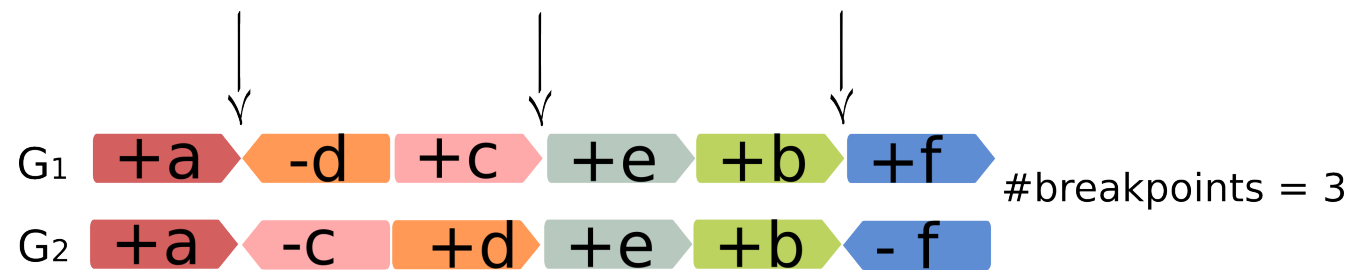
- ▶ Optimizing a given (dis)similarity measure
 - ▶ # breakpoints
 - ▶ # adjacencies
 - ▶ # conserved intervals
 - ▶ # common intervals
 - ▶ ...

Look for a conserved set of genes

- ▶ Optimizing a given (dis)similarity measure
 - ▶ **# breakpoints**
 - ▶ # adjacencies
 - ▶ # conserved intervals
 - ▶ # common intervals
 - ▶ ...

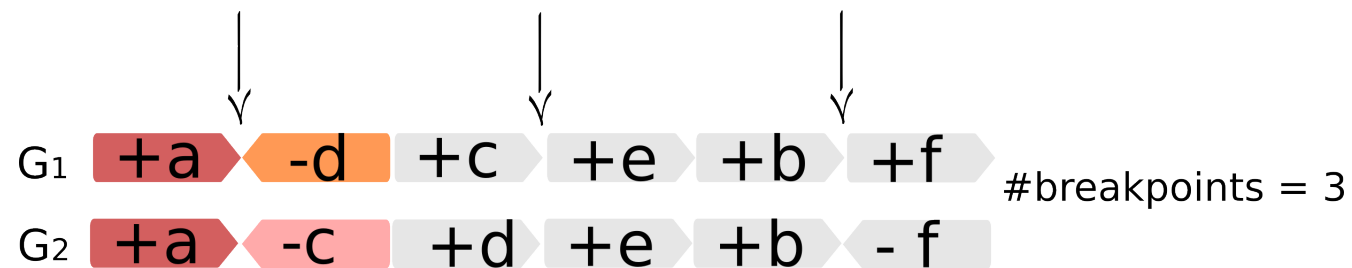
What is a breakpoint ?

- A non-conserved adjacency [Watterson *et al.* 1982]



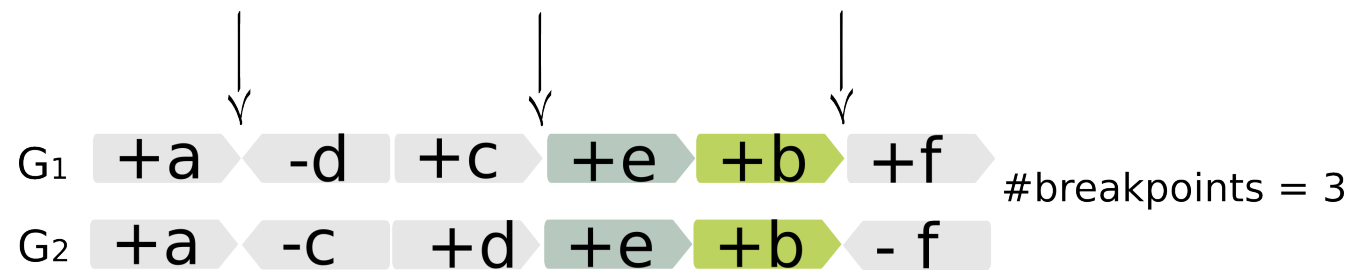
What is a breakpoint ?

- A non-conserved adjacency [Watterson *et al.* 1982]



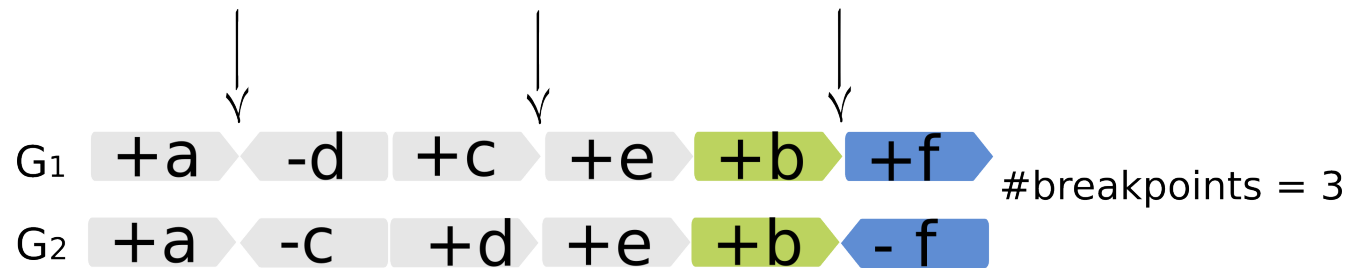
What is a breakpoint ?

