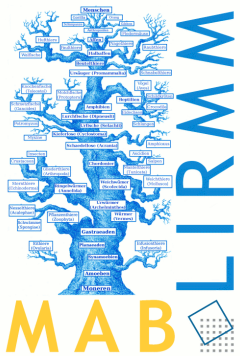
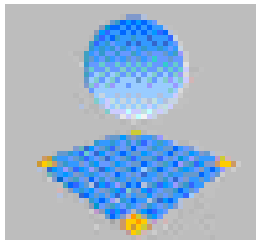
Novel definition and algorithm for chaining fragments with proportional overlaps

Raluca Uricaru, Alban Mancheron, Eric Rivals

LIRMM, CNRS Université de Montpellier 2

11 october 2010





Summary

Summary

Problem and Motivation

High resolution comparison of complete genomic sequences.

A real need for reliable WGA methods, knowing that :

- data accumulates excessively fast ;
- WGA is an essential process for extracting information about function, evolution and particularities of the studied genomes.

WGA Applications

- functional annotation in new genomes ;
- genome organization ;
- mechanisms of genome evolution ;
- phylogenomic and phylogenetic studies.

WGA in Bacterial Species

Examine the **mosaic** structure of bacterial genomes :

- backbone : conserved parts between sequences indicating **common biological components**

  genes, motifs, signals ;

- variable segments : differences between sequences probably associated with **strain-specific pathogenicity**

  mobile elements : prophages, insertion sequences

WGA – a complex problem

WGA brings new challenges :

- transition from classical to **large scale** sequence alignment ;
- dealing with a **big number** of sequences and high levels of **divergence** ;
- adjustment of a great number of **parameters** ;
- creation of **datasets** and **protocols** for evaluating the correctness and performance of WGA tools

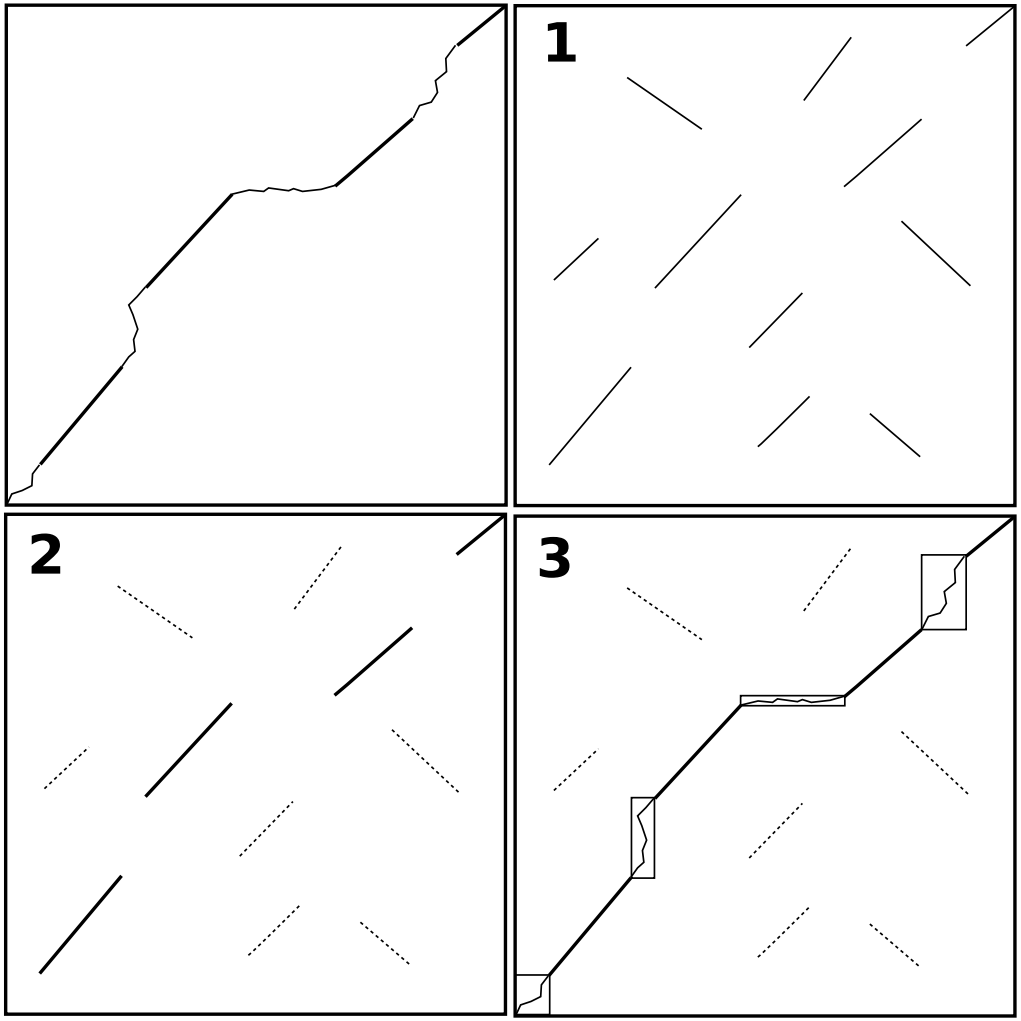
  see the robustness study by Hugo.

