

91學年度上學期「生物資訊學」課程

---

# Comparison & Integration

張傳雄

國立陽明大學 遺傳學研究所

12-16-2002



# Comparative Genomics

---

- The practice of analyzing & comparing the different genomes.

## **For the purpose of:**

- understanding the similarity & difference between the genomes that lead to special phenotypes or diseases.
- identifying genes and discovering their functions by studying their counterparts in other organisms.
- revealing the evolutionary relationships between different organisms.

# What to compare?

---

- Size of the genome: total number of base pairs
- Overall (G+C) content: percentage of (G+C)
- Regions of different GC content
- Total number of predicted ORFs
- Percentage of the genome (coding)
- Average length of ORF
- Functional assignment
  - function assigned by homology (known genes)
  - homologue found but no function assigned (conserved hypothetical genes)
  - no homolog found (hypothetical genes)
- Paralogs & orthologs
- Genomic organization & gene location/order
  - repeats, inversion, & translocation

# Genome sequencing projects

---



GOLD™  
Genomes OnLine Database



Contact: <u>GOLD</u>	Last Update: December 9, 2002	Sponsored by <u>Integrated Genomics Inc.</u>
	<u>Search GOLD:</u> 699 genome projects	
115 <u>Published Complete Genomes</u> including 2 chromosomes	349 <u>Prokaryotic Ongoing Genomes</u>	235 <u>Eukaryotic Ongoing Genomes</u> including 8 chromosomes

<http://wit.integratedgenomics.com/GOLD/>

# Some important model organisms

---

Mammals: *Homo sapiens*, Chimpanzee, mouse, rat

Fish: Zebrafish, Pufferfish

Insects: Fruitfly (*Drosophila melanogaster*)

Roundworms: *Ceanorhabditis elegans*

Protista: Malaria parasite (*Plasmodium falciparum*)

Fungi: Yeast (*Saccharomyces cerevisiae*, *S. pombe*)

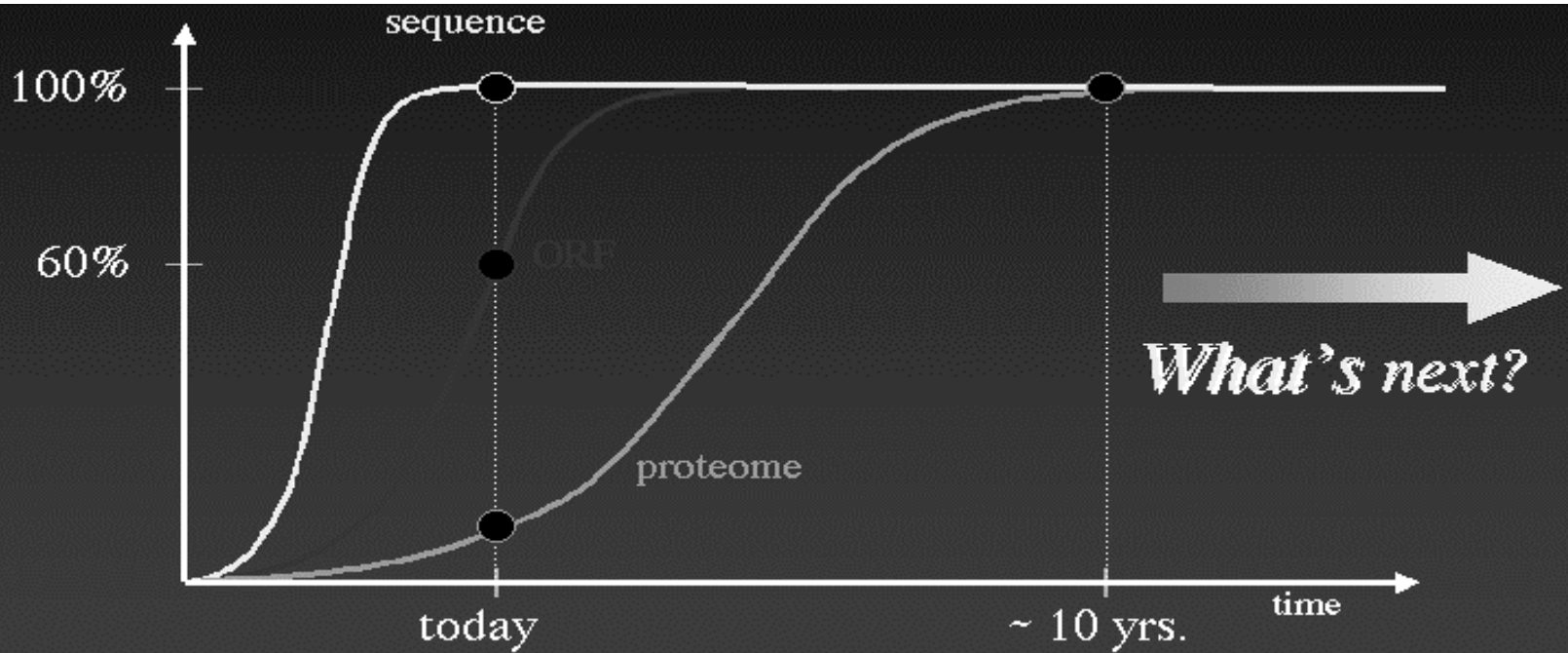
Plants: *Arabidopsis thaliana*, corn, rice

Bacteria: *Escherichia coli*, salmonella

Archea: *Methanococcus janaschii*

# Trends of Genomic Research

---



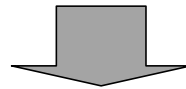


# GOLD™ Genomes OnLine Database



Contact: <u>GOLD</u>	Last Update: December 9, 2002	Sponsored by <u>Integrated Genomics Inc.</u>
	<u>Search GOLD:</u> 699 genome projects	
115 <u>Published Complete Genomes</u> including 2 chromosomes	349 <u>Prokaryotic Ongoing Genomes</u>	235 <u>Eukaryotic Ongoing Genomes</u> including 8 chromosomes

## Comparative Genomics



# *The Next Wave* of the Genomics Research



how we got to be  
**HUMAN**

## **What It Means to Be 98% Chimpanzee Apes, People, and Their Genes**

**Jonathan Marks**

**University of California Press, Berkeley, 2002**

**Hardback: 326 pp., illus. \$27.50, ?9.95. ISBN: 0-520-22615-1**

**Are we more like mice, or like cats?**

# Some questions

---

- Pathogenesis (virulence factors)
- Extremophiles
- What gene expression patterns differentiate one bacterium from another?
- Can we computationally design genomes with desirable properties?
- What genes were crucial for the evolution of new species?
- What variants in our DNA make us susceptible to specific diseases?
- more .....

Class	Species	Chromosome Number	Genome Size (bp)	Gene Number
Primate	<i>Homo sapiens</i>	2N=46	2.9 x10 <sup>9</sup>	30,000
Rodent	<i>Mus musculus</i>	2N=40	2.5 x10 <sup>9</sup>	30,000
Insect	<i>Drosophila melanogaster</i>	2N=8	1 x10 <sup>8</sup>	20,000
Nematode	<i>Caenorhabditis elegans</i>	N=6	0.97 x10 <sup>8</sup>	13,100
Fungi	<i>Schizosaccharomyces pombe</i>	3	0.14 x10 <sup>8</sup>	5,400
	<i>Saccharomyces cerevisiae</i>	16	0.12 x10 <sup>8</sup>	6,183
Higher plants	<i>Arabidopsis thaliana</i>	2N=10	1.3 x10 <sup>8</sup>	20,000-25,000
	<i>Oryza sativa</i>	2N=24	4.5 x10 <sup>8</sup>	40,000
Cyanobacteria	<i>Synechocystis PCC6803</i>	1	4 x10 <sup>6</sup>	2,400
Gram-positive bacteria	<i>Bacillus subtilis</i>	1	4.2 x10 <sup>6</sup>	4,100
	<i>Mycoplasma genitalium</i>	1	0.58 x10 <sup>6</sup>	470
Gram-negative bacteria	<i>Escherichia coli</i>	1	4.6 x10 <sup>6</sup>	4,000
	<i>Haemophilus influenzae</i>	1	1.8 x10 <sup>6</sup>	1,743
Archaeobacteria	<i>Pyrococcus shinkaii OT3</i>	1	2 x10 <sup>6</sup>	2,000
Spirochete	<i>Borrelia burgdorferi</i>	1	0.91 x10 <sup>6</sup>	1,700

Modified from <http://www.jaist.ac.jp/~ckim/gsize.html>

# Problems with existing sequence alignment algorithms for genomic analysis

---

- Most algorithms were developed for comparing single protein sequences or DNA sequences containing a single gene
- Most algorithms were based on assigning a score to all the possible alignments (usually by the sum of the similarity/identity values for each aligned residue minus a penalty for the introduction of gaps) and then finding the optimal or near-optimal alignment based on the chosen scoring scheme.
- Unfortunately, most of these programs cannot accurately handle long alignments.
- Linear-space type of Smith-Waterman variants are too computationally intensive requiring specialized hardware (memory-limited) or very time-consuming. Higher speed vs increased sensitivity.

# Challenges of whole genome comparison

---

Need methods that can scale well to

- align large size of the DNA sequences (millions of bp)
  - Dynamic programming suitable for pairwise alignment of small sequences
  - Heuristic algorithms (e.g., BLAST) suitable for local alignment of sequences against large databases
  - Program needs to be time & space efficient (i.e., it needs to be linear or close to linear). Aligning genomic sequences in linear time. [Memory & Speed!]
- identify large scale changes (i.e., matches between genomes as well as various non-match features):
  - occurrence of both short & long insertions & deletions
  - large-scale changes such as tandem repeats & large-scale reversals
  - high degree of divergence in the 3rd position of codons

# Genome-size comparative alignment tools

- MUMmer - Maximal Unique Match (mer)
  - <http://www.tigr.org/softlab/> (Delcher et al. 1999)
- ASSIRC - Accelerated Search for SIMilarity Regions in Chromosomes
  - <ftp://ftp.biologie.ens.fr/pub/molbio/> (Vincens et al. 1998)
- BLAT –
  - <http://genome.ucsc.edu/cgi-bin/hgBlat?command=start> (Kent et al. 2002)
- DIALIGN - DIagonal ALIGNment
  - <http://www.gsf.de/biodv/dialign.html> (Morgenstern et al. 1998; Morgenstern 1999)
- DBA - DNA Block Aligner
  - <http://www.sanger.ac.uk/Software/Wise2/dba.shtml> (Jareborg et al. 1999)
- GLASS - GLobal Alignment SyStem
  - <http://plover.lcs.mit.edu/> (Batzoglou et al. 2000)
- LSH-ALL-PAIRS - Locality -Sensitive Hashing in ALL PAIRS
  - Email: [jbuhler@cs.washington.edu](mailto:jbuhler@cs.washington.edu) (Buhler 2001)
- MegaBlast
  - <http://www.ncbi.nih.gov/blast/> (Zhang 2000)
- PIPMaker - Percent Identity Plot MAKER
  - <http://biocse.psu.edu/pipmaker/> (Schwartz et al. 2000)
- **SSAHA – Sequence Search and Alignment by Hashing Algorithm**
  - <http://www.sanger.ac.uk/Software/analysis/SSAHA/>
- WABA - Wobble Aware Bulk Aligner
  - <http://www.cse.ucsc.edu/~kent/xenoAli/> (Kent & Zahler 2000)

# What's MUMmer?

---

- A system for aligning whole genome sequences based on heuristics.
- Using a suffix tree data structure, it's able to rapidly align sequences containing millions of nucleotides.
- Developed by S.L. Salzberg & A. Phillippy at TIGR (The Institute for Genomic Research) and A.L. Delcher at Celera Genomics.  
(<http://www.tigr.org/software/mummer/>)
- References:
  - *Alignment of whole genomes*, by Delcher et al., Nucleic Acids Research 1999, 27:2369-2376. (MUMmer 1.0)
  - *Fast algorithms for large-scale genome alignment & comparison*, by Delcher et al., Nucleic Acids Research 2002, 30:2478-2483. (MUMmer 2)

# Alignment of whole genomes

Arthur L. Delcher<sup>1,2</sup>, Simon Kasif<sup>3</sup>, Robert D. Fleischmann<sup>4</sup>, Jeremy Peterson<sup>4</sup>, Owen White<sup>4</sup> and Steven L. Salzberg<sup>4,\*</sup>

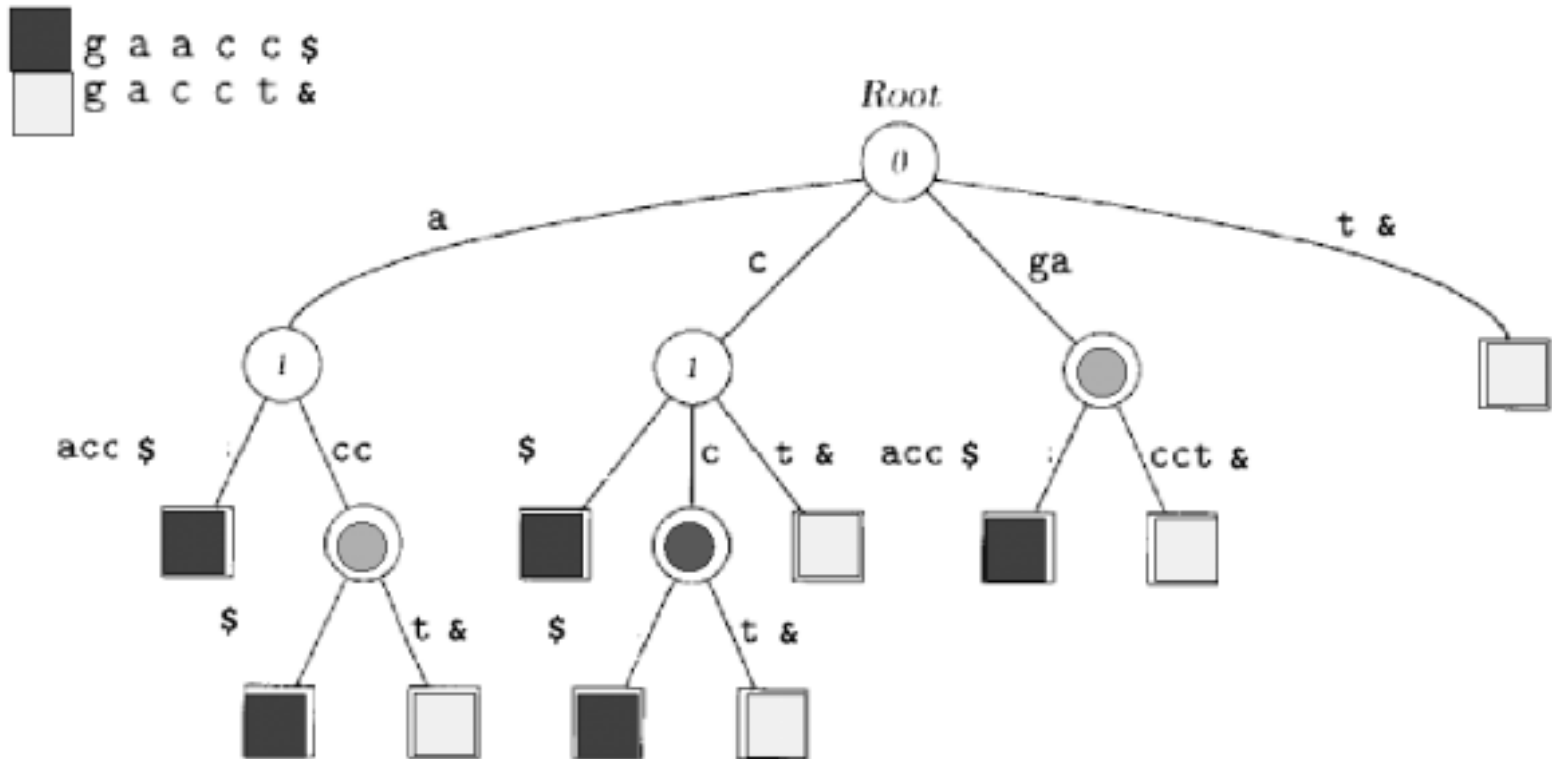
<sup>1</sup>Department of Computer Science, Loyola College in Maryland, Baltimore, MD 21210, USA, <sup>2</sup>Celera Genomics, 45 West Gude Drive, Rockville, MD 20850, USA, <sup>3</sup>Department of Electrical Engineering and Computer Science, University of Illinois, Chicago, IL 60607, USA and <sup>4</sup>The Institute for Genomic Research, 9712 Medical Center Drive, Rockville, MD 20850, USA

## ABSTRACT

A new system for aligning whole genome sequences is described. Using an efficient data structure called a suffix tree, the system is able to rapidly align sequences containing millions of nucleotides. Its use is demonstrated on two strains of *Mycobacterium tuberculosis*, on two less similar species of *Mycoplasma* bacteria and on two syntenic sequences from human chromosome 12 and mouse chromosome 6. In each case it found an alignment of the input sequences, using between 30 s and 2 min of computation time. From the system output, information on single nucleotide changes, translocations and homologous genes can easily be extracted. Use of the algorithm should facilitate analysis of syntenic chromosomal regions, strain-to-strain comparisons, evolutionary comparisons and genomic duplications.