- A non-conserved adjacency [Watterson *et al.* 1982]



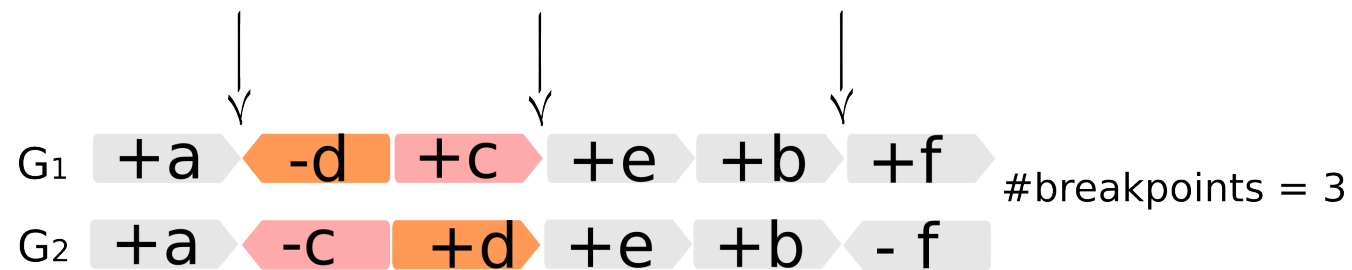
What is a breakpoint ?

- ▶ A non-conserved adjacency [Watterson *et al.* 1982]



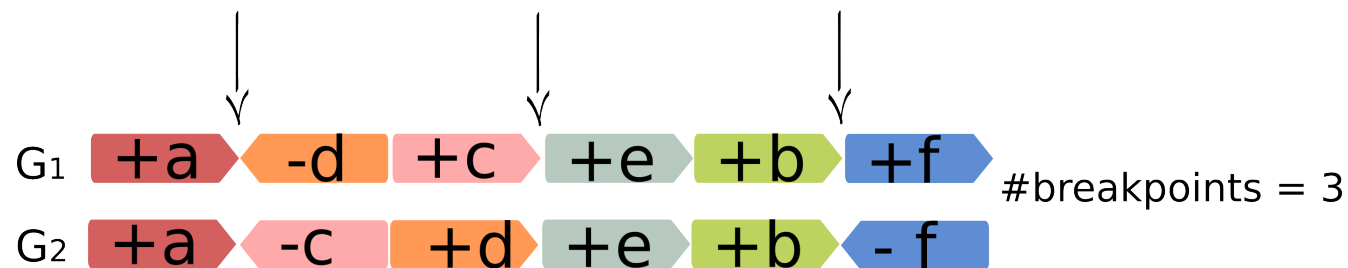
What is a breakpoint ?

- A non-conserved adjacency [Watterson *et al.* 1982]



What is a breakpoint ?

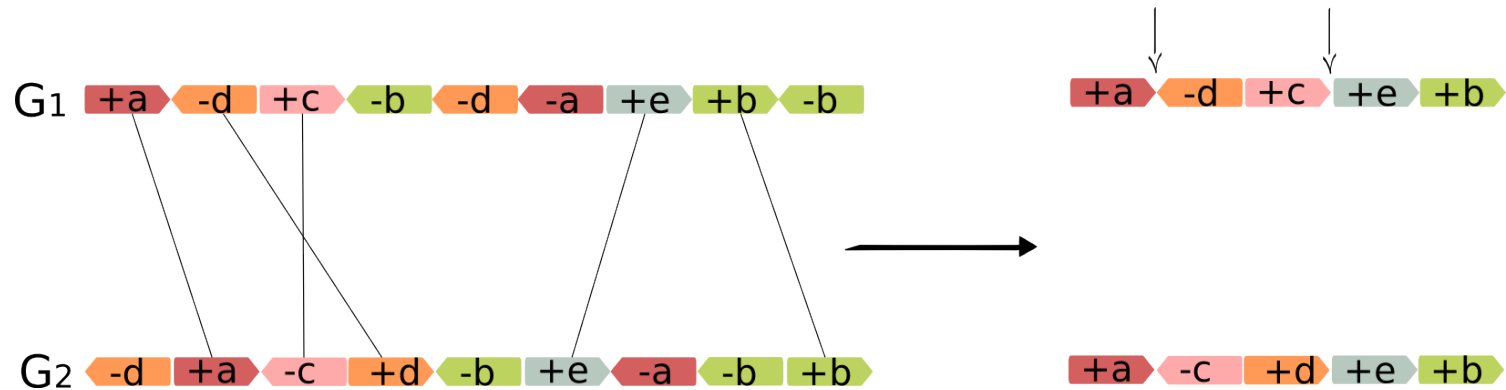
- ▶ A non-conserved adjacency [Watterson *et al.* 1982]



- ▶ Compute the number of breakpoints considering 2 genomes
- ▶ Done in **polynomial** time if there is **no duplicated** genes (only one occurrence of each gene family)
- ▶ Assumption which is not biologically valid

Exemplarization for dealing with duplicated genes

- ▶ Idea: perform an **exemplarization** which optimizes the number of breakpoints [Sankoff 1999]
- ▶ Exactly **one** occurrence of each gene family is kept (and matched)



- The gene kept is assumed to be the ancestral (the other derived from it)

Outline

Motivations and notations

The EXEMPLAR BREAKPOINT DISTANCE (EBD) Problem

The ZERO EXEMPLAR BREAKPOINT DISTANCE (ZEBD)
Problem

Definition of the EBD Problem

The EXEMPLAR BREAKPOINT DISTANCE Problem

Input: Two genomes G_1 , G_2 and an integer k

Question: Is it possible to establish an exemplar matching of G_1 and G_2 such that the number of breakpoints between the resulting exemplar genomes is at most k ?