Pairwise WGA - with anchors

3-phase heuristic, the "anchor based strategy" :

1. computation of local similarities, i.e. ***fragments*** ;
2. fragment chaining
   maximal scoring chain of ***collinear/unconstrained*** *non-overlapping fragments* ;
3. apply recursively the first 2 phases +
   force the alignment of yet not aligned regions.

Anchor Based Strategy



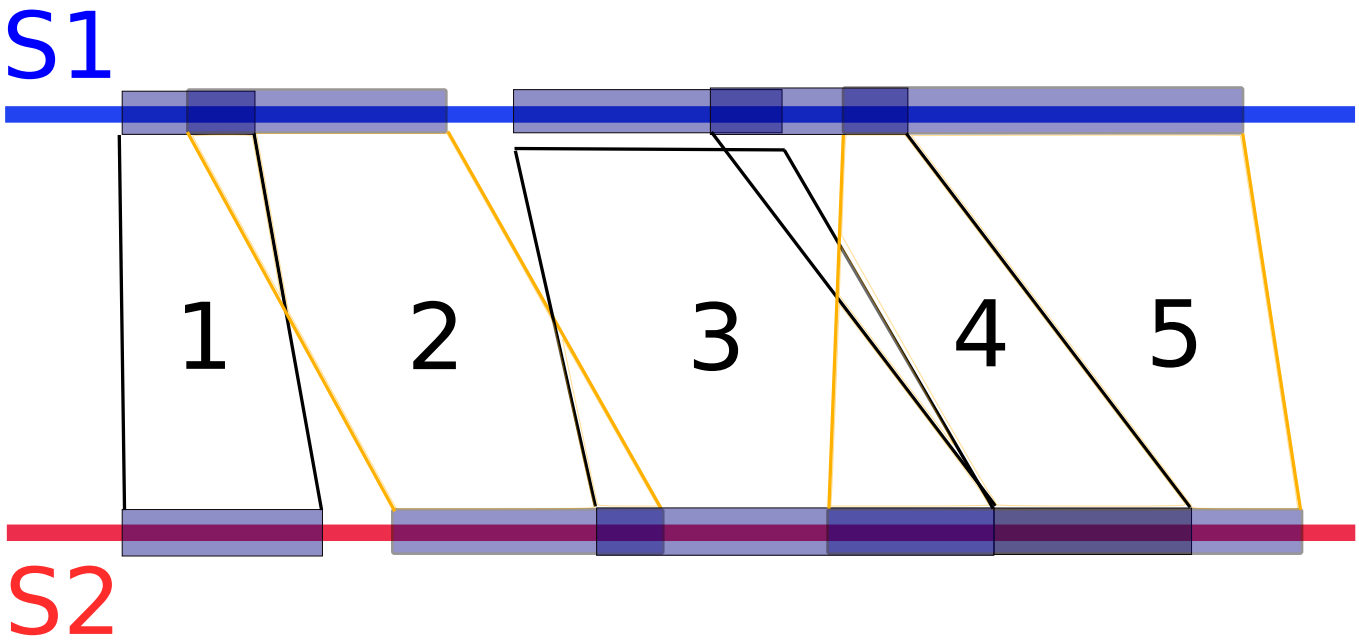
Improving the *anchor-based strategy*

1. work on the first phase
   *mainly responsible for the **sensitivity** of the method* ;
2. replace MEM-like fragments with real **local similarities** (LS) ;
3. classical fragment chaining + LS ; **ok ?**