# MUMmer 1.0:

Put two sequences in one suffix tree



- Check nodes with 2 daughters that are leaves from different sequences. It's a MUM if previous letter is different in the two sequences (e.g., acc & ga are MUMs).

# MUMmer 1.0

---

- Two strains of *Mycobacterium tuberculosis*
  - 4.4 Mbp each
  - 55 sec on DEC Alpha
- Two Mycoplasma genomes
  - 580 kbp vs. 816 kbp
  - 2 minutes (mostly for Smith-Waterman on gaps)
- Human & mouse
  - 223 kbp vs. 227 kbp
  - 29 sec

# Computation time vs. size & similarity

---

	Length	Sequence similarity	Step 1 (# sec)	Step 2 (# sec)	Step 3 (# sec)
<i>M. Tuberculosis</i> H37Rv Vs. <i>M. Tuberculosis</i> CDC1551	4Mb 4Mb	99% identical	5	45	5
<i>M. Genitalium</i> vs. <i>M. pneumoniae</i>	580Kb 816Kb	20Kb in MUMs of >15b; < 50%id in gap regions	6.5	0.02	116
Subsequences of Human chromosome 12p13 vs. Mouse chromosome 6	223kb 228kb	14kb in MUMs of >15b; Large gaps	1.6	29	?

# MUMmer 2

---

- Three times faster, uses 1/3 memory.
  - less memory per node in suffix tree data structure
  - insert only one genome into the tree, then “stream” second genome through tree, finding longest matches
- Can align & compare protein sequences.
- Has been used to find large scale ancient duplications in human chromosomes.
  - Reference: Venter et al. (2001), *The sequence of the human genome*. Science 291, 1304-1351.

# Fast algorithms for large-scale genome alignment and comparison

Arthur L. Delcher<sup>1,2</sup>, Adam Phillippy<sup>1</sup>, Jane Carlton<sup>3</sup> and Steven L. Salzberg<sup>3,4,\*</sup>

<sup>1</sup>Department of Computer Science, Loyola College in Maryland, Baltimore, MD 21210, USA, <sup>2</sup>Celera Genomics, 45 West Gude Drive, Rockville, MD 20850, USA, <sup>3</sup>The Institute for Genomic Research, Rockville, MD 20850, USA and <sup>4</sup>Department of Computer Science, Johns Hopkins University, Baltimore, MD, USA

## ABSTRACT

We describe a suffix-tree algorithm that can align the entire genome sequences of eukaryotic and prokaryotic organisms with minimal use of computer time and memory. The new system, MUMmer 2, runs three times faster while using one-third as much memory as the original MUMmer system. It has been used successfully to align the entire human and mouse genomes to each other, and to align numerous smaller eukaryotic and prokaryotic genomes. A new module permits the alignment of multiple DNA sequence fragments, which has proven valuable in the comparison of incomplete genome sequences. We also describe a method to align more distantly related genomes by detecting protein sequence homology. This extension to MUMmer aligns two genomes after translating the sequence in all six reading frames, extracts all matching protein sequences and then clusters together matches. This method has been applied to both incomplete and complete genome sequences in order to detect regions of conserved synteny, in which multiple proteins from one organism are found in the same order and orientation in another. The system code is being made freely available by the authors.

# MUMmer

---

- Assumes sequences are closely related.
- Performs high resolution comparison.
- Outputs base-to-base alignment.
- Identify the following features:
  - single nucleotide polymorphisms (SNPs)
  - regions of divergence  $> 1$  SNP
  - large inserts
  - repeats
  - tandem repeats: two or more adjacent, approximate copies of a DNA pattern
  - reversals

# MUMmer

---

- MUMmer is a system for aligning entire genomes extremely rapidly. For example, MUMmer v 2.1 can align two completely bacterial genomes of 3-4 million base pairs (Mbp) each in under 30 seconds using less than 100 Mb of memory on a typical desktop computer.
- MUMmer can also align incomplete genomes; it handles the 100s or 1000s of contigs from a shotgun sequencing project with ease, and will align them to another set of contigs or a genome, using the NUCmer utility included with the system. The PROmer utility takes this a step further by generating alignments based upon the six-frame translations of both input sequences. PROmer permits the alignment of genomes for which the proteins are similar but the DNA sequence is too divergent to detect similarity.

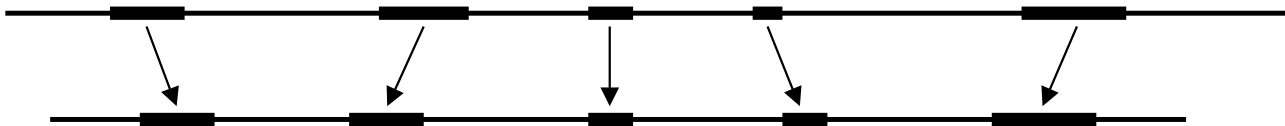
# Main idea

---

Genomic regions of interest contain ordered islands of similarity

– E.g. genes

1. Find local alignments
2. Chain an optimal subset of them



# Algorithmic methods

---

MUMmer is a combination of 3 ideas:

- Find local alignments using suffix tree.
- Longest Increasing Subsequence (LIS).
- Chain local alignments using Smith-Waterman.

# Three steps

---

- Identify all Maximal Unique Matches (MUMs) between two sequences.
- Sort the MUMs by position on the first genome, & find longest possible set of matches in the same order on both genomes (i.e., extract the longest set of matches that occur in the same order in both genomes).
- Close the local gaps by identifying inserts, repeats, tandem repeats, small mutated regions, & SNPs and perform local alignment on portion between the aligned MUMs.



Output the alignment

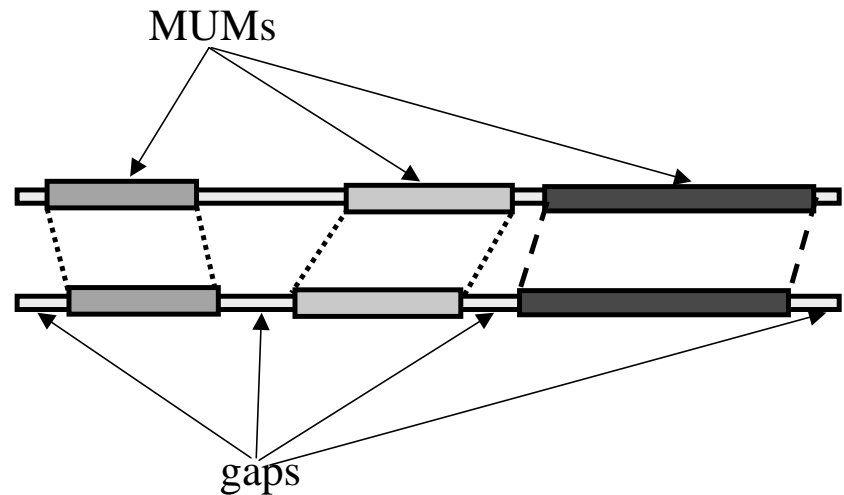
(including MUM alignment & detailed alignment of non-MUM regions)

# MUMmer overview

---

- algorithm

- Finding MUMs
- Matching MUMs
- Closing gaps

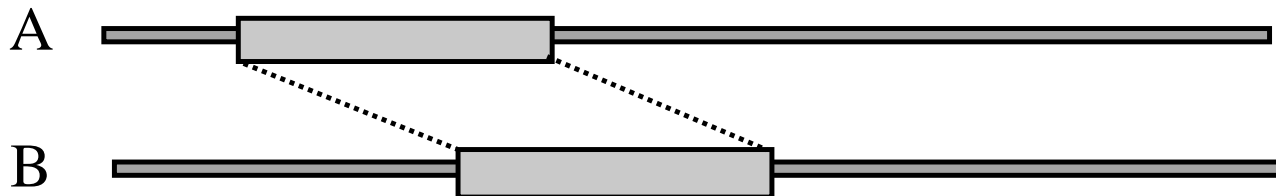


# Maximal unique matching subsequences

---

- Sequences in genomes A and B that :
  - occur exactly once in A and in Bare not contained in any larger such sequence

Genome A: tcgatcGACGATCGCGGCCGTAGATCGAATAACGAGAGAGCATAAcgactta  
Genome B: gcattaGACGATCGCGGCCGTAGATCGAATAACGAGAGAGCATAAtccagag



- A MUM cannot be extended. Any extension of the MUM will result in a mismatch. (the sequences have different letters on each side of the MUM).
- By definition, a MUM does NOT occur anywhere else in either genome.

# Step 1: Identify all MUMs

---

MUM1

MUM2

A: tcgatcaAGCTCTGATatgtaccatacgtgaATCGACGTACATGtactgta

B: agcgAGCTCTGATcctgcatcaagATCGACGTACATGatgaat

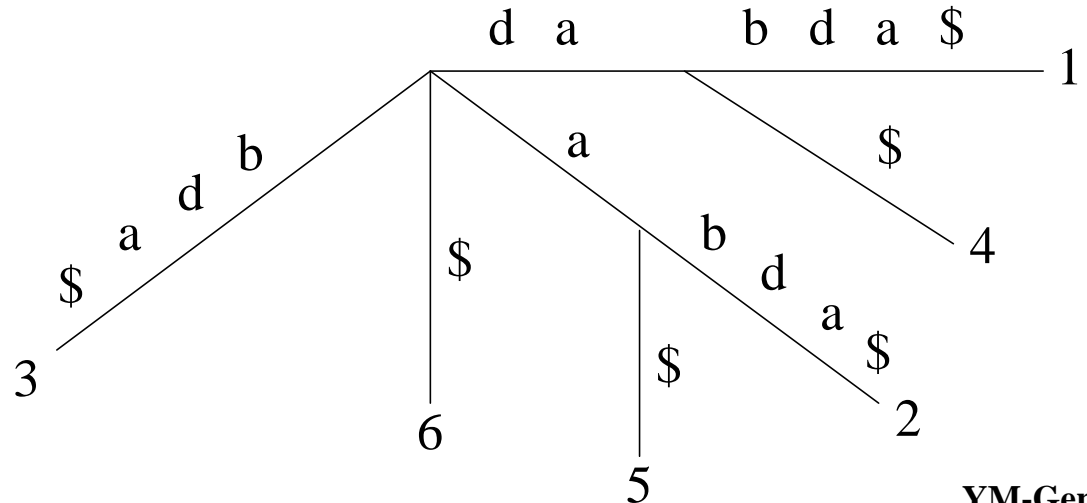
- A Maximal Unique Match (MUM) is a subsequence that occurs exactly once in Genome A & once in Genome B, & is NOT contained in any longer such sequence.

# Suffix tree

- The key idea in identifying MUMs is to build a suffix tree for genomes A & B. Suffix tree is a method to find all maximal matches between two strings.

Example of a suffix tree construction for  $x = d a b d a \$$

1. Insert  $d a b d a \$$
2. Insert  $a b d a \$$
3. Insert  $b d a \$$
4. Insert  $d a \$$
5. Insert  $a \$$
6. Insert  $\$$





# Definition of a suffix tree

---

- A suffix tree is a compact representation that stores all possible suffixs of an input sequence.
- For string  $x = x_1 \dots x_m$ 
  - A rooted tree with  $m$  leaves
    - Leaf  $i$ :  $x_i \dots x_m$
  - Each edge is a substring
  - No two edges out of a node, start with same letter
- It follows, every substring has a path

# Suffix tree

---

- A suffix is a subsequence that begins at any position in the sequence & extends to the end of the sequence.
- Concatenate the two genomes into one sequence separated by a dummy character.
- Build suffix tree in linear time.
- Label each leaf node to indicate which suffix it represents in which genome.
- Identify all MUMs in one scan
  - every unique matching sequence is represented by an internal node with exactly two child leaf nodes, one from each genome.
  - unique matches that are maximal can be identified by mismatches at their ends.
- Identify MUMs on both DNA strands.

# Suffix tree

---

- A tree with edges labelled by strings
  - Labels of child edges of a node begin with distinct letters
  - Each leaf **L** represents a sequence—the labels on the path to **L** from the root
- Holds *all* suffixes of a set of sequences
  - A *suffix* is a subsequence that extends to the end of its sequence
- The suffix tree for sequences *A* and *B* :
  - Contains less than  $2(|A| + |B|)$  nodes.
  - Can be constructed in  $O(|A| + |B|)$  time!
- Still need lots of RAM

# Suffix tree

---

- String : any sequence of characters.
- Substring of string  $S$  :  $S[i..j]$   
string composed of characters  $i$  through  $j$ ,  
 $i \leq j$  of  $S$ .
  - $S = \text{biology} \Rightarrow \text{bio}$  is a substring.
  - $\text{blg}$  is not a substring.
  - Empty string is a substring of  $S$ .

# Suffix tree

---

- A suffix tree  $T$  of an  $m$ -character string  $S$  is a rooted tree with exactly  $m$  leaves numbered 1 to  $m$ . Each internal node other than the root has at least two children and each edge is labeled with a non-empty substring of  $S$ . No two edges out of a node can have edge-labels beginning with the same character.
- The key feature of the suffix tree is that for any leaf  $i$ , the concatenation of the edge-labels on the path from the root to leaf  $i$  exactly spells out the suffix of  $S$  that starts at position  $i$ . That is, it spells out  $S[i..m]$ .