Previous results

G_1 

- ▶ $occ(G_1) = 3$ (maximum # occurrences of a gene in the genome, here b)
- ▶ $EBD(p, q)$: EBD problem with $occ(G_1) = p$ and $occ(G_2) = q$

Previous results

G_1 +a -d +c -b -d -a +e +b -b

- ▶ $occ(G_1) = 3$ (maximum # occurrences of a gene in the genome, here b)
- ▶ $EBD(p, q)$: EBD problem with $occ(G_1) = p$ and $occ(G_2) = q$
- ▶ $EBD(1, 2)$: **NP-Complete** [Bryant 2000]
- ▶ $EBD(1, 2)$: **APX-Hard** [Angibaud *et al.* 2008]
- ▶ $EBD(3, 3)$: **no approximation** algorithm [Chen *et al.* 2006]
- ▶ **Open** : Is $EBD(2, 2)$ approximable ?

EBD Approximable ?

- ▶ OPT : the optimal solution of a problem Π
- ▶ $\mathcal{A}$: a polynomial algorithm with a performance guarantee of α for each instance of Π
- ▶ $\mathcal{A} \leq \alpha \times OPT$

EBD Approximable ?

- ▶ OPT : the optimal solution of a problem Π
- ▶ $\mathcal{A}$: a polynomial algorithm with a performance guarantee of α for each instance of Π
- ▶ $\mathcal{A} \leq \alpha \times OPT$
- ▶ OPT can be *zero* (if $k = 0$, no breakpoints allowed)
- ▶ If EBD with $k = 0$ is NP-Complete, no such algorithm $\mathcal{A}$ exists, Π is not approximable

Outline

Motivations and notations

The EXEMPLAR BREAKPOINT DISTANCE (EBD) Problem

The ZERO EXEMPLAR BREAKPOINT DISTANCE (ZEBD) Problem

Definition of the ZEBD Problem

The ZERO EXEMPLAR BREAKPOINT DISTANCE Problem

Input: Two genomes G_1 , G_2

Question: Is it possible to establish an exemplar matching of G_1 and G_2 such that the number of breakpoints between the resulting exemplar genomes is zero ?

Previous results

- ▶ $ZEBD(3, 3)$: **NP-Complete** [Chen *et al.* 2006]
- ▶ $ZEBD(2, q)$: **NP-Complete**, q unbounded [Angibaud *et al.* 2008]
- ▶ **Open** : Is $ZEBD(2, q)$ NP-Complete for bounded q ?

ZEBD is NP-Complete

- ▶ Our negative result: $\text{ZEBD}(2,q)$ is **NP-Complete**
- ▶ Reduction from 3-SAT (proof in our article)

FPT for ZEBD

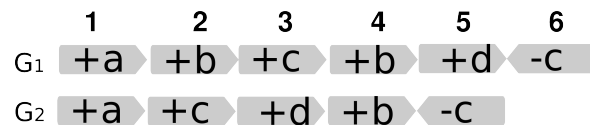
- ▶ Our positive result: ZEBD is **Fixed Parameter Tractable** (FPT)
- ▶ An FPT algorithm [Downey & Fellows 1999]:
exact algorithm **exponential** only in its **parameter** (not in the input size)

First FPT Algorithm – color-coding

- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]