Summary

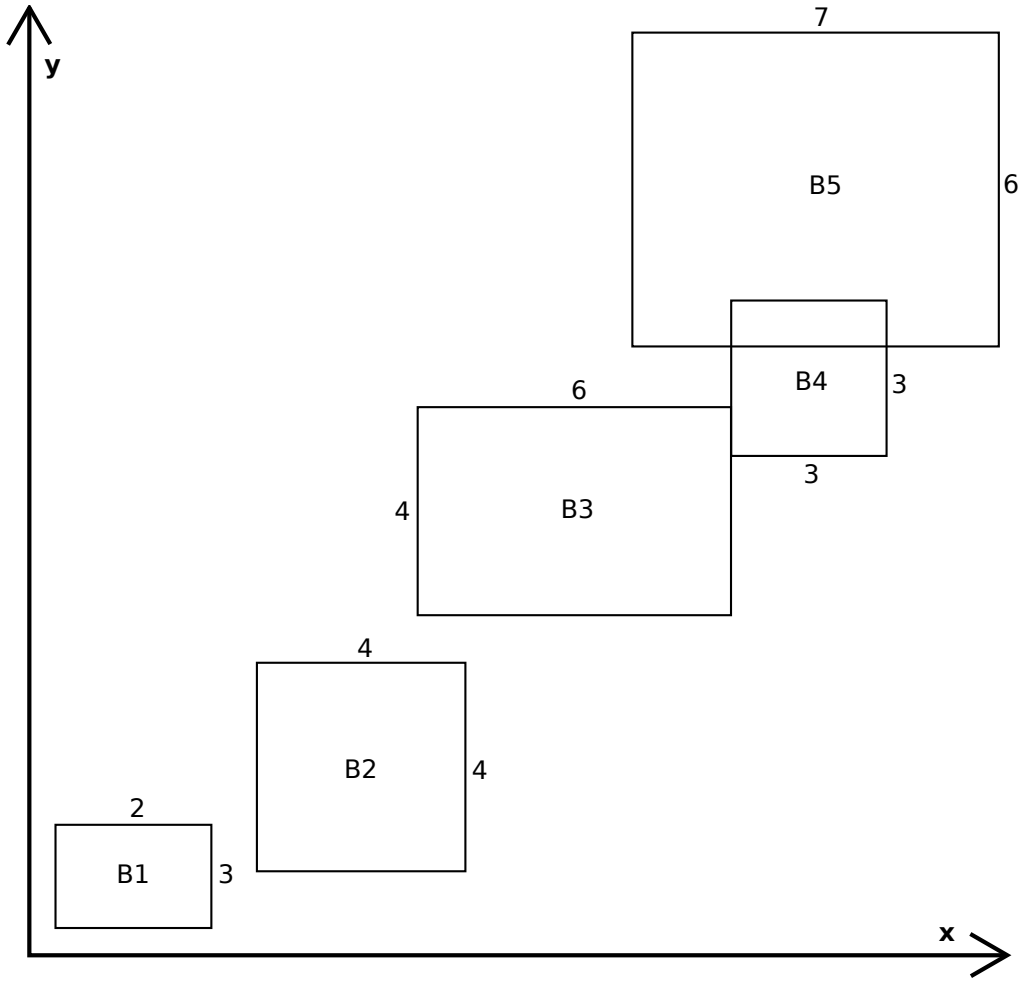
Fragment Chaining

Trapezoid Representation



Fragment Chaining

Box Representation



Notation and Definitions

Notation :

- $\alpha \in \{1, 2\}$ index the axis ;
- $<$: the classical **dominance order** between points of $\mathbb{R}^2$;
- $u(B)$, resp. $l(B)$: **upper**, resp. **lower**, corner of box B ;
- $P_\alpha(.)$: projection on axis α for a point $x \in \mathbb{R}^2$;
- $P_\alpha(B)$: an interval corresp. to the projection on α for a box B.

Definition [Overlap free box dominance order]

Let B_x, B_y be two boxes of $\mathbb{R}^2$.
We say that B_y **dominates** B_x, i.e. $B_x \ll B_y$, if $u(B_x) < l(B_y)$ in $\mathbb{R}^2$.

If neither $B_x \ll B_y$, nor $B_y \ll B_x$, then B_x and B_y are **incomparable**.

Definition of Overlap Free Chaining

Input :

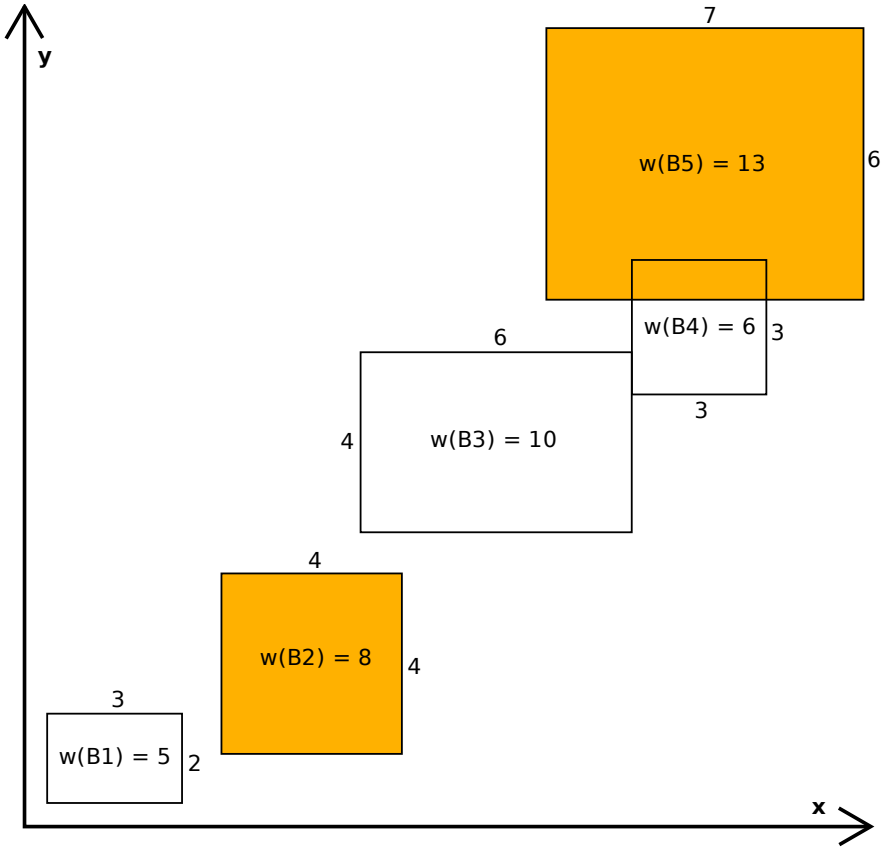
- $\mathcal{B}' := \{B_2, \ldots B_{n-1}\}$ the set of input boxes ;
- two dummy boxes, B_1, B_n, such that $B_1 \ll B_i \ll B_n$;
- we set $w(B_1) = w(B_n) := 0$;
- the input consists in $\mathcal{B} := \{B_1, \ldots, B_n\}$.