# Suffix tree

---

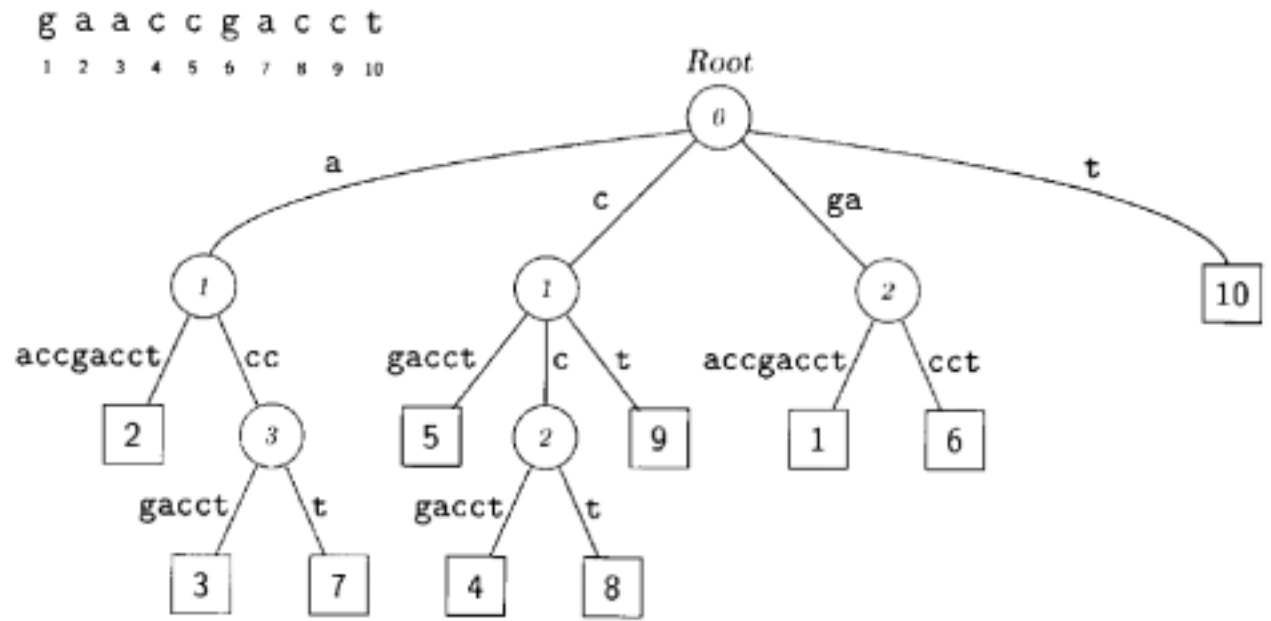
If S is ATCACATCATCA, its 12 suffixes are listed as below:

ATCACATCATCA	$S_{(1)}$
TCACATCATCA	$S_{(2)}$
CACATCATCA	$S_{(3)}$
ACATCATCA	$S_{(4)}$
CATCATCA	$S_{(5)}$
ATCATCA	$S_{(6)}$
TCATCA	$S_{(7)}$
CATCA	$S_{(8)}$
ATCA	$S_{(9)}$
TCA	$S_{(10)}$
CA	$S_{(11)}$
A	$S_{(12)}$



# Suffix tree

1 g a a c c g a c c t  
 2 a a c c g a c c t  
 3 a c c g a c c t  
 4 c c g a c c t  
 5 c g a c c t  
 6 g a c c t  
 7 a c c t  
 8 c c t  
 9 c t  
 10 t



- Square nodes are leaves & represent complete suffixes. They are labeled by the starting position of the suffix.
- Circular nodes represent repeated sequences & are labeled by the length of that sequence.
- In this example, the longest repeated sequence is acc occurring at positions 3 & 7.

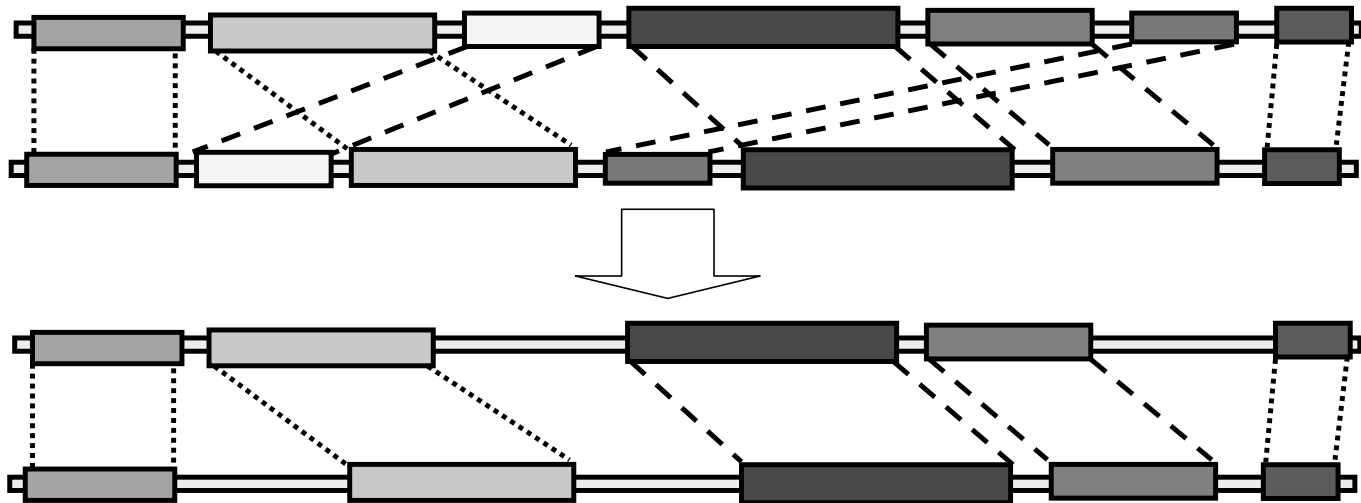
## Step 2: Sort the MUMs & find longest subsequence

---

- Set length of the shortest MUM.
  - e.g., 50 for highly similar genomes, 20 for similar ones.
- Sort the MUMs found in the MUM alignment according to their position in genome A.
- Extract the longest possible set of matches that occur in the same order in both genomes [i.e., solve variation of Longest Increasing Subsequence (LIS) problem to find sequences in ascending order in both genomes].
  - computer an ordered MUM alignment that provides an easy & natural way to scan the alignment from left to right
  - lengths of sequences represented by MUMs
  - overlaps
- Requires  $O(k \log k)$  time, where  $k$  = number of MUMs.

# Matching MUMs

---



# Longest Increasing Subsequence (LIS)

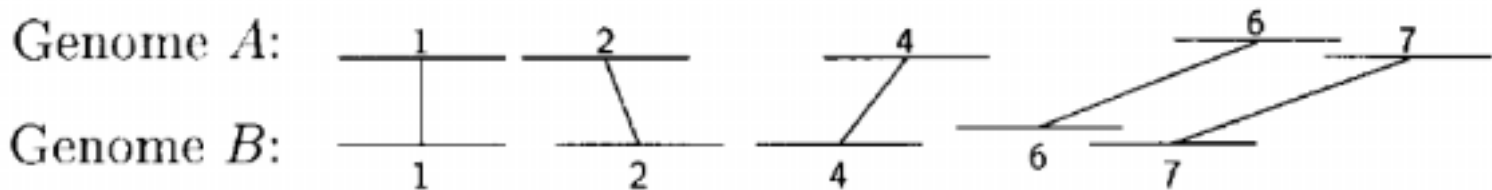
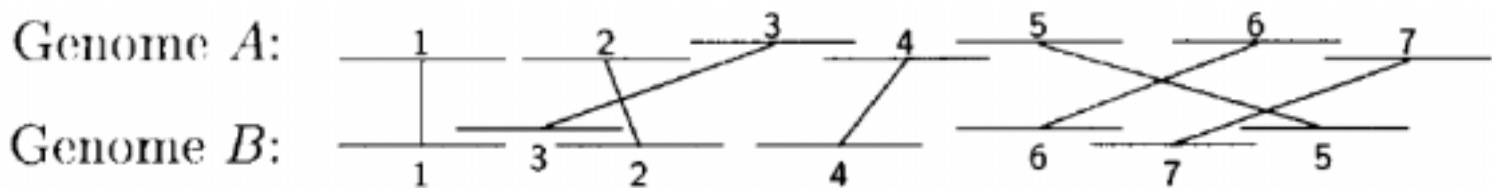
---

- Sequence {**1**, **2**, 10, 4, 5, 8, 6, 7, 9, 3}  
LIS is {1, 2, 4, 5, 6, 7, 9}

# Longest Increasing Subsequence (LIS)

---

- Aligning Genome A & Genome B after locating the MUMs.

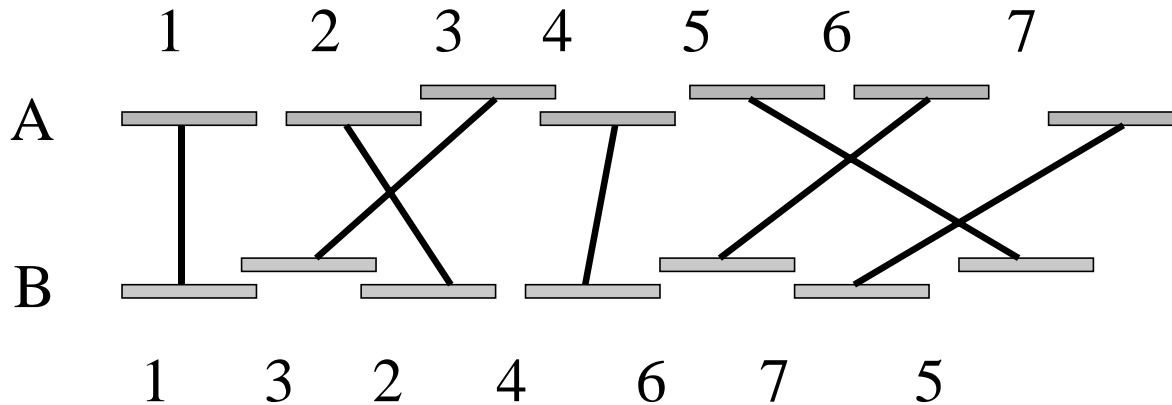


- Each MUM is here indicated only by a number, regardless of its length.

# Longest Increasing Subsequence (LIS)

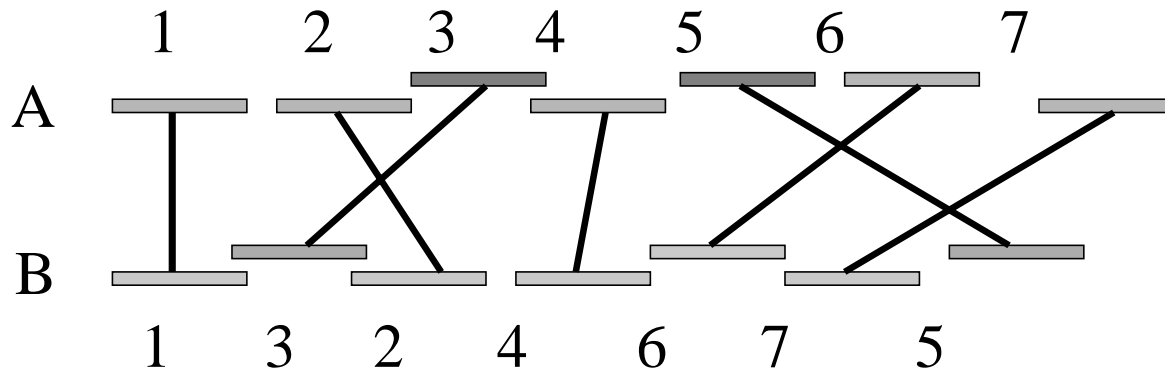
---

- Alignment to show all the MUMs.



# Longest Increasing Subsequence (LIS)

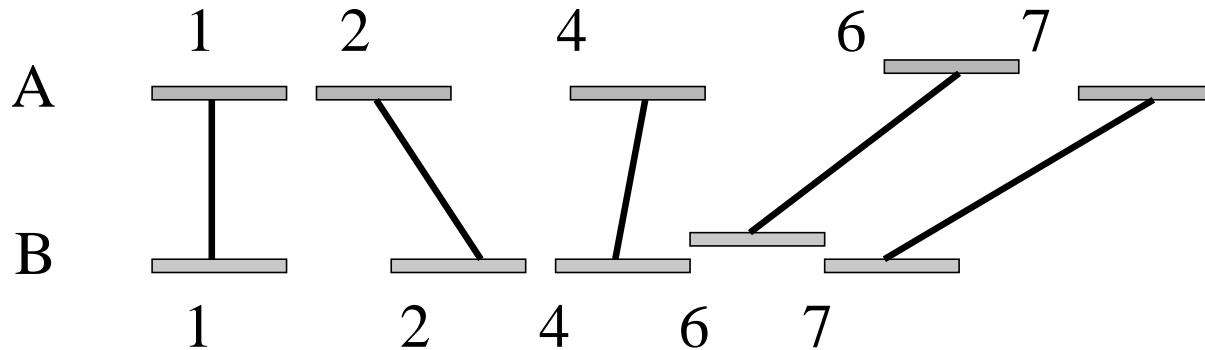
---



- The shift of MUM 5 in Genome B indicates a transposition.
- The shift of MUM 3 could be simply a random match or part of an inexact repeat sequence.

# Longest Increasing Subsequence (LIS)

---



- The alignment shows just the LIS of MUMs in Genome B.

# Analyze the gaps between adjacent MUMs

- Small gaps can be aligned with Smith-Waterman algorithm
- Large gaps can be aligned recursively
- Large inserts can be searched for separately.  
Many will be inconsistent MUMs
- Overlapping MUMs indicate variation in copy number of small repeats

# Types of gaps in a MUM alignment

---

1. SNP: exactly one base (indicated by ~) differs between the two sequences. It is surrounded by exact-match sequence.

Genome A: cgtcatggg<sup>c</sup>ggttcgtcgttg  
 Genome B: cgtcatggg<sup>c</sup>atttcgtcgttg