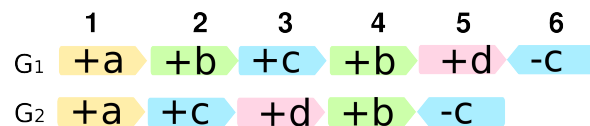
First FPT Algorithm – color-coding

- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



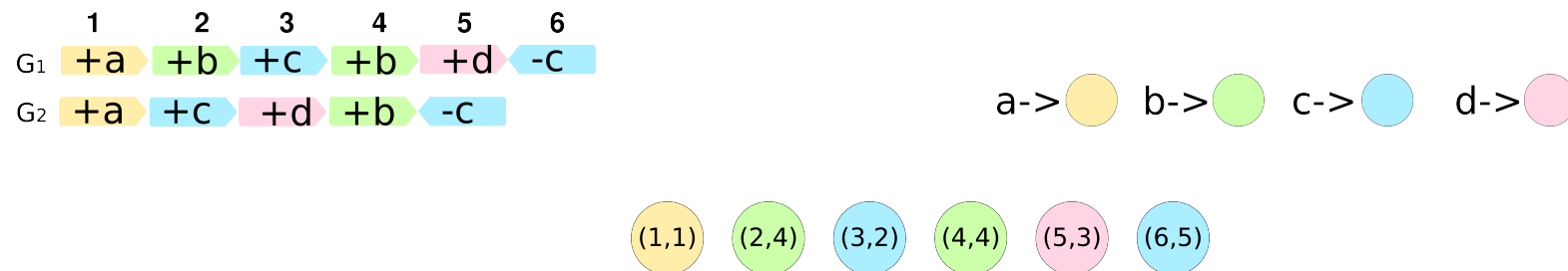
First FPT Algorithm – color-coding

- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



First FPT Algorithm – color-coding

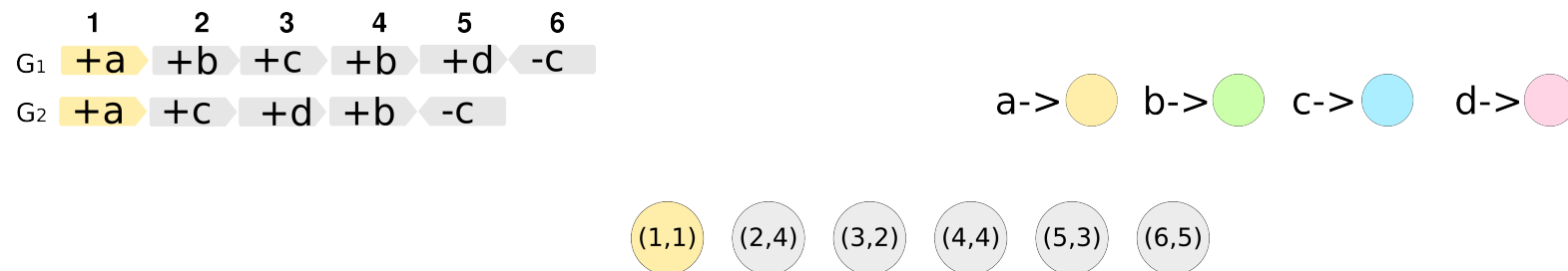
- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign

First FPT Algorithm – color-coding

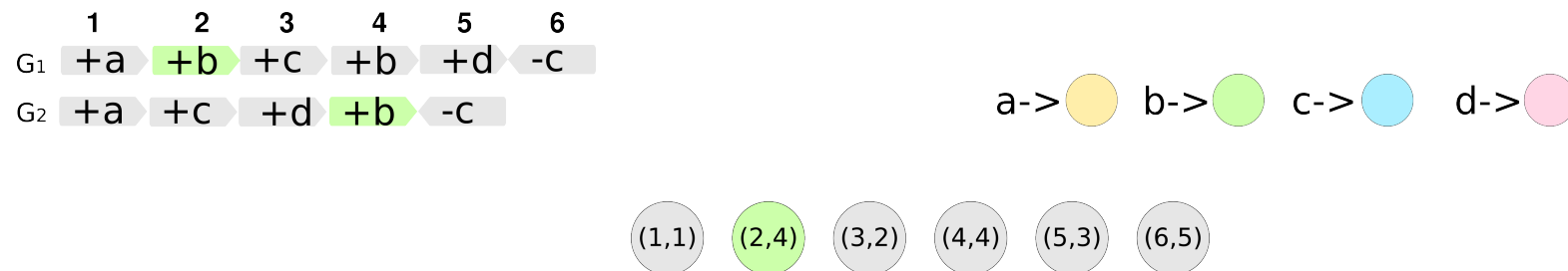
- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign

First FPT Algorithm – color-coding

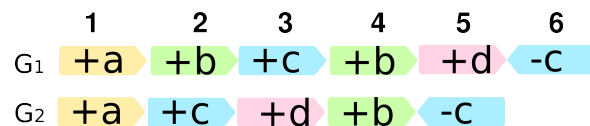
- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign

First FPT Algorithm – color-coding

- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



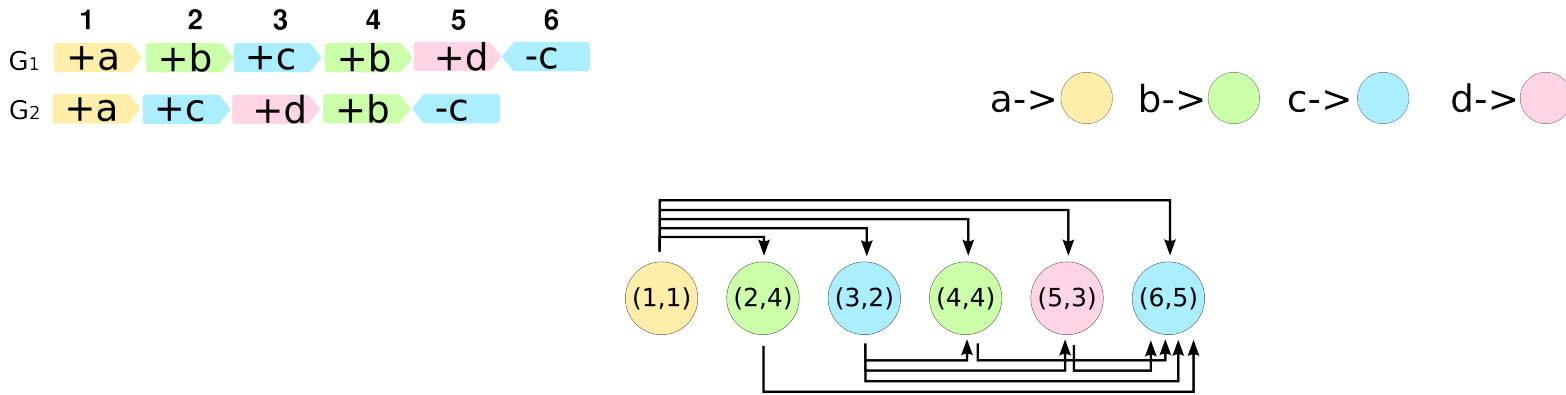
a->  b->  c->  d-> 



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign

First FPT Algorithm – color-coding

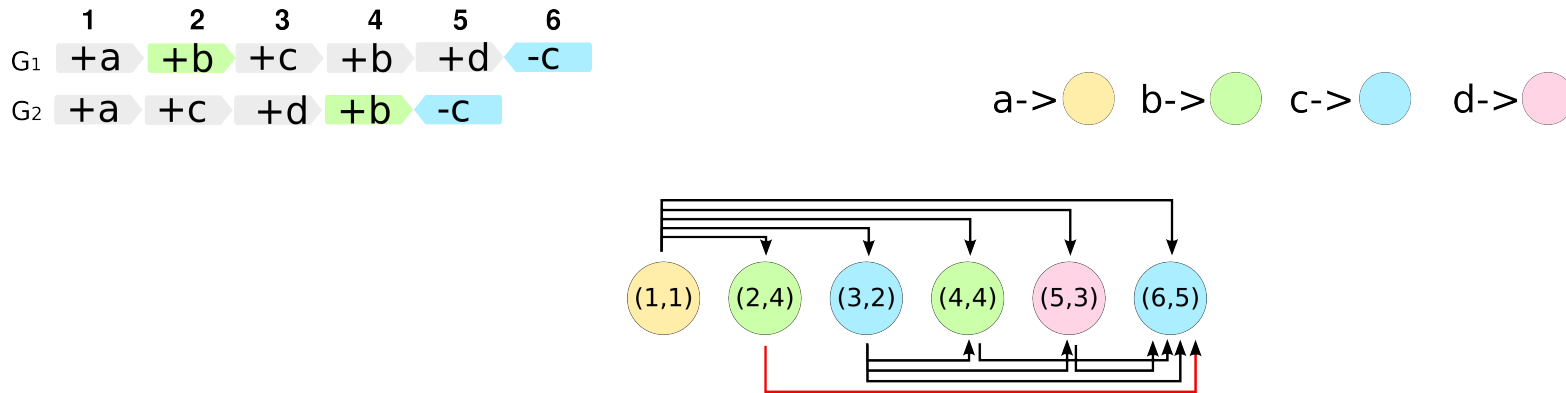
- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign
- ▶ For all $\{(i, j), (p, q)\} \in V^2$, an **edge** from (i, j) to (p, q) if $i < p$ and $j < q$

First FPT Algorithm – color-coding

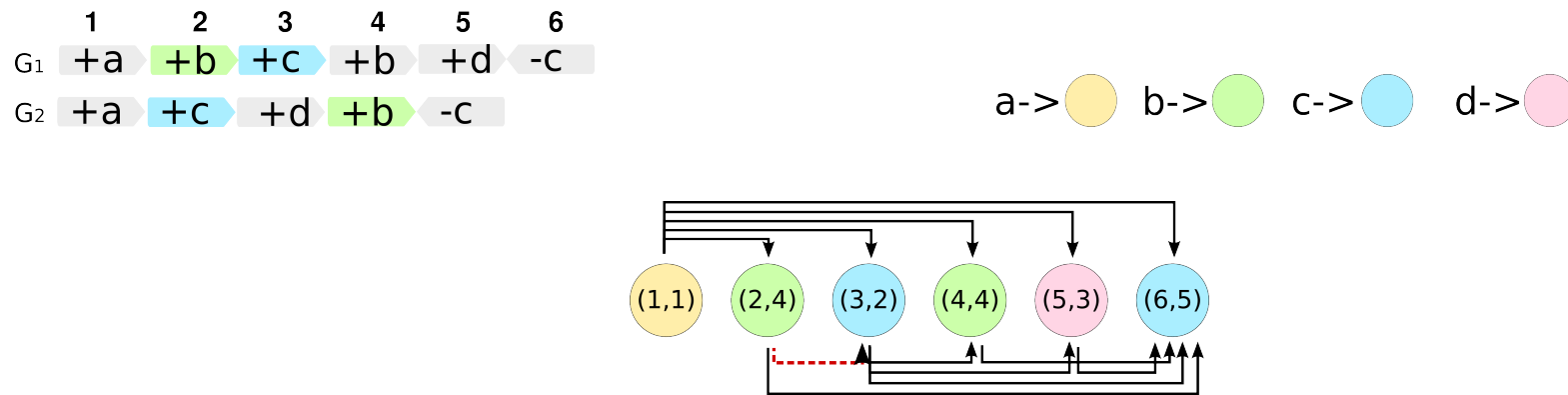
- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign
- ▶ For all $\{(i, j), (p, q)\} \in V^2$, an **edge** from (i, j) to (p, q) if $i < p$ and $j < q$