Definition [Overlap Free Chaining problem]

Let $\mathcal{B} := \{B_1, \ldots, B_n\}$ a set of boxes and let the weight of each box be constant and given by $w(B_i)$ for $1 \leq i \leq n$.
A **chain** C of boxes from $\mathcal{B}$ is a set of mutually comparable boxes.

Find in $\mathcal{B}$ a **chain** C ending in B_n, which **maximizes** $\sum_{B \in C} w(B)$.

Overlap Free Chaining



$$W(C) = w(B_2) + w(B_5) = 8 + 13 = 21$$

Overlap Free Chaining Solutions

- equivalent to the maximum weighted independent set in the intersection graph corresponding to the trapezoid/box representations above ;

  [S. Felsner et al, *Trapezoid graphs and generalizations, geometry and algorithms*, 1995.]

- DP algorithm in $O(n \log(n))$ [Felsner et al., 95]
  based on the sweep line paradigm ;

- solution with reduced time and space complexity
  [Abouelhoda and Ohlebusch, 05] - **Chainer**

Overlap Free Chaining Limitations

- due to biological and methodological reasons, **LS overlap**
  tandem repeats, extension phase ;
- *e.g.* 2 strains of *S. aureus* :
  - overlap free chain **interrupted** by 17 holes ($> 10Kbp$ each)
  - in 14 holes one **large fragment** was not included due to overlaps
  - overlaps $> 1bp$ and $< 1.8kbp$

Summary

How to take overlaps into account

- allow overlaps but do not count them twice on the genomes *subtract overlaps from the chain weight* ;
- maximize the **union of intervals**, i.e. *boxes projections* ;
- **maximal allowed overlap size ?** (variable overlap lengths) ;
- let $r \in [0; 1[$ be the **maximal allowed overlap ratio** overlaps allowed if the intervals do not overlap by more than $r\times$ their lengths ;
- introduce the **tolerant dominance order**.

Introducing Proportional Overlaps

Definition [r tolerant dominance order]

- Let B_u and B_v be two boxes. B_v *dominates* B_u *on axis* α in this **tolerant dominance order**, denoted by $B_u \ll_{r,\alpha} B_v$, iff

$$P_\alpha(u(B_u)) - P_\alpha(l(B_v)) \leq r \min(|P_\alpha(B_u)|, |P_\alpha(B_v)|).$$

- We denote by $B_u \ll_r B_v$ the fact that B_v **dominates** B_u iff for each $\alpha \in \{1, 2\}$, $B_u \ll_{r,\alpha} B_v$.

Weights

Definition [Weight of a box, of a chain]

- Let B be a box and $\alpha \in [1,2]$. Its weight on axis α is $w_\alpha(B) := |P_\alpha(B)|$

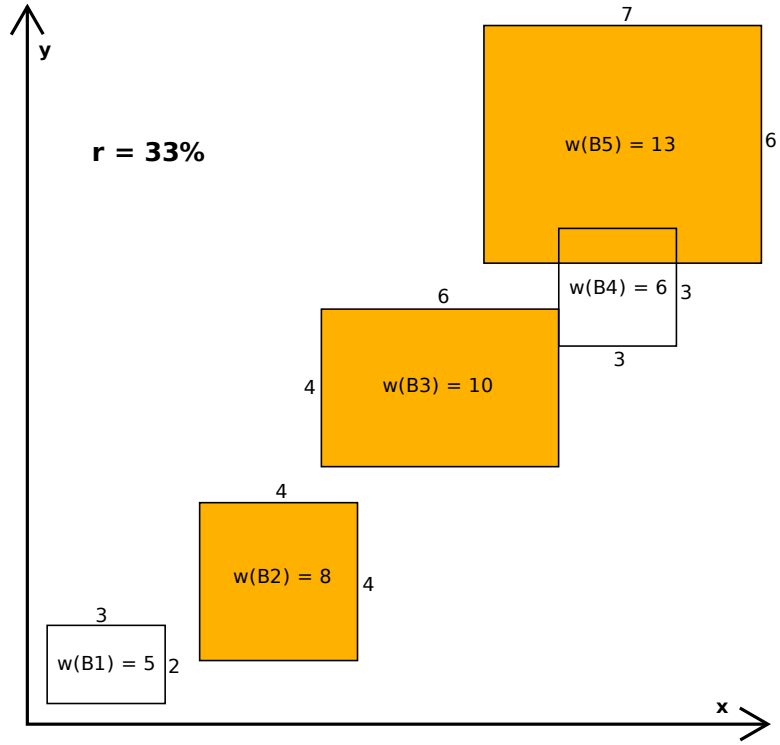
  Box weight : $w(B) := \sum_{\alpha=1}^{2} w_\alpha(B)$.