2. Insertion: a sequence that occurs in one genome but not the other.

Genome A: cggggtaaccgc.....cctggtcggg  
 Genome B: cggggtaaccgcg<sup>ttgctcggggtaaccgc</sup>cctggtcggg  
 ~~~~~

3. Highly polymorphic region: many mutations in a short region.

Genome A: ccg<sup>cctcgcctgg.gctggcgccc</sup>gctc  
 Genome B: ccg<sup>cctcgcctgg.gctggcgccc</sup>gctc  
 ~ ~ ~ ~ ~

4. Repeat sequence: the repeat is shown in uppercase. Note that the first copy of the repeat in Genome B is imperfect, containing one mismatch to the other three identical copies.

Genome A: cTGGGTGGGACAACGTaaaaaaaTGGGTGGGACAACGTc  
 Genome B: aTGGGTGGGGCgACGTgggggggggTGGGTGGGACAACGTa  
 ~ ~ ~ ~ ~

# Detecting the gaps in a MUM alignment

---

1. Repeats
  - Can be identified by MUMs overlapping each other
2. SNPs
  - Simple case: gap of one base between MUMs.
  - SNP adjacent to repeat sequences: use repeat processing
3. Insertions
  - Simple insertion: large gap in alignment in one genome but not the other
  - Transposition: appear in MUM alignment out of sequence.
4. Variable/Polymorphic regions:
  - Appear as gap in MUM alignment
  - If short, use Smith-Waterman dynamic programming
  - If long, run MUM detection with reduced minimum length.

## Step 3: Close the gaps

---

- Close the gaps in the alignment by performing local identification of:
  - SNPs:
    - between MUMs: trivial to detect
    - handle like repeats
  - Inserts:
    - simple insertions: trivial to detect
    - transpositions (subsequences that were deleted from one location & inserted elsewhere): look for out-of-sequence MUMs
  - Polymorphic regions:
    - short ones: align them with dynamic programming method
    - long ones: call MUMmer recursively with reduced minimum MUM length
  - repeats:
    - detected by overlapping MUMs

# Summary on MUMmer

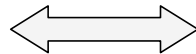
---

- Faster:  $O(n)$ .
  - very fast for alignment of genomes of different strains of the same species or genomes of similar species.
  - can handle long insertions & deletions
  - can detect reverses, SNPs, & repeats
- Large memory:  $O(n)$ .
- Require perfect match to construct alignment.
- Only useful to align closely related species.
  - not suitable for more divergent genomes such as human & mouse genomes --- it can find too few MUMs
  - speed suffer significantly for less similar sequences
  - minimum MUM length needs to be set lower
  - Many more runs of Smith-Waterman in Step 3
  - relies on other programs to mask repetitive sequences

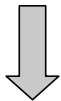
# New approach

---

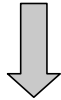
exact matching



find similar sequences



LIS

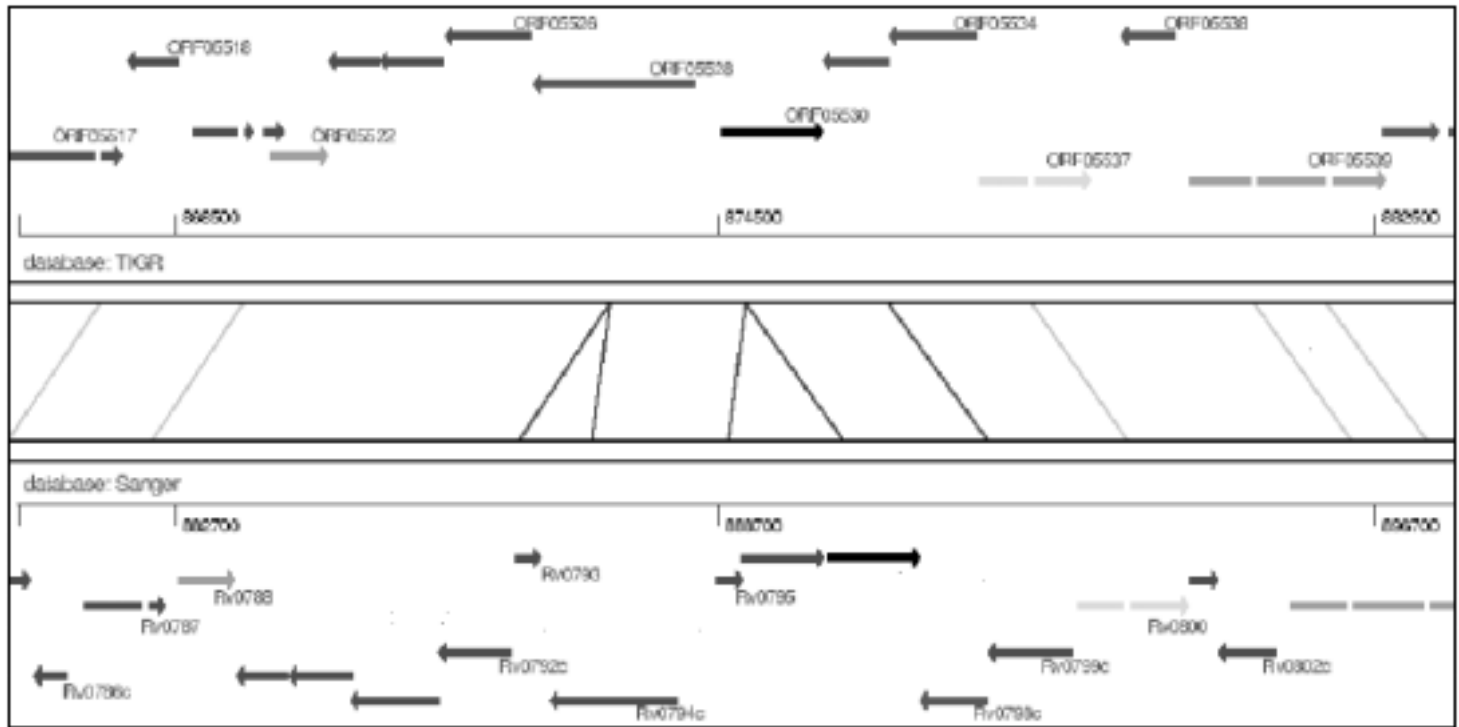


Closing gaps

- Slower?
- Small memory?
- Perfect match is not required.
- Align any related species.

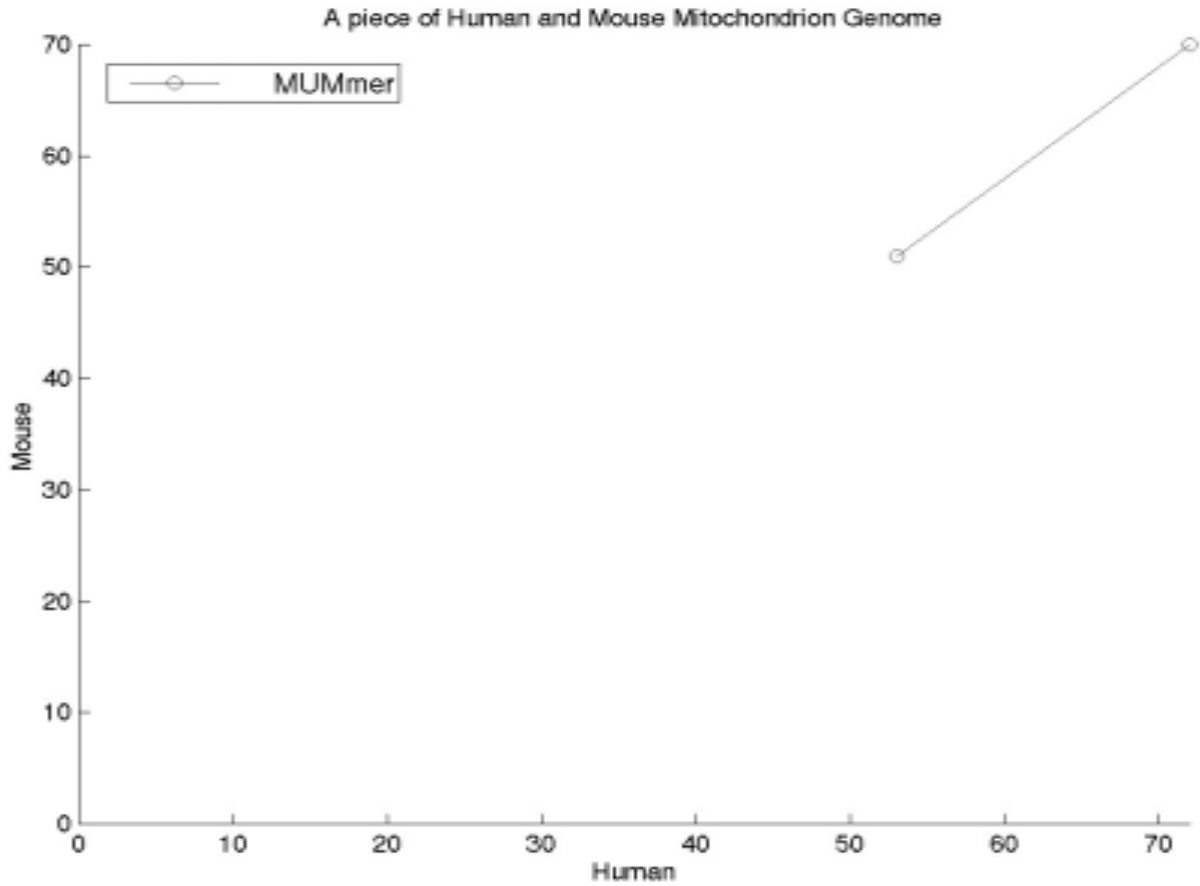
# Comparing two strains of *Mycobacterium tuberculosis*

CDC1551



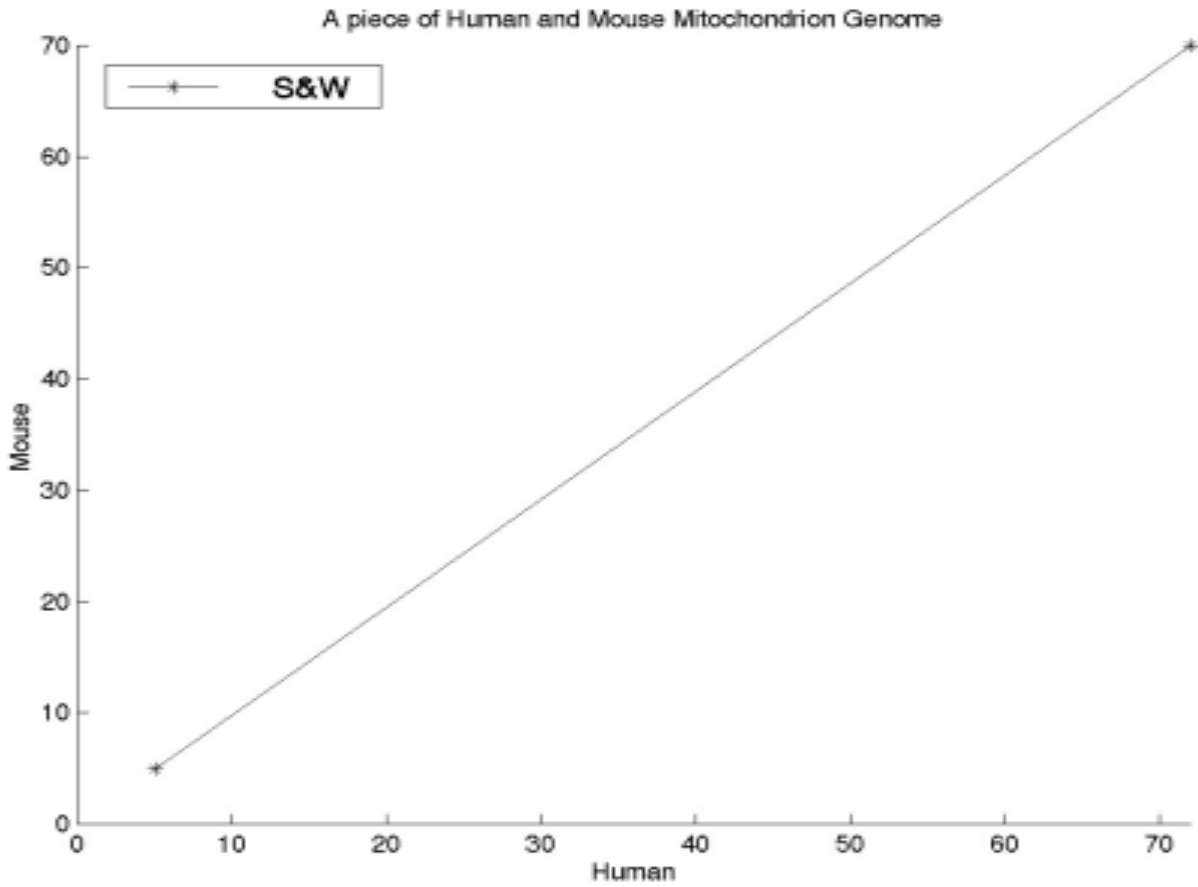
H37Rv

- Single green lines in the center connect single-base differences between the genomes. Blur v-shaped lines indicate insertions. The genes from both genomes are displayed as arrows, with color-coding to indicate the role assigned to each gene. White lines (gaps) appearing in the middle of some arrows indicate silent mutations in those genes.



```

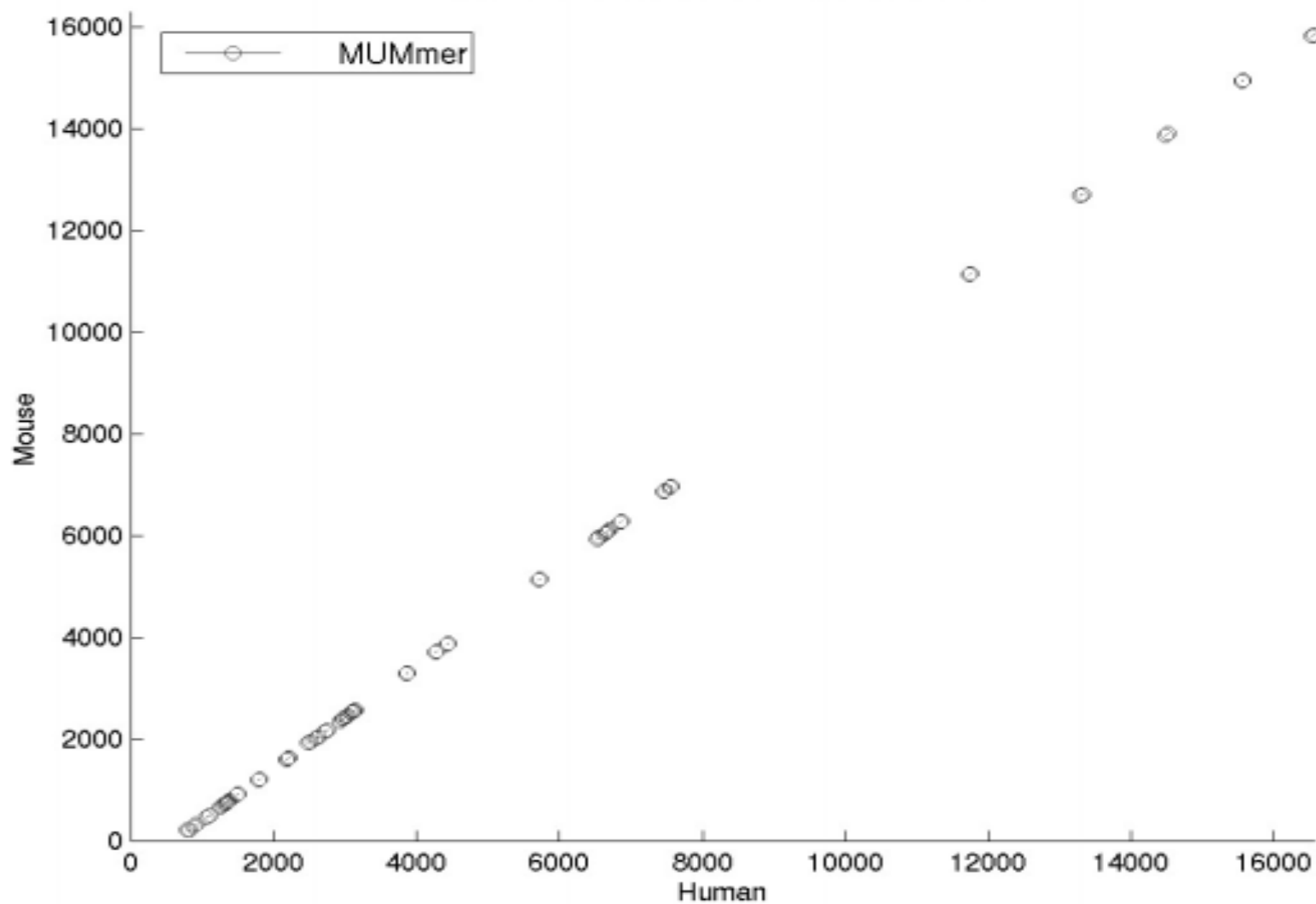
GGACTTAAACCCACAAACACTTAGTTAACAGCTAAGCACCTAATCAACTGGCTTCAATCTACTTCT
||| ||||| || || | ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
GGAATTAAACCTACGAAAATTTAGTTAACAGCTAAATACCCTATT--ACTGGCTTCAATCTACTTCT
  
```