First FPT Algorithm – color-coding

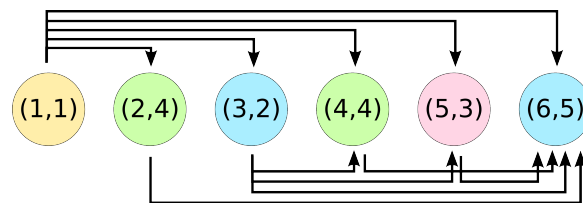
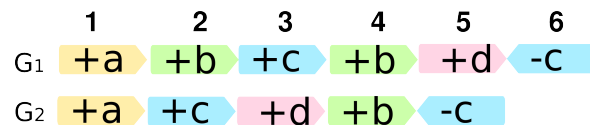
- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign
- ▶ For all $\{(i, j), (p, q)\} \in V^2$, an **edge** from (i, j) to (p, q) if $i < p$ and $j < q$

First FPT Algorithm – color-coding

- ▶ Parametrized by m , the number of genes families
- ▶ Using the **color-coding** technique by [Alon *et al.* 1995]



- ▶ A **vertex** v of V for each pair of genes of the same family that carry the same sign
- ▶ For all $\{(i, j), (p, q)\} \in V^2$, an **edge** from (i, j) to (p, q) if $i < p$ and $j < q$
- ▶ Looking for a colorful path: $\mathcal{O}(m2^m)$

Second FPT Algorithm

- The span: maximum distance between 2 occurrence of a gene family

G₁ 

Second FPT Algorithm

- The span: maximum distance between 2 occurrence of a gene family

G₁ +a -d +c -b -d -a +e +b -b

Second FPT Algorithm

- The span: maximum distance between 2 occurrence of a gene family

G₁ +a -d +c -b -d -a +e +b -b

Second FPT Algorithm

- ▶ The span: maximum distance between 2 occurrence of a gene family

G_1 

- ▶ Algorithm parametrized by the genome's span s (maximum span of a gene family in the genome)
- ▶ $span(G_1) = 5$ (for a family)

Second FPT Algorithm

$$G_1 = \overbrace{+a - b + a} \quad \overbrace{+c + d + c} \quad \overbrace{+d + e + f}$$

$$G_2 = \underbrace{+a - b + c - b + d - f + e - f + e}$$

- ▶ A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2

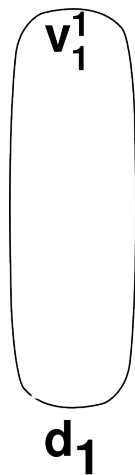
Second FPT Algorithm

$$G_1 = \overbrace{+a - b + a} + c + d + c + d + e + f$$

$$G_2 = +a - b + c - b + d - f + e - f + e$$

123456789

- A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2



$$v_1^1 = (+a, 1, 1)$$

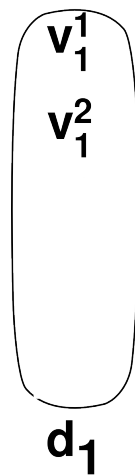
Second FPT Algorithm

$$G_1 = \overbrace{+a \text{ -b}} + a + c + d + c + d + e + f$$

$$G_2 = +a \text{ -b} + c - b + d - f + e - f + e$$

$$\begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{matrix}$$

- A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2



$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

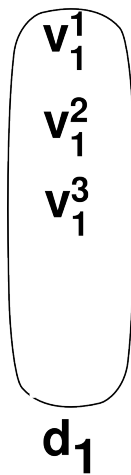
Second FPT Algorithm

$$G_1 = \overbrace{+a \text{ -b} + a} + c + d + c + d + e + f$$

$$G_2 = +a - b + c \text{ -b} + d - f + e - f + e$$

$$\begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{matrix}$$

- A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2



$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

$$v_1^3 = (-b, 4, 4)$$

Second FPT Algorithm

$$G_1 = \overbrace{+a -b} + a + c + d + c + d + e + f$$

$$G_2 = +a -b + c -b + d -f + e -f + e$$

$$\begin{array}{ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{array}$$

- A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2

$$\begin{array}{c} v_1^1 \\ v_1^2 \\ v_1^3 \\ v_1^4 \end{array}$$

d_1

$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

$$v_1^3 = (-b, 4, 4)$$

$$v_1^4 = (+a -b, 1, 2)$$

Second FPT Algorithm

$$G_1 = \overbrace{+a -b} + a + c + d + c + d + e + f$$

$$G_2 = +a -b +c -b + d - f + e - f + e$$

123456789

- A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2

v_1^1
 v_1^2
 v_1^3
 v_1^4
 v_1^5

d_1

$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

$$v_1^3 = (-b, 4, 4)$$

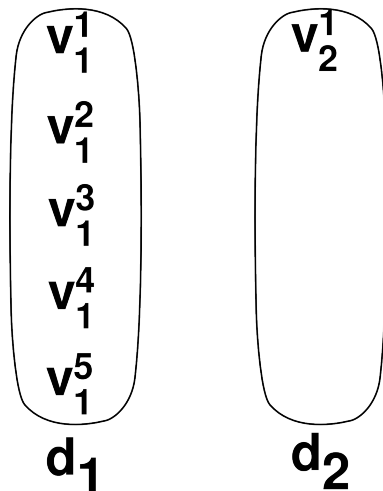
$$v_1^4 = (+a -b, 1, 2)$$

$$v_1^5 = (+a -b, 1, 4)$$

Second FPT Algorithm

$$\begin{array}{rcl}
 G_1 & = & +a - b + a \overbrace{+c + d + c} + d + e + f \\
 G_2 & = & +a - b +c - b + d - f + e - f + e \\
 & & \begin{array}{ccccccccc}
 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9
 \end{array}
 \end{array}$$