- Let $C := (B_1 \ll_r \ldots \ll_r B_m)$ be a chain. The weight of C on axis α, denoted $W_\alpha(C)$, is $W_\alpha(C) := \left|\bigcup_{i=1}^{m} P_\alpha(B_i)\right|$,

  Chain weight : $W(C) := \sum_{\alpha=1}^{2} W_\alpha(C)$.

Chaining with Overlaps

Box Representation



$$W(C) = \sum_{\alpha=1}^{2} |P_\alpha(B_2) \cup P_\alpha(B_3) \cup P_\alpha(B_5)| = 28$$

Properties of r-tolerant dominance order

Prop. [Corners]

Let B_t, B_u two boxes such that $B_t \ll_r B_u$. Then $l(B_t) < l(B_u)$ and $u(B_t) < u(B_u)$.

Prop. [Transitiveness]

The dominance order $\ll_r$ is transitive.

Corollary

Let B_t, B_u, B_v be three boxes such that $B_t \ll_r B_u \ll_r B_v$. Then $(B_t \cap B_v) \subset (B_u \cap B_v)$.

Definition of Chaining with Overlaps

Definition (Chaining With Proportional Overlaps problem)

Let $r \in [0, 1[$ and $\mathcal{B} := \{B_1, \ldots, B_n\}$ a set of boxes.

Find in $\mathcal{B}$, according to the dominance order $\ll_r$, the **chain** C that ends in B_n and whose weight $W(C)$ is maximal.

DP solution based on a recurrence eq. to compute $W(B_i)$:

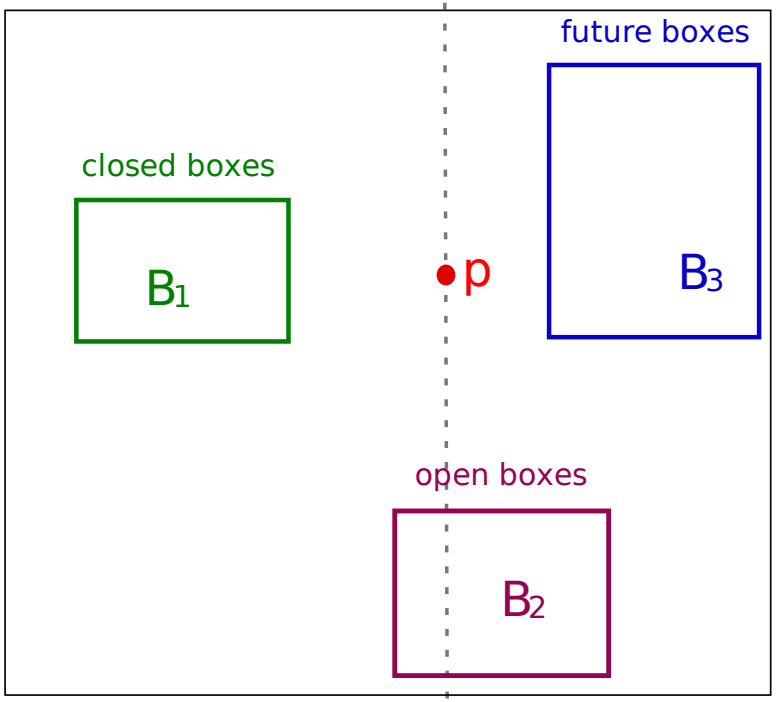
$$W(B_1) = 0$$

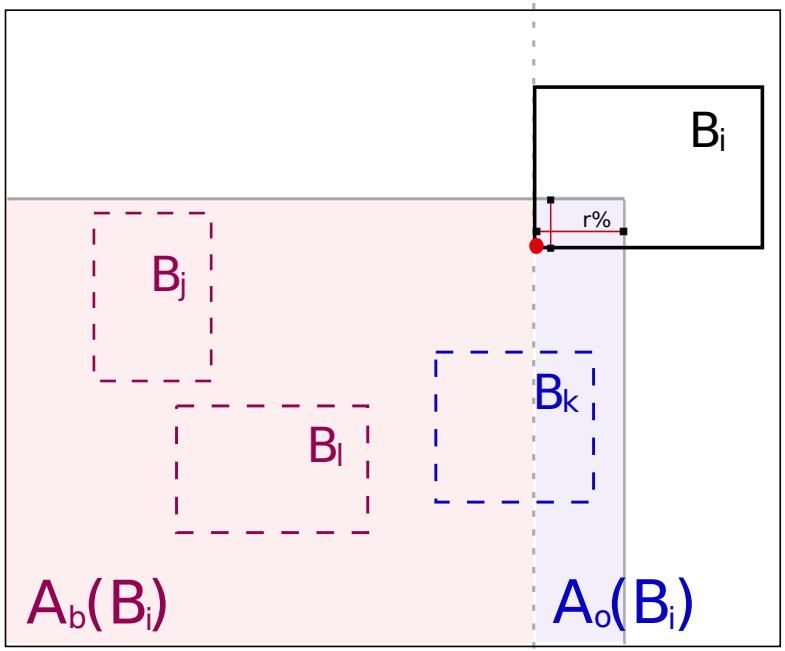
$$W(B_i) = \max_{B_j:\, B_j \ll_r B_i} W(B_j) + \sum_{\alpha=1}^{2} \left| P_\alpha(B_i) \setminus P_\alpha(B_j) \right|, 1 < i \leq n$$