```

GGACTTAAACCCACAAACACTTAGTTAACAGCTAAGCACCTAATCAACTGGCTTCAATCTACTTCT
||| ||||| || || | ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
GGAATTAAACCTACGAAAATTTAGTTAACAGCTAAATACCCTATT--ACTGGCTTCAATCTACTTCT
  
```

Human and Mouse Mitochondrion Genome



# Alignment of mouse and Human ATP synthase F<sub>0</sub>

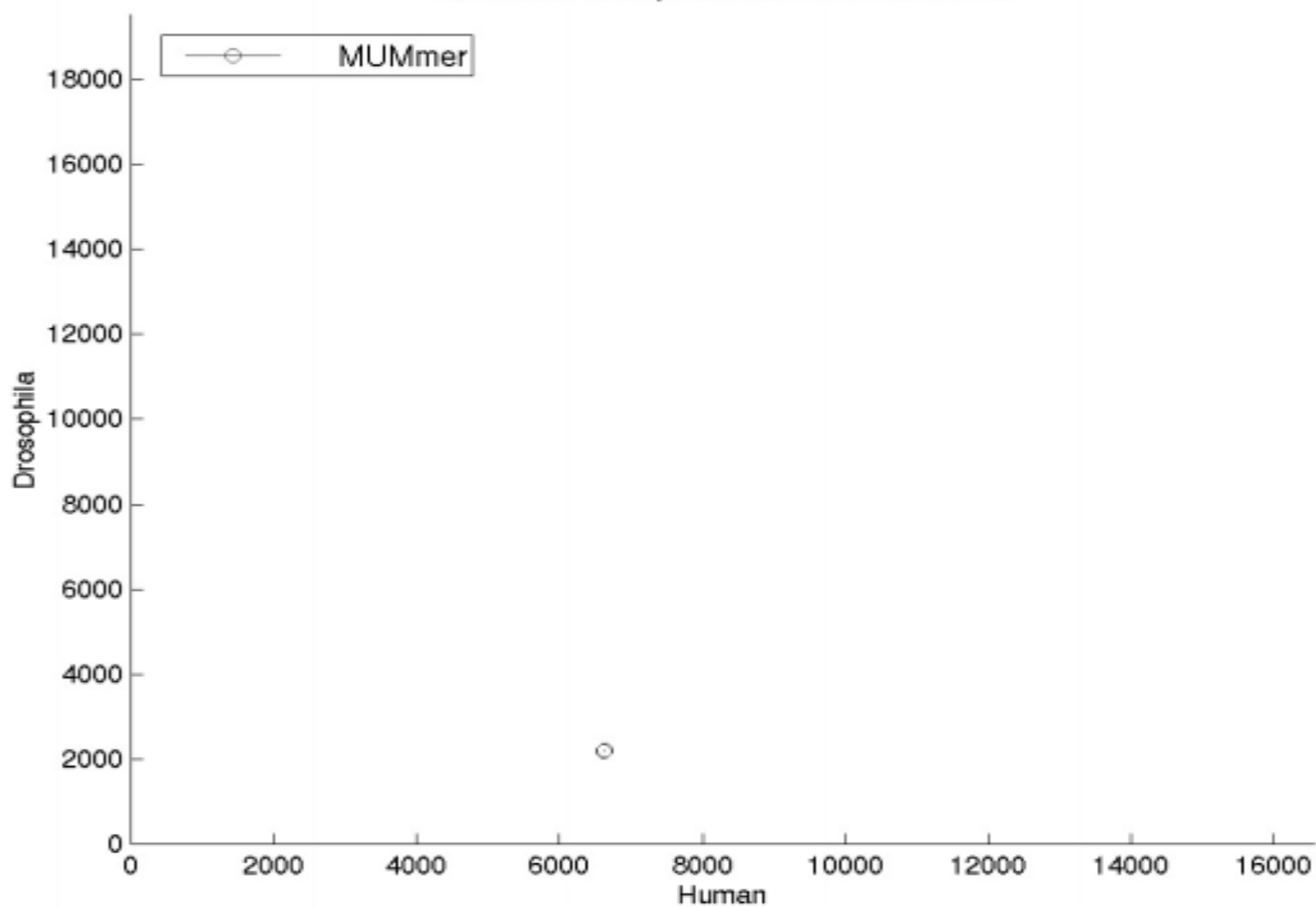
---

Query: human mitochondrial genome  
Subject: mouse mitochondrial genome

```
8716 aaaggacgaacctgatctcttatactagtagtacccttaatcattttttattgccacaactaac 8775
64   K G R T W S L M L V S L I I F I A T T N
    ||||| ||||| ||| | || ||| | || ||| ||||| ||||| ||||| ||
8116 aaaggacgaacatgaaccctaataattgtttccctaatacatatttattggatcaacaaat 8175
64   K G R T W T L M I V S L I M F I G S T N

8776 ctcctcggactcctgcctcactcatttacaccaaccaccaactatctataaacctagcc 8835
84   L L G L L P H S F T P T T Q L S M N L A
    ||||| || ||| | || ||| ||||| ||||| ||||| ||||| ||||| |||
8176 ctcctaggccttttaccacatacatatttacacactactaccaactatccataaatctaagt 8235
84   L L G L L P H T F T P T T Q L S M N L S
```

Human and Drosophila Mitochondrion Genome



# Alignment of Human and Drosophila Cytochrome B

---

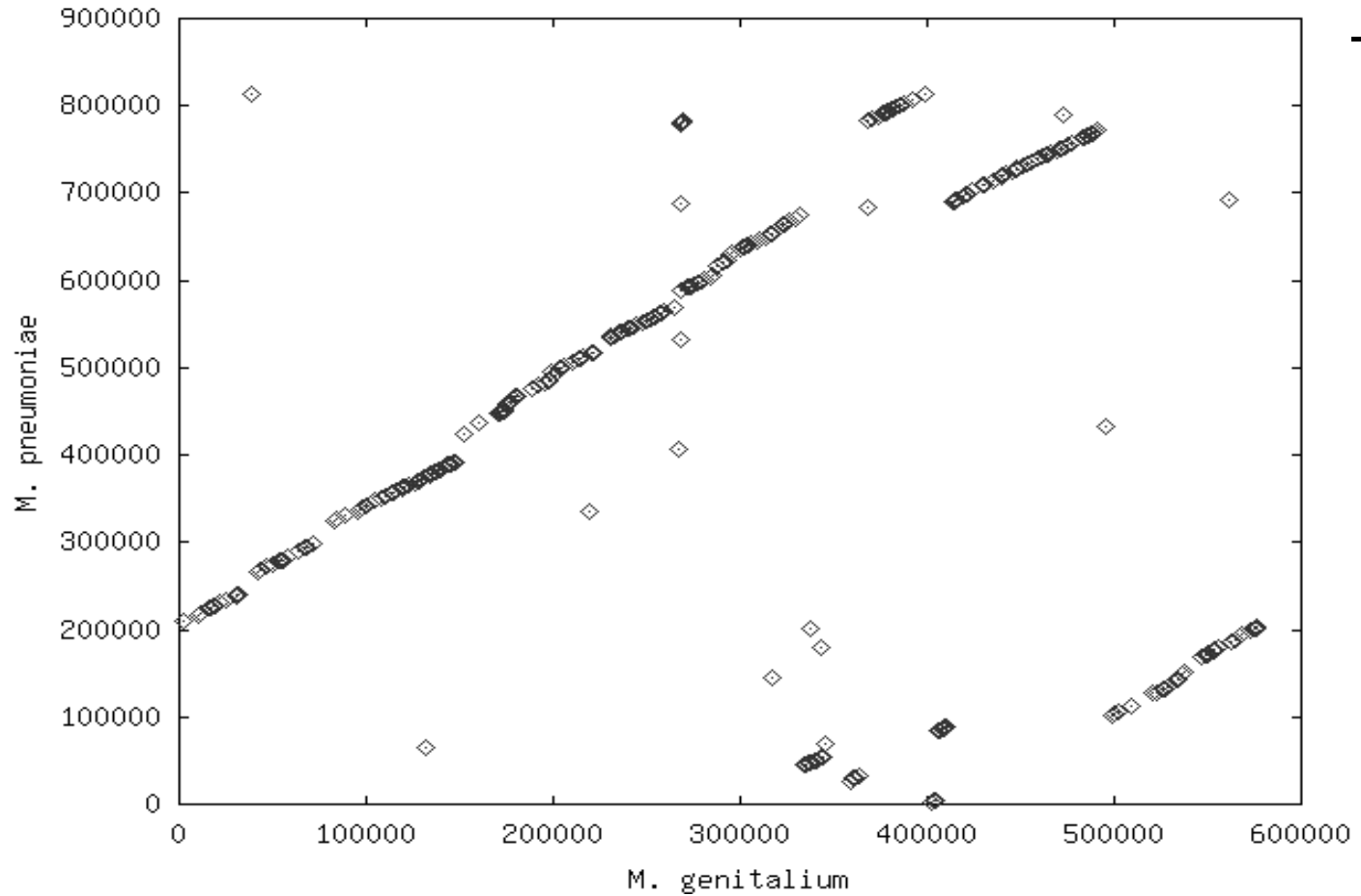
Query: human mitochondrial genome

Subject: drosophila mitochondrial genome

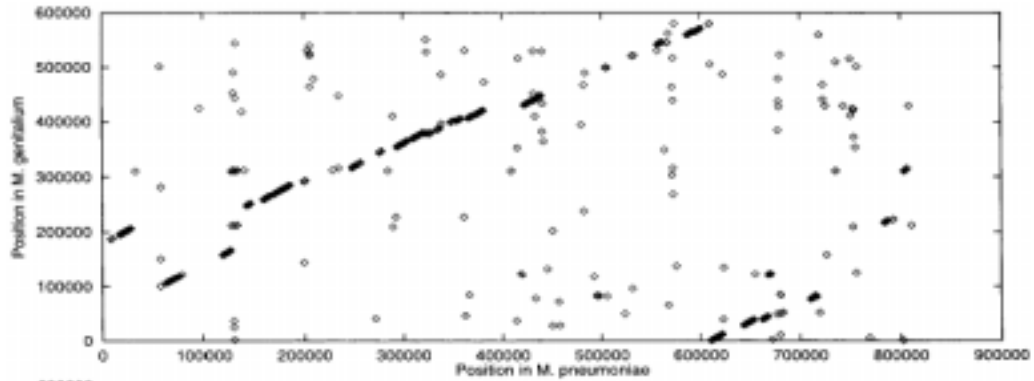
```
15116 atagcaacagccttcataggctatgtcctcccgtgaggccaaatatcattctgaggggcc 15175
124   M A T A F M G Y V L P W G Q M S F W G A
      |||| | ||||| || ||||| || || | || ||||| ||||| ||||| ||||| ||
10870 ataggaacagcttttataggatacgtattaccttgaggacaaatatcattttgagtagct 10929
125   M G T A F M G Y V L P W G Q M S F W V A