- A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2



$$\begin{array}{l}
 v_1^1 = (+a, 1, 1) \\
 v_1^2 = (-b, 2, 2) \\
 v_1^3 = (-b, 4, 4) \\
 v_1^4 = (+a - b, 1, 2) \\
 v_1^5 = (+a - b, 1, 4) \\
 v_2^1 = (+c, 3, 3)
 \end{array}$$

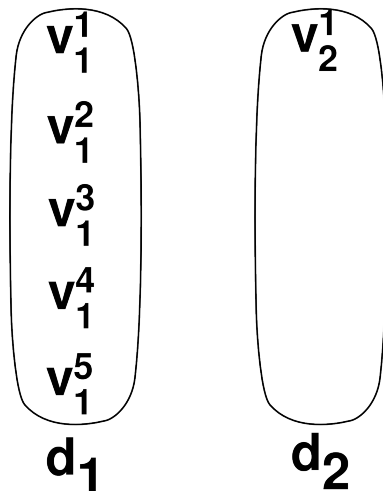
Second FPT Algorithm

$$G_1 = +a - b + a + c + d + c + d + e + f$$

$$G_2 = +a - b + c - b + d - f + e - f + e$$

1 2 3 4 5 6 7 8 9

- ▶ A vertex in the bag d_i for each common subsequence between each segment i of size s in G_1 and all G_2
- ▶ There is at most $|G_1|2^s s$ vertices



$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

$$v_1^3 = (-b, 4, 4)$$

$$v_1^4 = (+a - b, 1, 2)$$

$$v_1^5 = (+a - b, 1, 4)$$

$$v_2^1 = (+c, 3, 3)$$

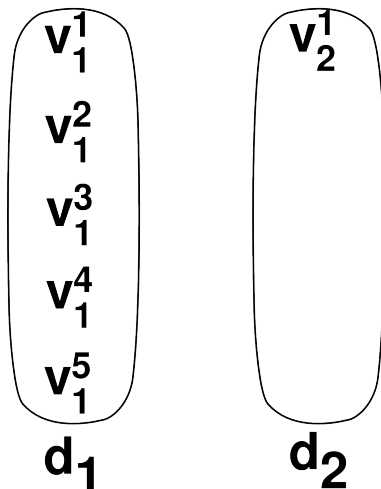
Second FPT Algorithm

$$G_1 = +a - b + a + c + d + c + d + e + f$$

$$G_2 = +a - b + c - b + d - f + e - f + e$$

1 2 3 4 5 6 7 8 9

- Add an edge if no gene of the same family in common (exemplarization) and not overlapping (no breakpoint)



$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

$$v_1^3 = (-b, 4, 4)$$

$$v_1^4 = (+a - b, 1, 2)$$

$$v_1^5 = (+a - b, 1, 4)$$

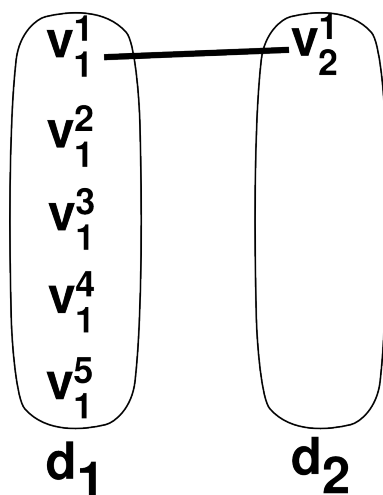
$$v_2^1 = (+c, 3, 3)$$

Second FPT Algorithm

$$G_1 = +a - b + a + c + d + c + d + e + f$$

$$G_2 = \underset{1}{+a} - \underset{2}{b} \underset{3}{+c} - \underset{4}{b} + \underset{5}{d} - \underset{6}{f} + \underset{7}{e} - \underset{8}{f} + \underset{9}{e}$$

- Add an edge if no gene of the same family in common (exemplarization) and not overlapping (no breakpoint)
- Ok



$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

$$v_1^3 = (-b, 4, 4)$$

$$v_1^4 = (+a - b, 1, 2)$$

$$v_1^5 = (+a - b, 1, 4)$$

$$v_2^1 = (+c, 3, 3)$$

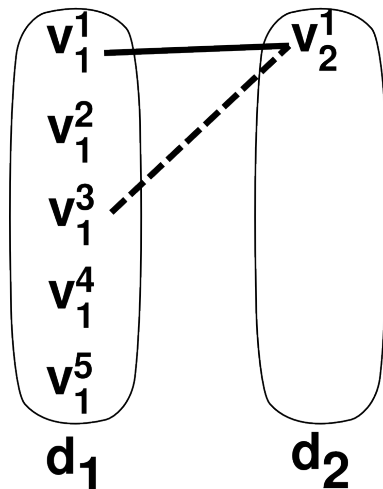
Second FPT Algorithm

$$G_1 = +a - b + a + c + d + c + d + e + f$$

$$G_2 = +a - b + \mathbf{c} - \mathbf{b} + d - f + e - f + e$$

1 2 3 4 5 6 7 8 9

- Add an edge if no gene of the same family in common (exemplarization) and not overlapping (no breakpoint)
- Overlapping