(by Corollary)

Summary

Box partition and areas





Two dynamic sets

We maintain two dynamic sets ; at each point :

- Open boxes : $\mathcal{O}$, all boxes cut by the sweep line ;
- Active boxes : $\mathcal{A}$, the set of interesting predecessors for all future boxes.

Sweep-line Solution for Chaining With Overlaps

Algorithm 1: Maximum Weighted Chain with Overlaps

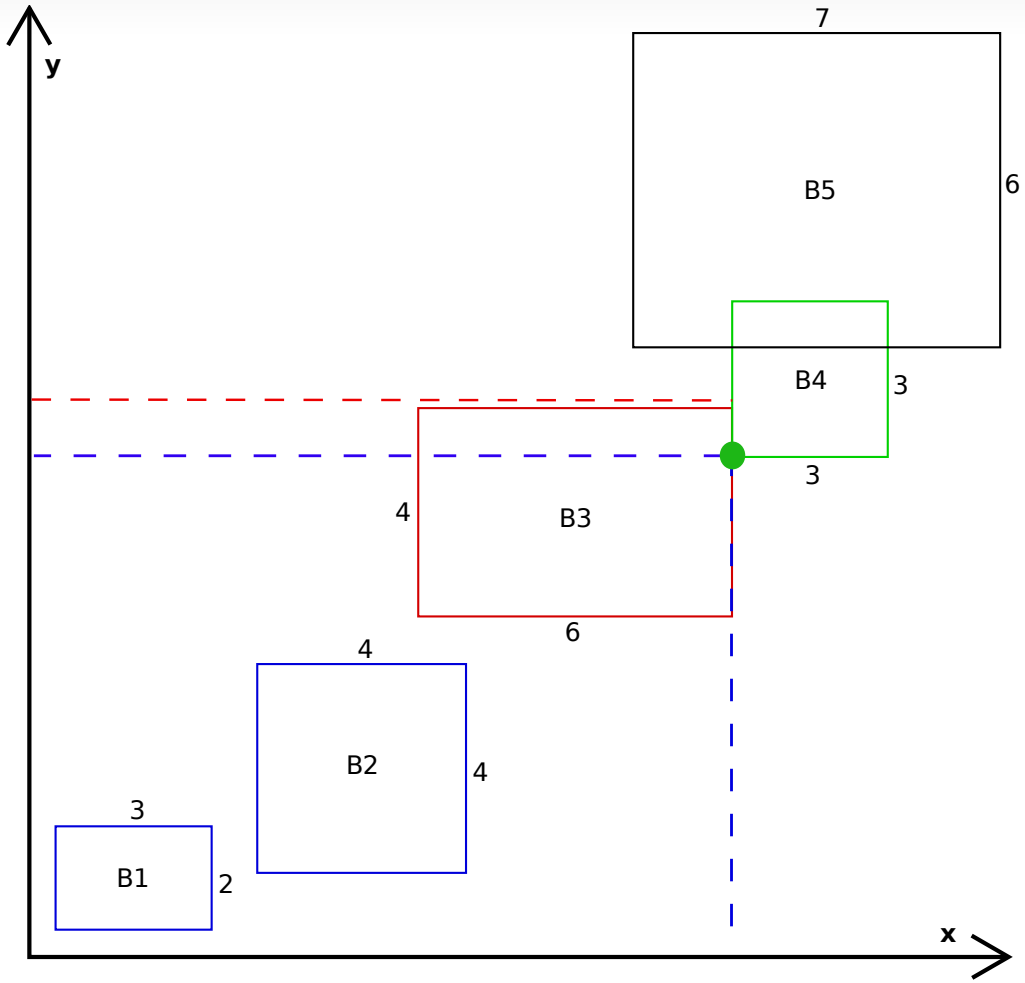
Data: $\mathcal{P}$ a ordered set of $2n$ points corresponding to the boxes' corners
Result: *Prev* a vector of previous boxes in the maximum weighted chain
begin
  foreach $p \in \mathcal{P}$ **in ascending order on** x**-coordinate do**
    if $p = l(B_i)$**, i.e. it is the lower corner of** B_i **then**
      ▸ Case 1 *Compute* B_j*, the best previous box ending before* B_i ;
        /* B_j may overlap B_i on y-axis */
    else /* $p = u(B_i)$, i.e. it is the upper corner of B_i */
      ▸ Case 2 *Update the weight and the predecessor of opened boxes* ;
        /* deal with overlaps on x-axis */
        Update the list of potential predecessors ;
  traceback(*Prev*[$\top$]);
end