15176 acagtaattacaaacttactatccgccatcccatacattgggacagacctagttcaatga 15235
144   T V I T N L L S A I P Y I G T D L V Q W
      || || ||||| || ||| ||| ||| ||||| ||| | || | ||| ||||| |||||
10930 actgttattactaatttattatacgctatcccttacttaggtatagatttagttcaatga 10989
145   T V I T N L L Y A I P Y L G M D L V Q W
```

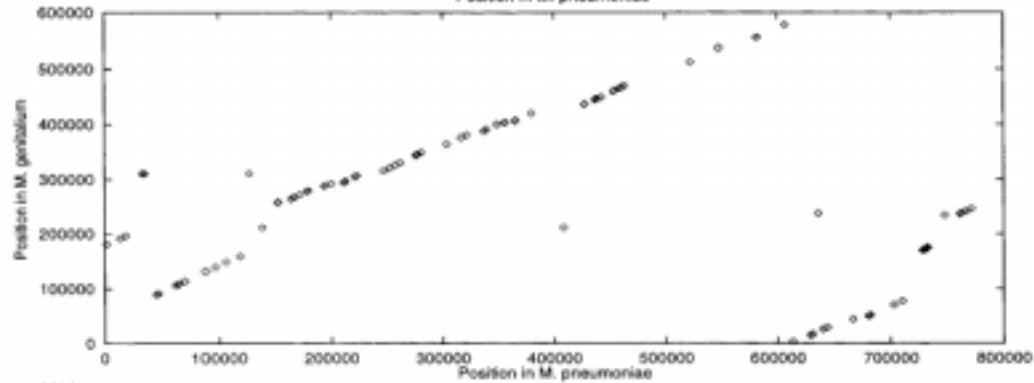
# *Mycoplasma genitalium* vs. *Mycoplasma pneumoniae*



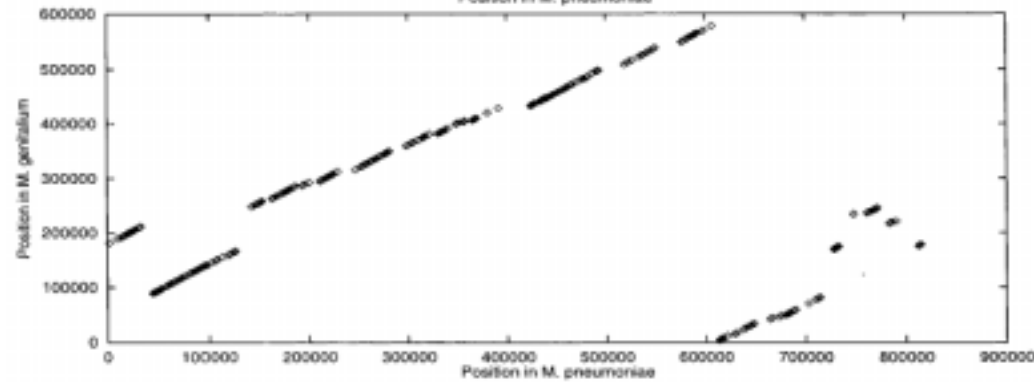
# Comparing *M. genitalium* and *M. pneumoniae*



FASTA

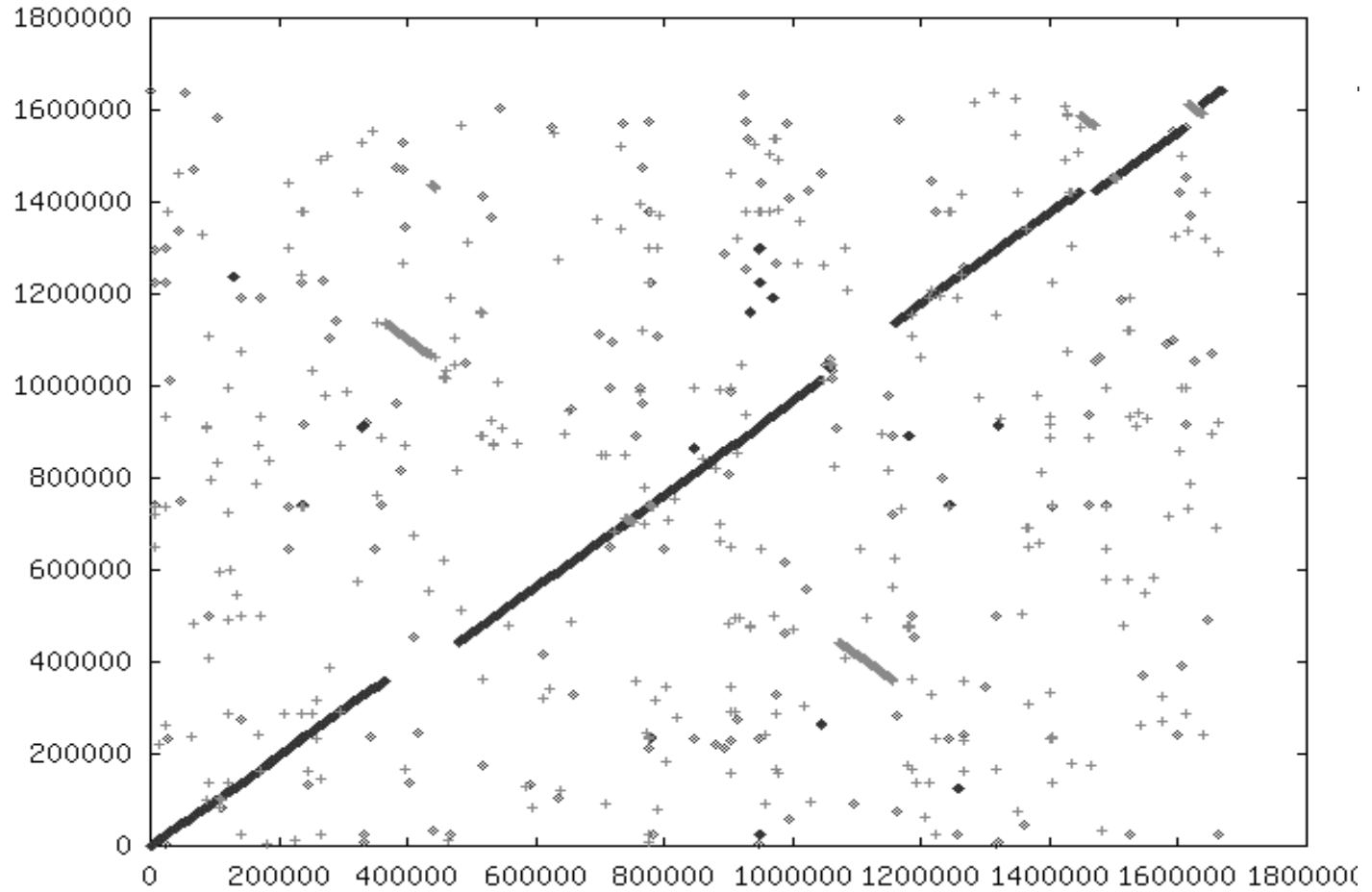


Matching  
25-mers

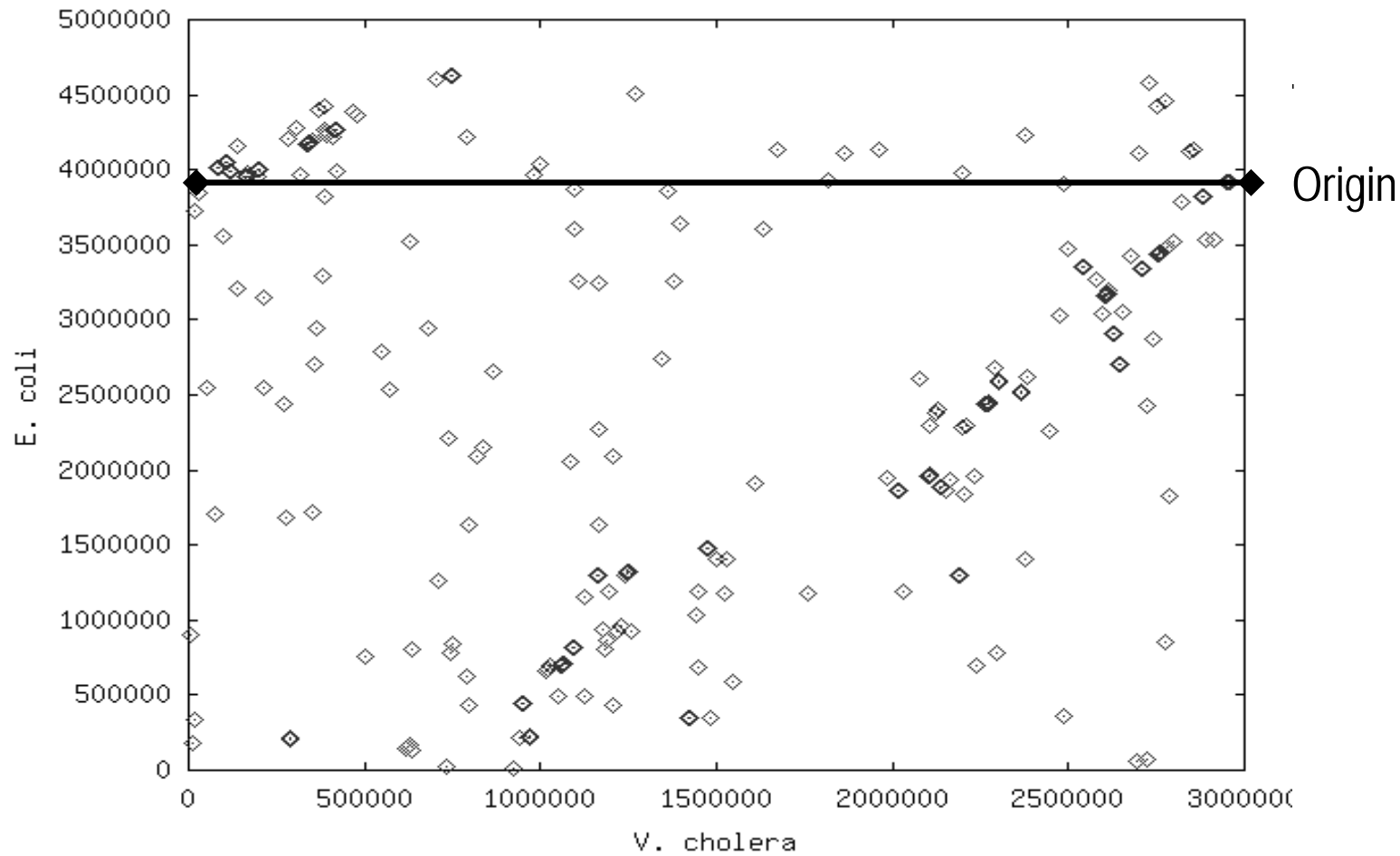


MUMs

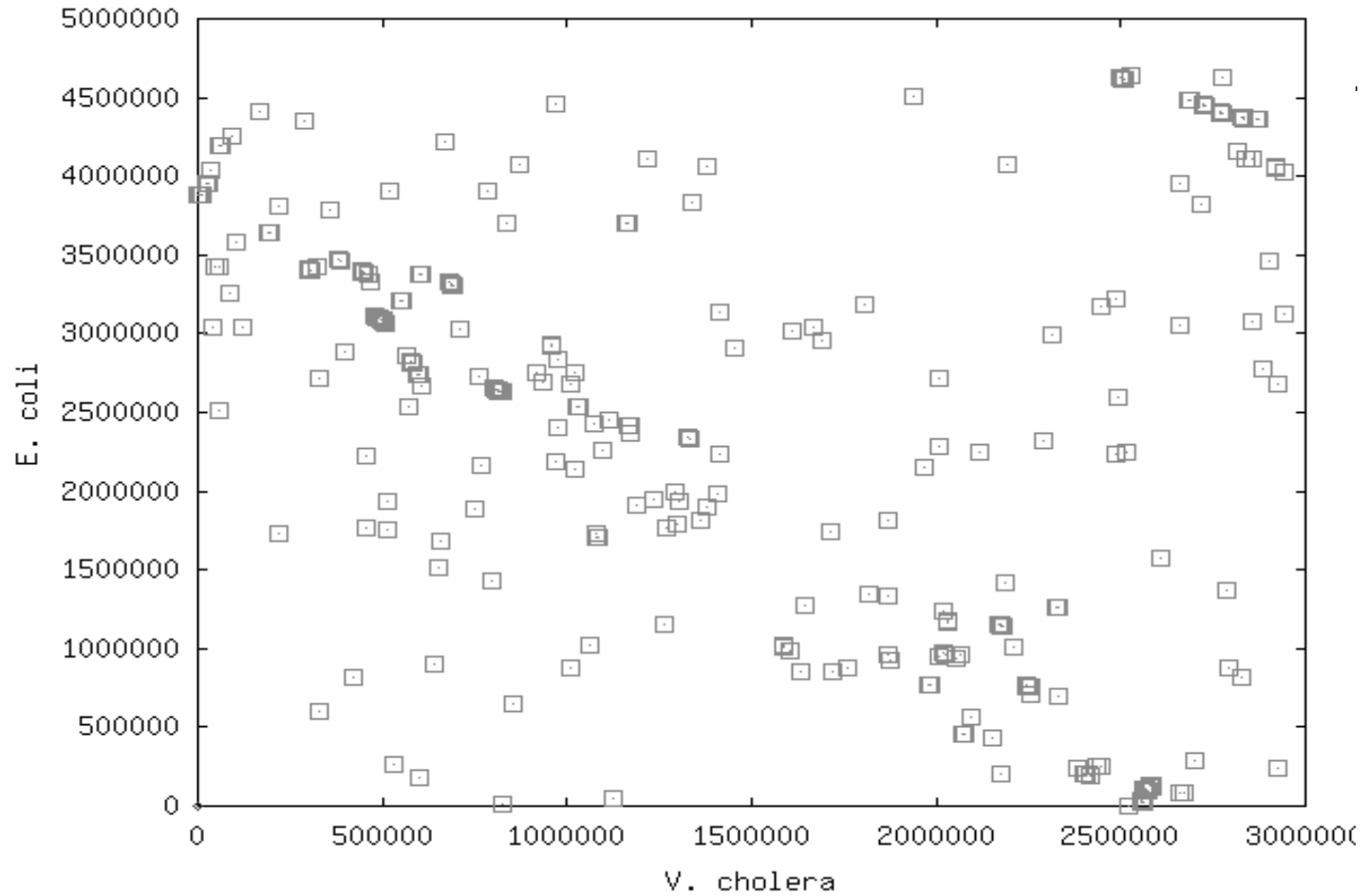
# *Helicobacter pylori* 26695 vs. J99



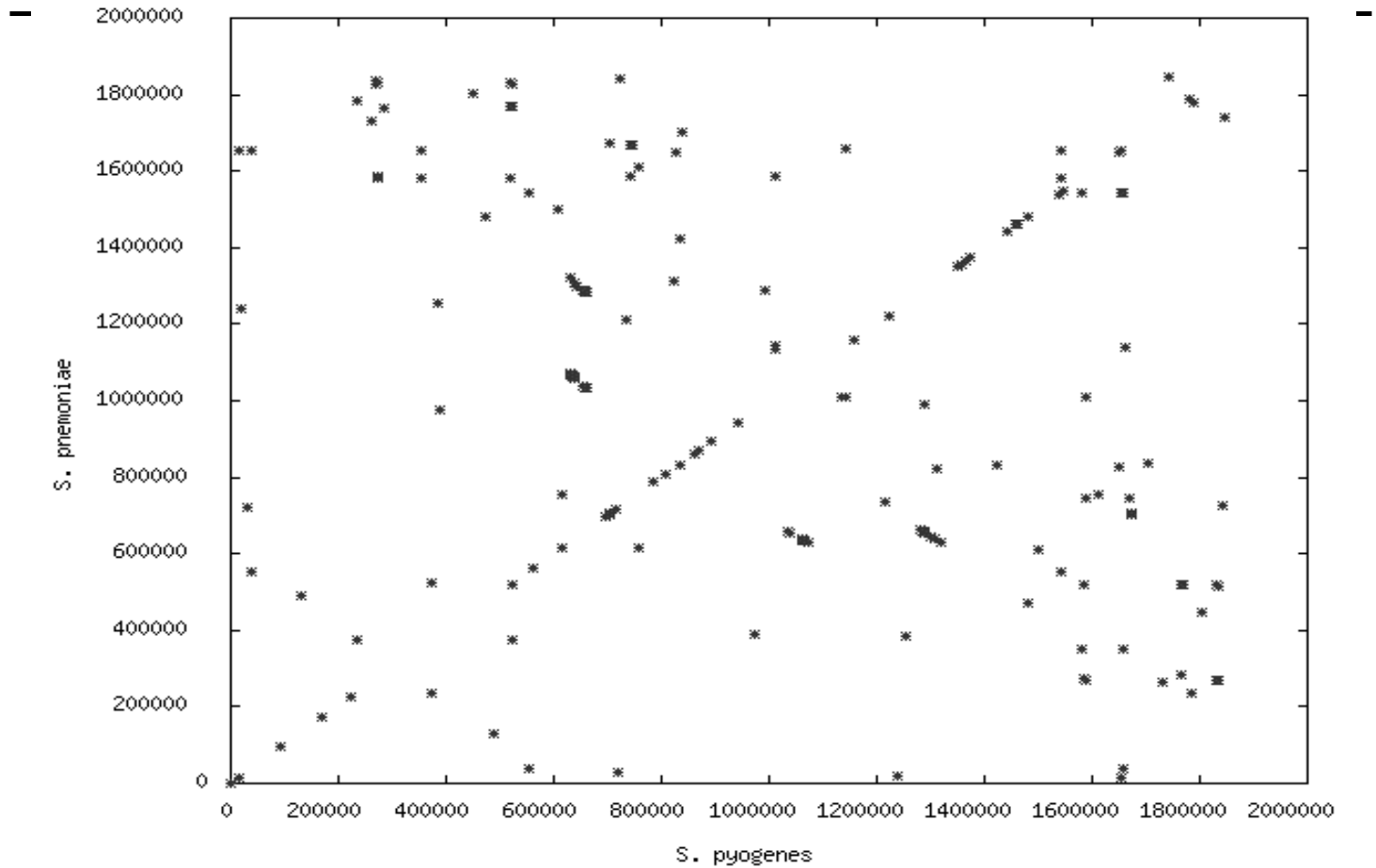
# *Vibrio cholera* (forward) vs. *E. coli*



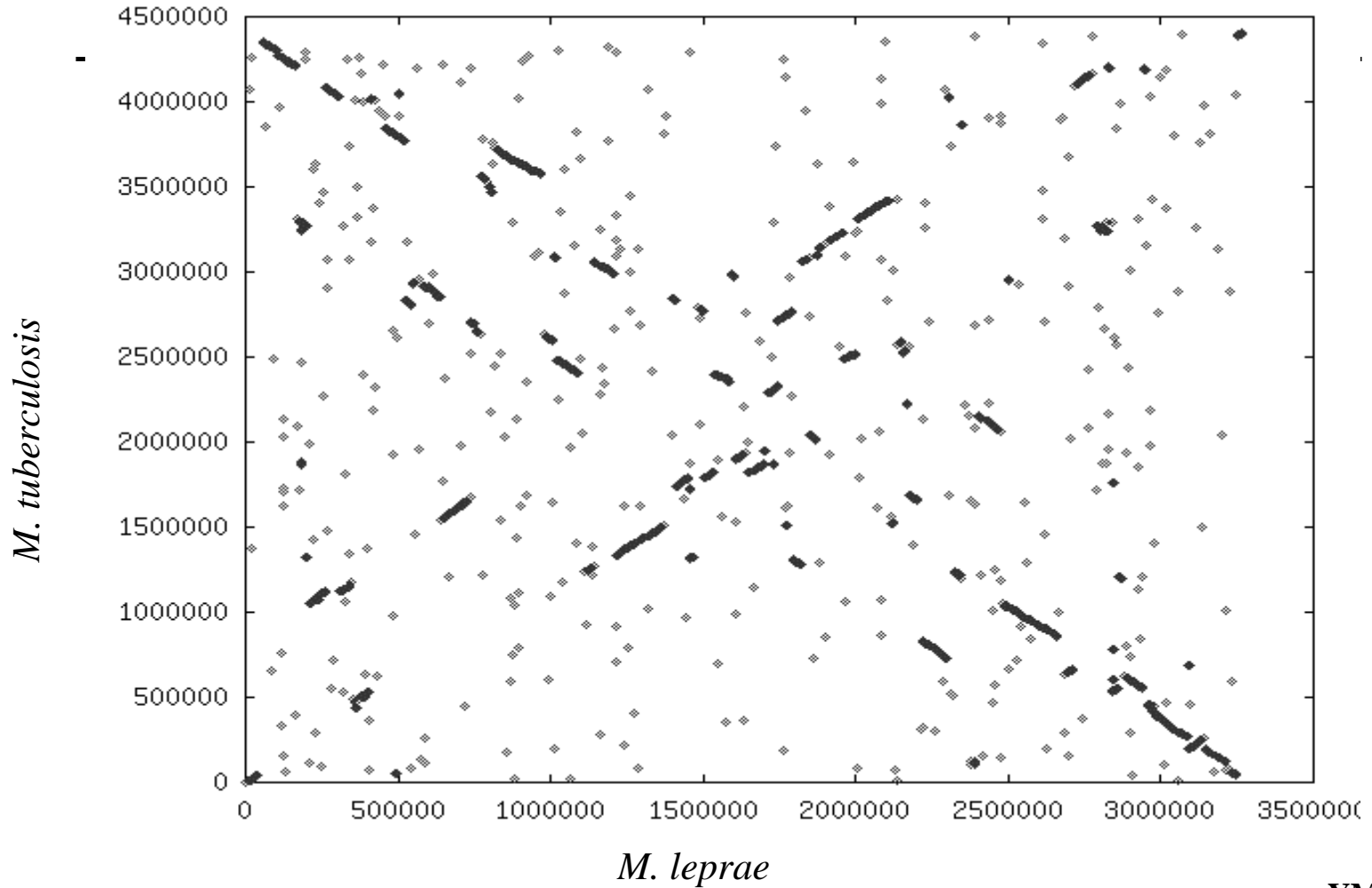
# *Vibrio cholera* (reverse) vs. *E. coli*



# *Streptococcus pyogenes* vs. *Streptococcus pneumoniae*



# *Mycobacterium leprae* vs. *Mycobacterium tuberculosis*

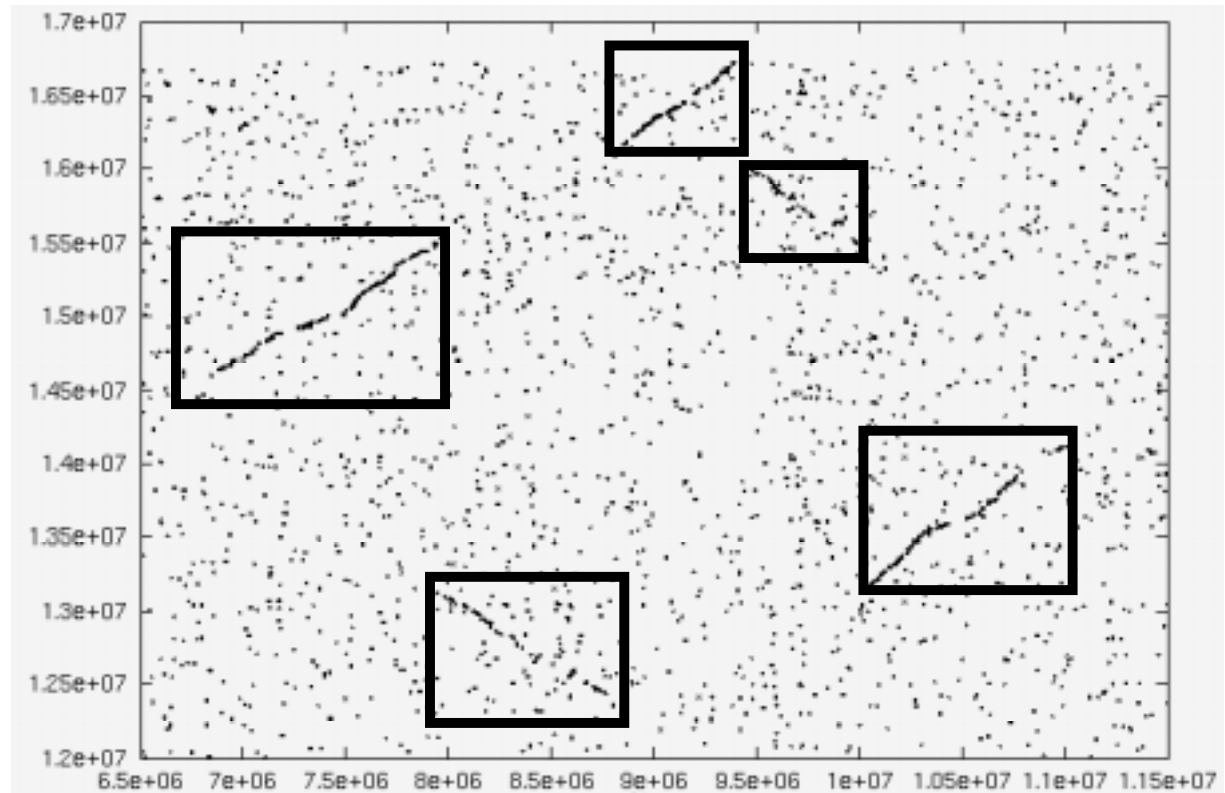


# Chr 2 vs. Chr 4 of *Arabidopsis thaliana*: discovery of a 4 Mb duplication

---

Chr 4

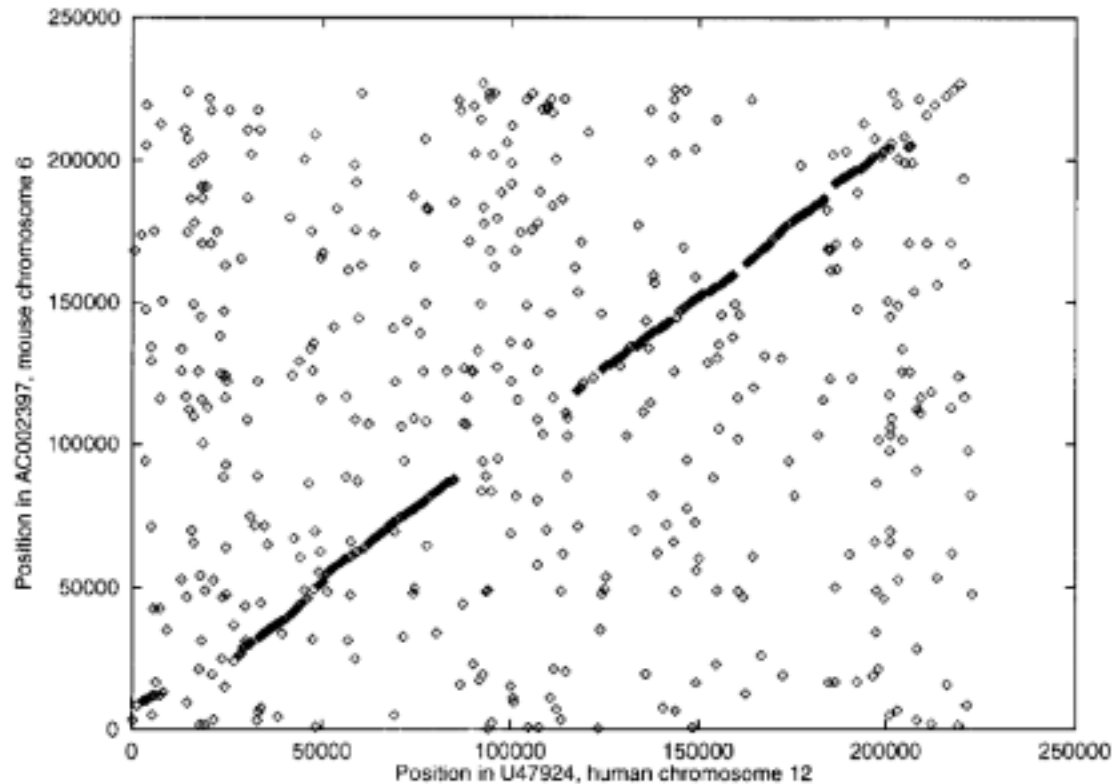
- 430 (39%) of the 1100 genes are duplicated.



Chr 2

# Human Chr 12p13 vs. Mouse Chr 6

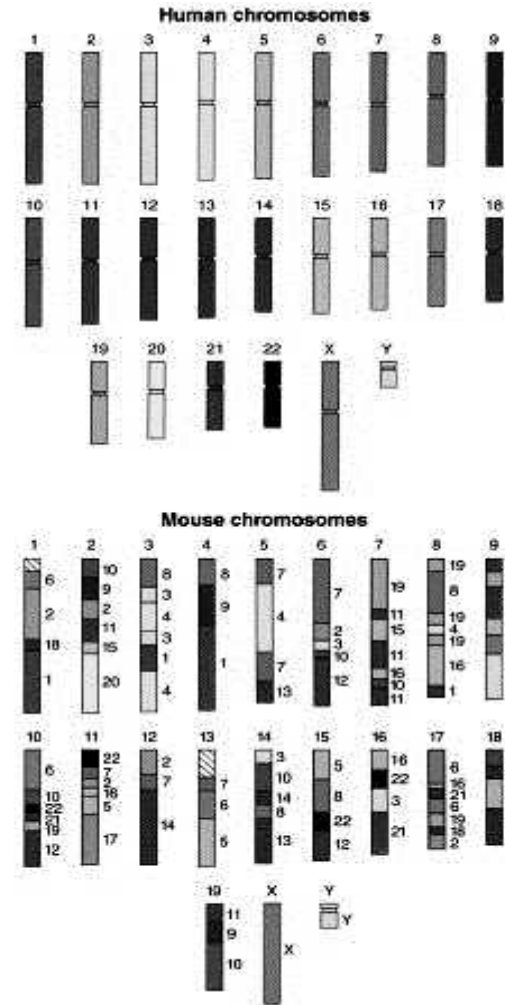
---



- Each point in the plot corresponds to a MUM of  $\geq 15$  bp.

# Gene rearrangements – mouse vs. human

- Approx. same no. of chromosomes & local gene order in mammals
- Chromosomes become broken & rejoined



# ASSIRC

---

- Accelerated Search for SImilarity Regions in Chromosome
- ASSIRC finds regions of similarity in pair-wise genomic sequence alignments.
- The method involves 3 steps:
  - (i) identification of short exact chains of fixed size, called 'seeds', common to both sequences, using hashing functions;
  - (ii) extension of these seeds into putative regions of similarity by a 'random walk' procedure (i.e. the four bases are associated);
  - (iii) final selection of regions of similarity by assessing alignments of the putative sequences.
- Uses simulations to estimate the proportion of regions of similarity not detected for particular region sizes, base identity proportions and seed sizes.
- This approach can be tailored to the user's specifications.
- They looked for regions of similarity between two yeast chromosomes (V and IX). The efficiency of the approach was compared to those of conventional programs BLAST and FASTA, by assessing CPU time required and the regions of similarity found for the same data set.
- <http://www.biologie.ens.fr/perso/vincens/assirc.html>
- <ftp://ftp.biologie.ens.fr/pub/molbio/assirc.tar.gz>

# BLAT

(<http://genome.ucsc.edu/goldenPath/help/hgTracksHelp.html>)

(BLAT--the BLAST-like alignment tool. Genome Res. 2002 Apr;12(4):656-64.)

---

- Only DNA sequences of 25,000 or less bases and protein or translated sequence of 5000 or less letters will be processed. If multiple sequences are submitted at the same time, the total limit is 50,000 bases or 12,500 letters.
- BLAT on DNA is designed to quickly find sequences of 95% & greater similarity of length 40 bases or more. It may miss more divergent or shorter sequence alignments. It will find perfect sequence matches of 33 bases, & sometimes find them down to 22 bases. BLAT on proteins finds sequences of 80% & greater similarity of length 20 amino acids or more. In practice DNA BLAT works well on primates, & protein blat on land vertebrates