$$v_1^1 = (+a, 1, 1)$$

$$v_1^2 = (-b, 2, 2)$$

$$\mathbf{v_1^3 = (-b, 4, 4)}$$

$$v_1^4 = (+a - b, 1, 2)$$

$$v_1^5 = (+a - b, 1, 4)$$

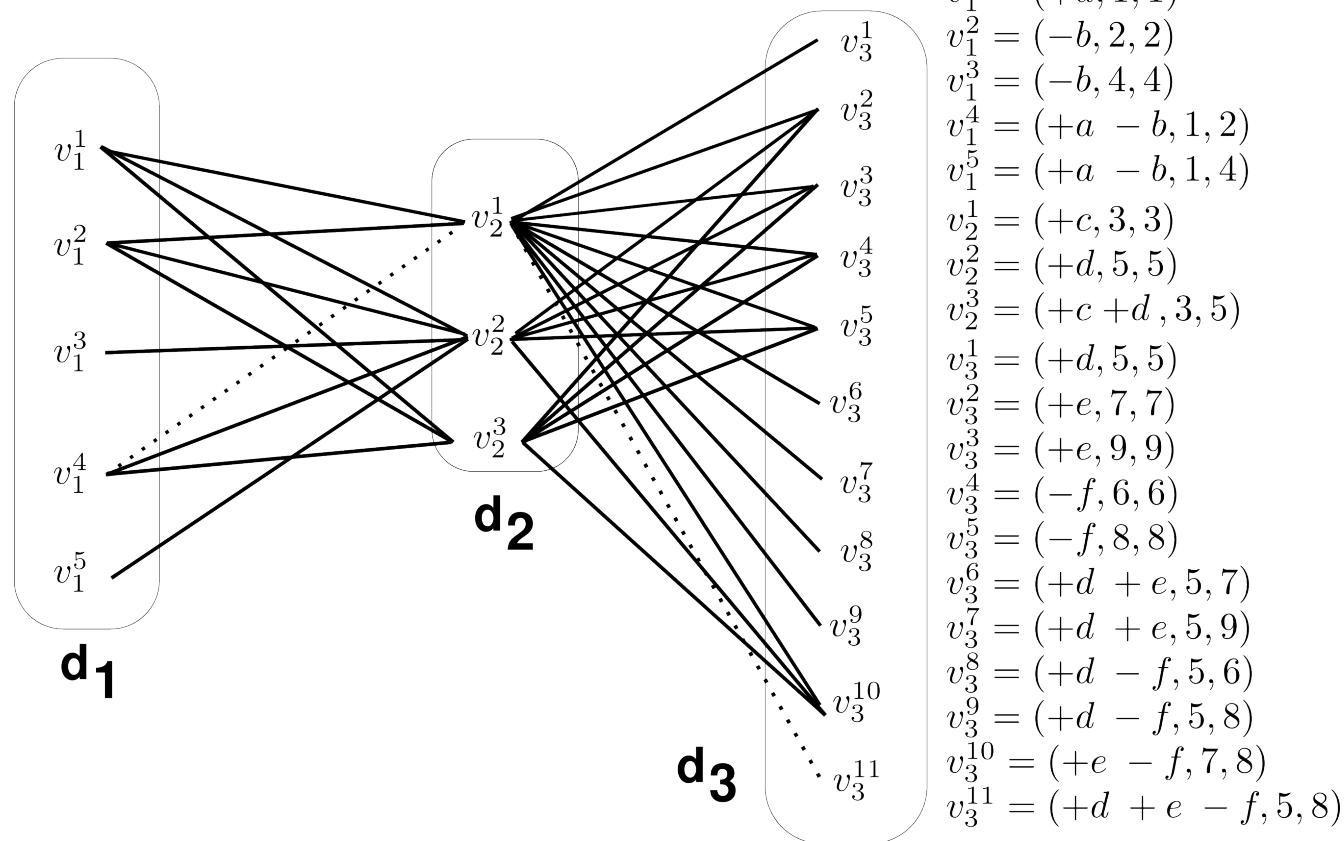
$$\mathbf{v_2^1 = (+c, 3, 3)}$$

Second FPT Algorithm

$$G_1 = \mathbf{+a -b} + a \mathbf{+c +d} + c + d \mathbf{+e +f}$$

$$G_2 = \mathbf{+a -b +c} - b \mathbf{+d} - f \mathbf{+e -f} + e$$

- ▶ Looking for a path in the DAG
- ▶ On the whole, time complexity of $\mathcal{O}(n2^{2s}s^3)$



Conclusion

- ▶ The EXEMPLAR BREAKPOINT DISTANCE problem cannot be approximated at all if both genomes contain duplicates
- ▶ The ZERO EXEMPLAR BREAKPOINT DISTANCE problem is NP-complete
- ▶ Two FPT algorithms for the ZERO EXEMPLAR BREAKPOINT DISTANCE problem
- ▶ Is there a constant ratio for $EBD(1,q)$? ($ZEBD(1,q)$ is polynomial, $EBD(1,q)$ is APX-hard [Angibaud *et al.* 2008])

Questions on The EXEMPLAR BREAKPOINT DISTANCE is not approximable for genomes with duplicated genes ?

Guillaume Blin¹ Guillaume Fertin² Florian Sikora¹
 Stéphane Vialette¹

¹Université Paris-Est, LIGM - UMR CNRS 8049 - France
 {gblin,sikora,vialette}@univ-mlv.fr

²Université de Nantes, LINA - UMR CNRS 6241 - France
 guillaume.fertin@univ-nantes.fr

WALCOM February 2009