Sweep-line Solution for Chaining With Overlaps

Algorithm 2: Maximum Weighted Chain with Overlaps

Data: $\mathcal{P}$ a ordered set of $2n$ points corresponding to the boxes' corners
Result: *Prev* a vector of previous boxes in the maximum weighted chain
begin
  foreach $p \in \mathcal{P}$ **in ascending order on** x**-coordinate do**
    if $p = l(B_i)$**, i.e. it is the lower corner of** B_i **then**
      ▸ Case 1 *Compute* B_j*, the best previous box ending before* B_i ;
      `/* ` B_j `may overlap` B_i `on` y`-axis */`
    else `/* ` $p = u(B_i)$`, i.e. it is the upper corner of` B_i `*/`
      ▸ Case 2 *Update the weight and the predecessor of opened boxes* ;
      `/* deal with overlaps on` x`-axis */`
      Update the list of potential predecessors ;
  traceback(*Prev*[$\top$]);
end

Case 1 Lower corners



Algorithm

Case 1 Lower Corner of B_i

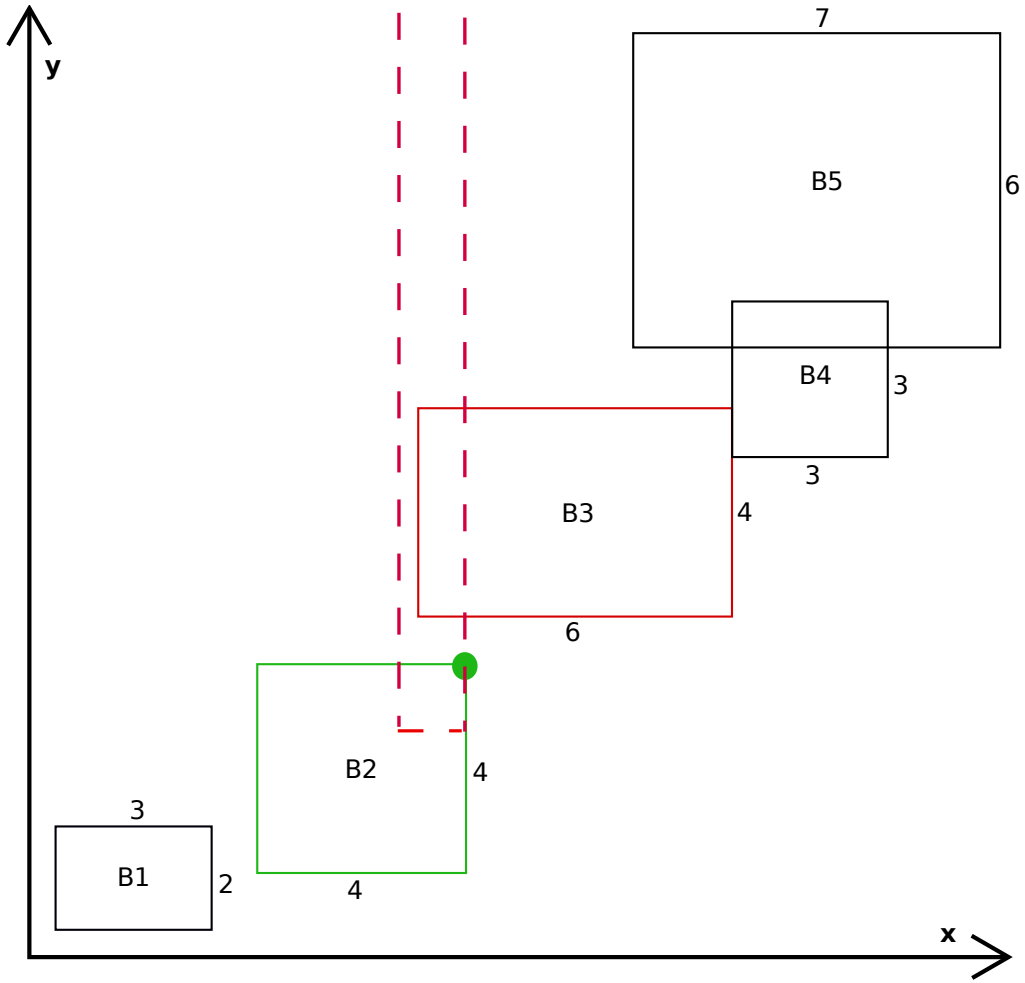
$$\mathcal{O} \longleftarrow \mathcal{O} \cup \{B_i\}$$

$$Pred[B_i] \longleftarrow \underset{B_j \ll_r B_i, B_j \in \mathcal{A}}{\arg\max} \left(W[B_j] + \sum_{\alpha=1}^{2} \left|P_\alpha(B_i) \setminus P_\alpha(B_j)\right|\right)$$

$$W[B_i] \longleftarrow W[Pred[B_i]] + \sum_{\alpha=1}^{2} |P_\alpha(B_i) \setminus P_\alpha(Pred[B_i])|$$