- BLAT is not BLAST. DNA BLAT works by keeping an index of the entire genome in memory. The index consists of all non-overlapping 11-mers except for those heavily involved in repeats. The index takes up a bit less than a gigabyte of RAM. The genome itself is not kept in memory, allowing BLAT to deliver high performance on a reasonably priced Linux box. The index is used to find areas of probable homology, which are then loaded into memory for a detailed alignment. Protein BLAT works in a similar manner, except with 4-mers rather than 11-mers. The protein index takes a little more than 2 gigabytes
- Some common uses of BLAT include: finding the genomic coordinates of mRNA or protein within a given assembly, determining the exon structure of a gene, displaying a coding region within a full-length gene, isolating an EST of special interest as its own track, searching for gene family members, & finding human homologs of a query from another species.

# MegaBLAST

---

- MegaBLAST uses the greedy algorithm for nucleotide sequence alignment search. This program is optimized for aligning sequences that differ slightly as a result of sequencing or other similar "errors". When larger word size is used (see explanation below), it is up to 10 times faster than more common sequence similarity programs. Mega BLAST is also able to efficiently handle much longer DNA sequences than the blastn program of traditional BLAST algorithm.
- <http://www.ncbi.nlm.nih.gov/blast/megablast.html>

# SSAHA

---

- Sequence Search and Alignment by Hashing Algorithm
- Software tool for very fast matching and alignment of DNA sequences.
- Achieves fast search speed by converting sequence information into a hash table data structure which can then be searched very rapidly for matches
- <http://www.sanger.ac.uk/Software/analysis/SSAHA/>
- SSAHA is available as a precompiled executable for Compaq Alpha & linux
- Run from the Unix command line
- Need > 1GB RAM (needs a lot of memory)
- SSAHA algorithm best for application requiring exact or “almost exact” matches between two sequences – e.g. SNP detection, fast sequence assembly, ordering and orientation of contigs

# WABA

---

- The wobble-aware bulk aligner (WABA)
- WJ Kent, AM Zahler, 2000, Genome Research, 10(8): 1071-1073
- Three passes
  - Identify homologous regions
  - Align in detail overlapping 2000x5000 base regions
  - Join the overlapping alignments
- Aligned 8 million bases of *Caenorhaditis briggsae* against the entire 97 million bases of *Caenorhaditis elegans* genome.
  - Overall similarity: 59% sequence identity.
- Run time on a Pentium III 450 mHz,
  - First pass: 20 hrs.
  - Second pass: 11 days.
  - Third pass: 15 min.

# Problems with Visualizing Genomes

---

- Alignment programs output often were visualized by text file, which can be intuitively difficult to interpret when comparing genomes.
- Visualization tools needed to handle the complexity and volume of data and present the information in a comprehensive and comprehensible manner to a biologist for interpretation.
- Genome Alignment Visualization tools need to provide:
  - interpretable alignments,
  - gene prediction and database homologies from different sources
  - Interactive features: real time capabilities, zooming, searching specific regions of homologies
  - Represent breaks in synteny
  - Multiple alignments display