▸ Algorithm

Case 2 Upper Corners



Case 2 Upper Corner of B_i

$\mathcal{O} \longleftarrow \mathcal{O} \setminus \{B_i\}$

foreach $B_k \in \mathcal{O}$ with $B_i \ll_r B_k$ **do**
    update $W[B_k]$ and $Pred[B_k]$ *if necessary* ;

eventually ***add*** B_i *to* $\mathcal{A}$;

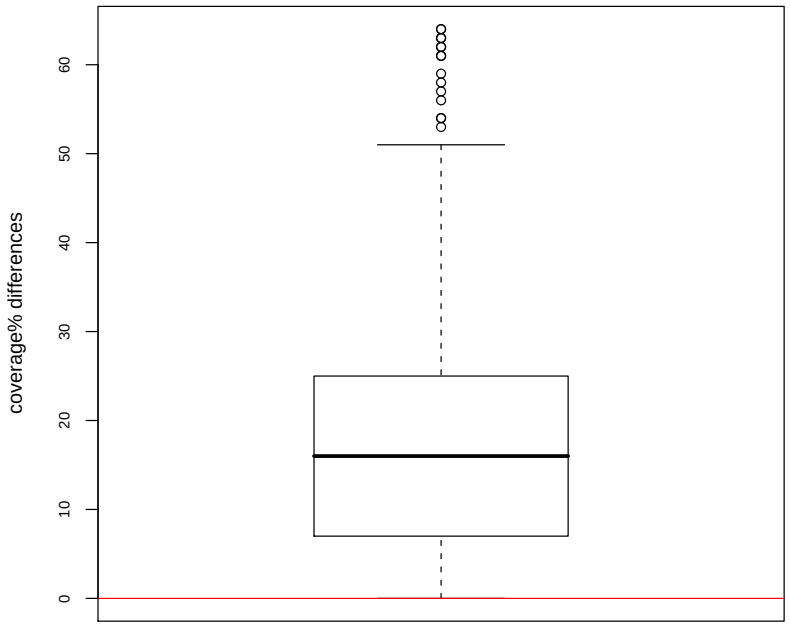
delete *all useless boxes from* $\mathcal{A}$.

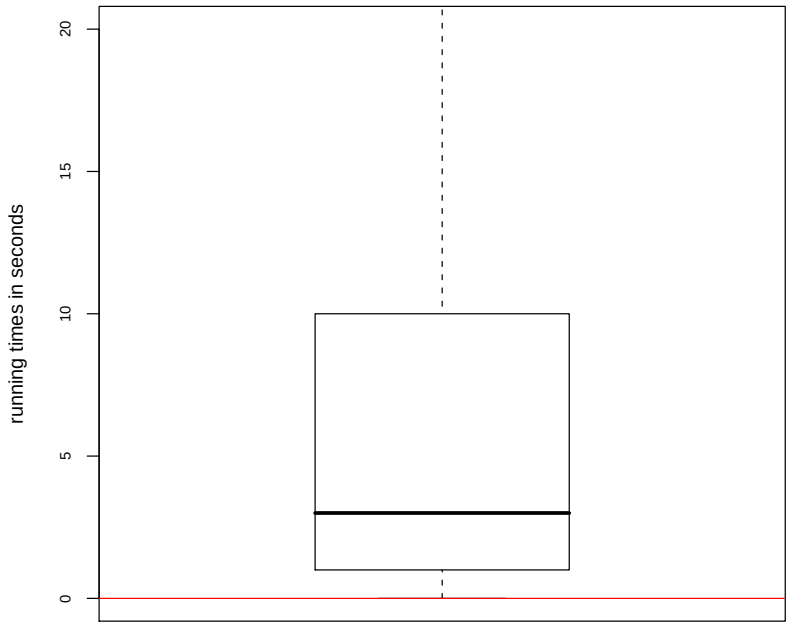
Summary

XPs

- 694 **intra-species bacterial genomes** pairwise comparisons ;
- **overlap ratio** : $r = 0.1$
- **Chainer** vs **OverlapChainer**
- differences in **coverage%**
  1% of additional cov% $\approx 28Kbp$
- **running times**
  Chainer $< 1s$ in average, max $17s$

XPs


coverage% differences
0
10
20
30
40
50
60


running times in seconds
0
5
10
15
20

Conclusions

- Formulation of **Chaining with Proportional Overlaps problem** ;
- $O(n^2)$ algorithms : DP and **sweep line** ;
- Sweep line outperforms DP algorithm in running time
- Important improvement of **genome coverage** on bacterial strains comparisons

Future

- Robustness and impact of ratio of allowed overlaps
- Biological causes of long overlaps
- Average complexity of sweep line algorithm

Support and thanks :

- ANR project : CoCoGEN
  http://www.lirmm.fr/~rivals/CoCoGEN
- INRA, *Jouy-en-Josas*

C. Michotey, H. Chiapello

- E.T.H, *Zurich*

C. Dessimoz

Thank you for your attention ! Questions ?