  - Displaying contigs of unfinished genomes with finished genomes
  - Handle various data formats

# Genome Comparison Visualization Tool

---

- ACT - Artemis Comparison Tool (displays parsed BLAST alignments; based on Artemis – an annotation tool)
  - <http://www.sanger.ac.uk/Software/ACT/>
- Alfresco (displays DBA alignments)
  - <http://www.sanger.ac.uk/Software/Alfresco/> (Jareborg & Durbin 2000)
- PipMaker (displays BlastZ alignments)
  - <http://bio.cse.psu.edu/pipmaker/> (Schwartz et al. 2000)
- Enteric/Menteric/Maj (displays Blastz alignments)
  - <http://glovin.cse.psu.edu/enterix/> (Florea et al. 2000; McClelland et al. 2000)
- Intronerator (displays WABA alignments)
  - <http://www.cse.ucsc.edu/~kent/intronerator/> (Kent & Zahler 2000b)
- VISTA (Visualization Tool for Alignment) (displays GLASS alignments)
  - <http://www-gsd.lbl.gov/vista/>
- SynPlot (displays DIALIGN and GLASS alignments)
  - <http://www.sanger.ac.uk/Users/igrg/SynPlot/>

# Artemis Comparison Tool (ACT)

---

- ACT is a DNA sequence comparison viewer based on Artemis
- Can read complete EMBL and GenBank entries or sequence in Fasta or raw format
- Comparisons usually done by blastn or tblastx
- ACT is free software and is distributed under the GNU Public License
- Java based software
- Latest release 2.0 better support Eukaryotic Genome Comparison

<http://www.sanger.ac.uk/Software/ACT/>

<http://www.sanger.ac.uk/Software/ACT/>



The Wellcome Trust  
Sanger Institute



[Sanger Home](#) | [Acedb](#) | [YourGenome](#) | [Ensembl](#) | [Trace Server](#) | [Library](#)

[Info](#) | [Databases](#) | [Blast](#) | [Genomics](#) | [Infrastructure](#) | [HGP](#) | [CGP](#) | [Projects](#) | [Software](#) | [Teams](#) | [Search](#)

[Data Release Policy](#) | [Conditions of Use](#)



### Software Home

- Production s/w
- Mapping s/w
- Analysis s/w
- Format Specs
- Perldocs

### ACT

[Up a level](#)

[Overview](#)

[Introduction](#)

[Examples and Screenshots](#)

[License](#)

[Getting ACT](#)

[Latest Stable Release](#)

[User Manual](#)

## ACT: A DNA Sequence Comparison Viewer

### Introduction

ACT (Artemis Comparison Tool) is a DNA sequence comparison viewer based on [Artemis](#). In common with Artemis, ACT is written in [Java](#) and runs on UNIX, GNU/Linux, Macintosh and MS Windows systems. It can read complete [EMBL](#) and [GENBANK](#) entries or sequence in FASTA or raw format. Extra sequence features can be in EMBL, GENBANK or [GFF](#) format. ACT is distributed under the [same license as Artemis](#).

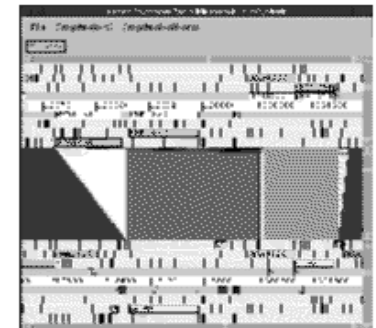
The sequence comparison displayed by ACT is usually the result of a blastn or tblastx search that has been processed by [MSPcrunch](#). MSPcrunch must be run with the -d flag for the output to be usable by ACT.

To see ACT in action go to [the examples page](#).

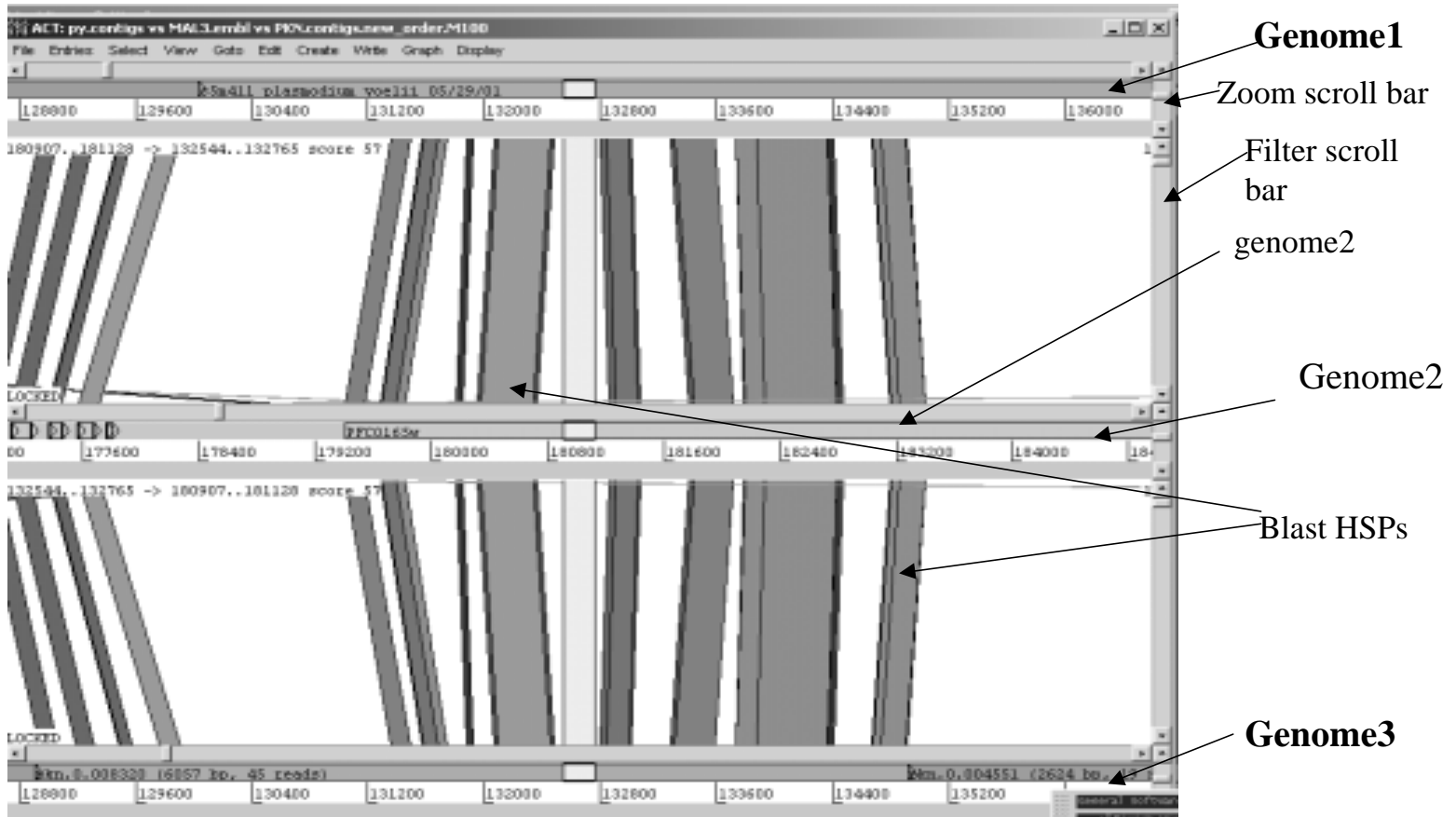
### Releases

The [first release](#) of ACT is now available along with the corresponding [user manual](#).

### Documentation

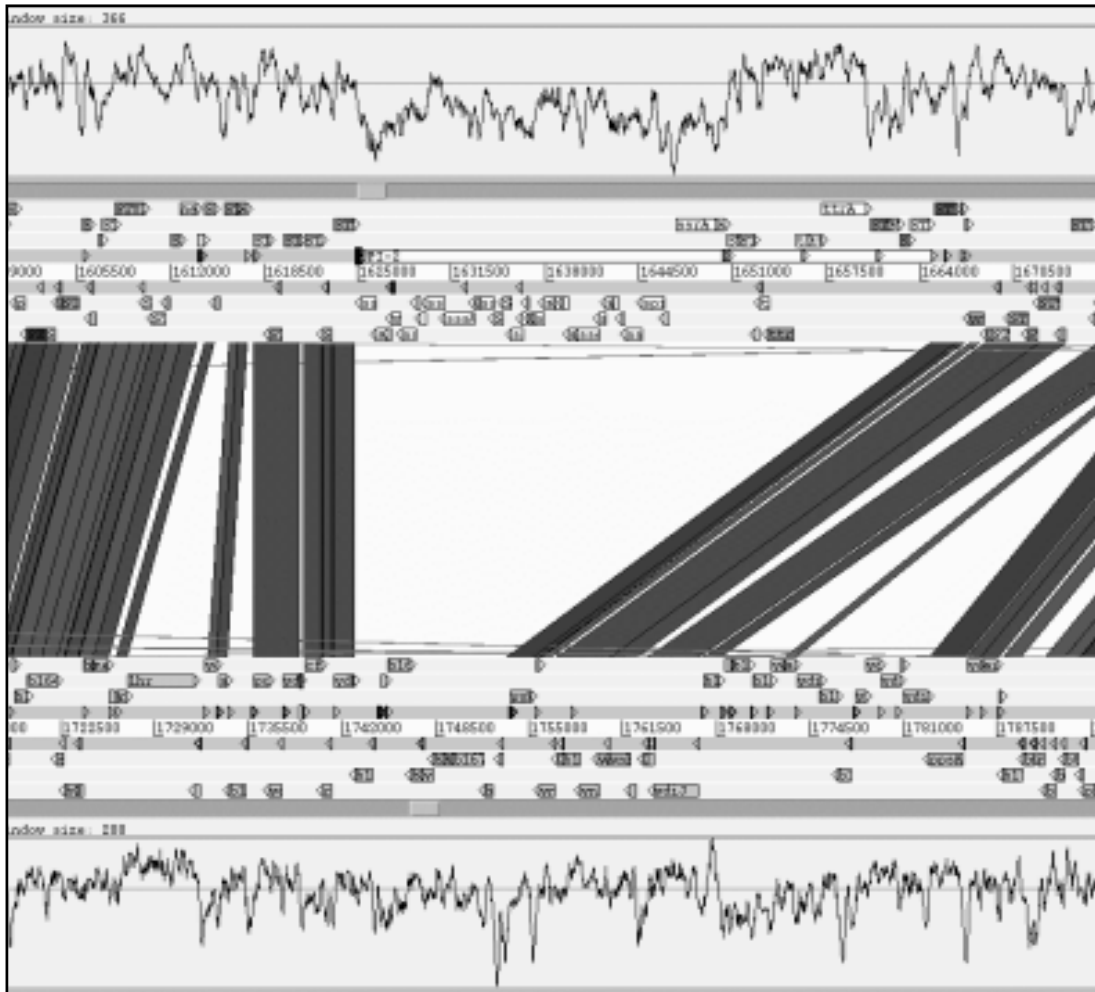


# The ACT Display



# Salmonella typhi vs. E. coli – SPI-2

*S.typhi*



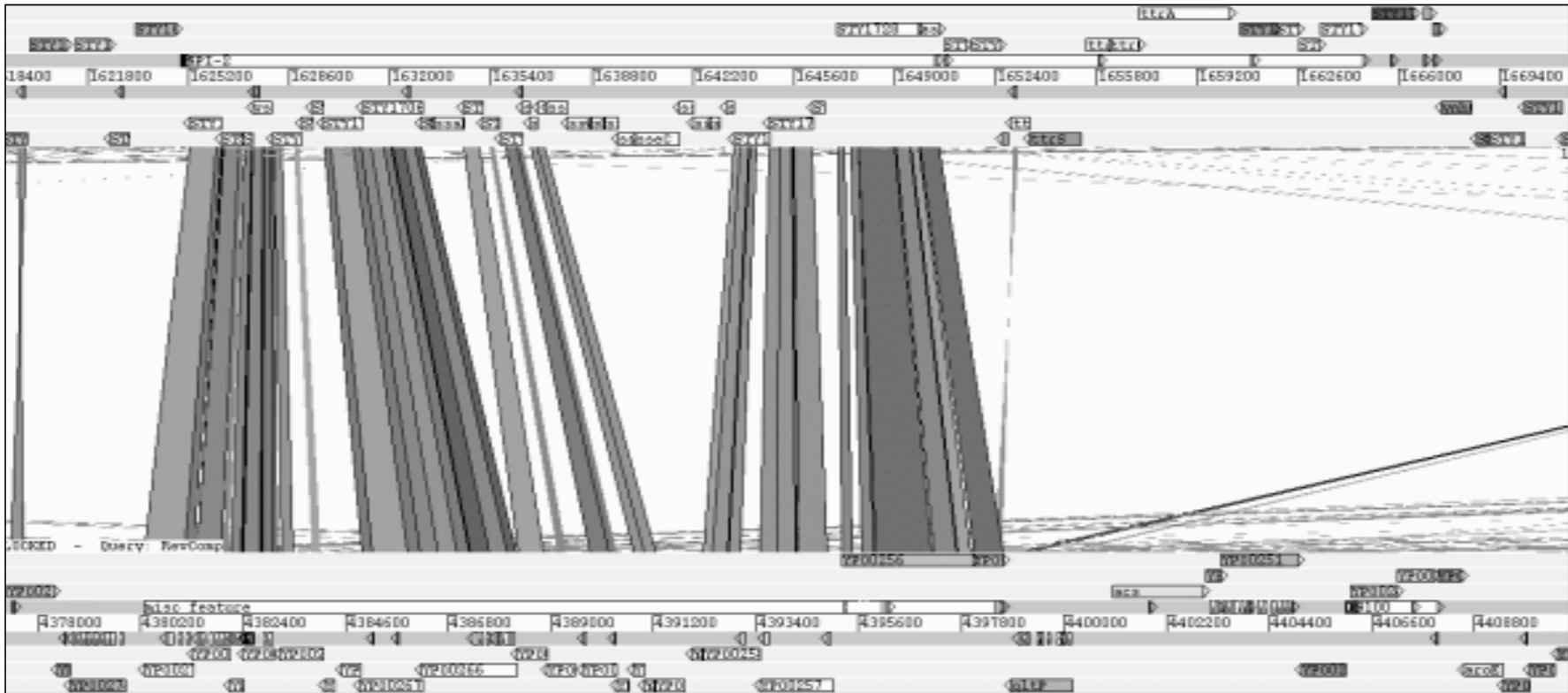
G+C

tRNA  
phage/IS genes  
Pseudogenes

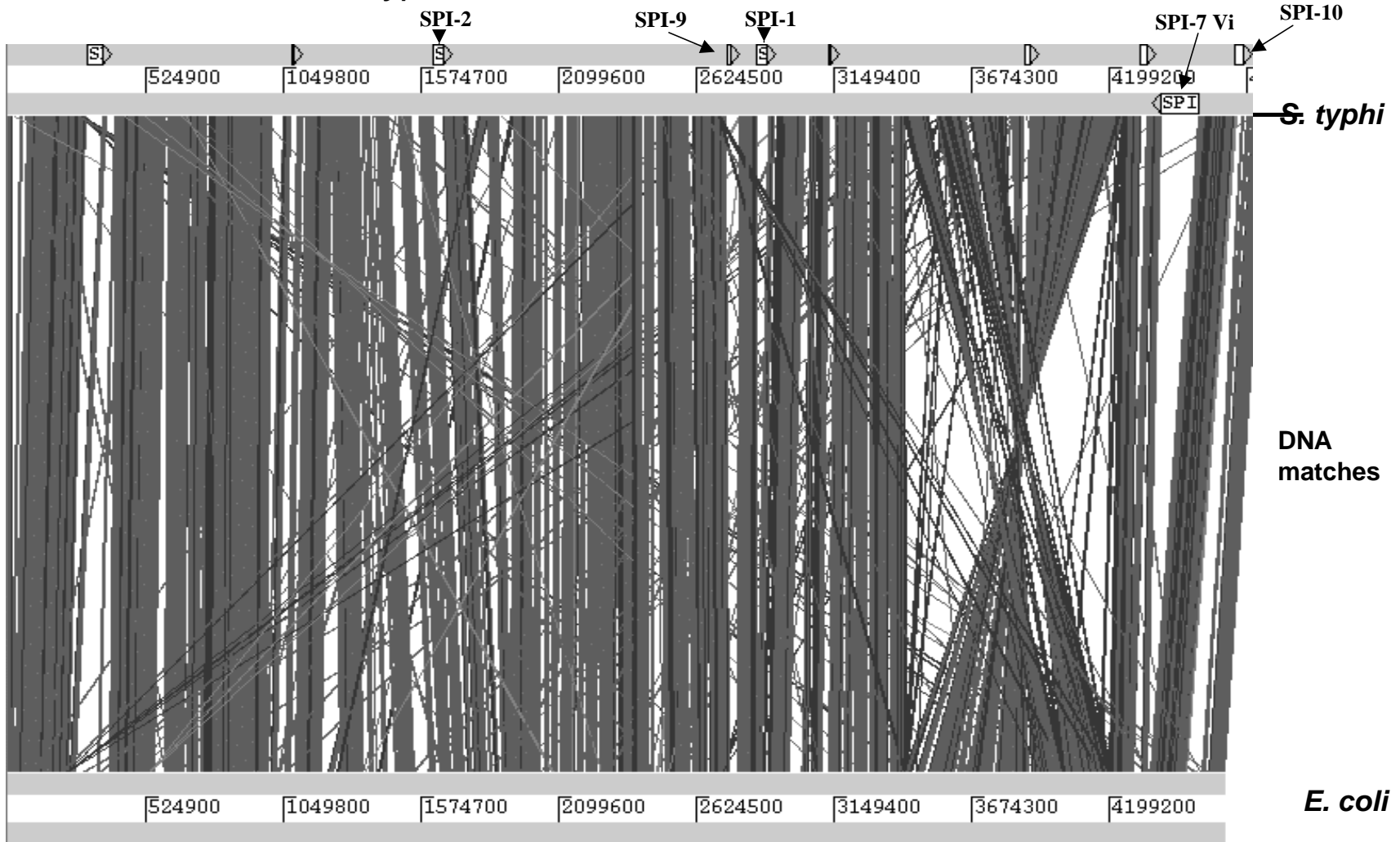
Blast hits

*E.coli*

# *Salmonella typhi* and *Yersinia pestis* type III secretion systems

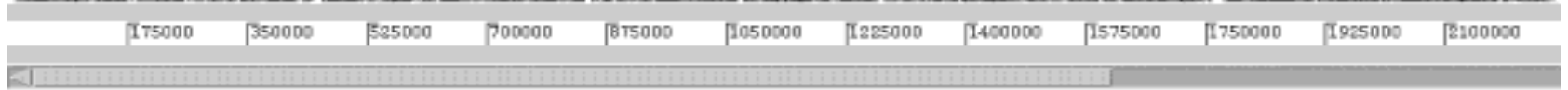
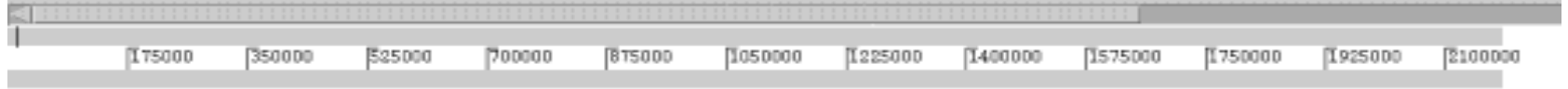
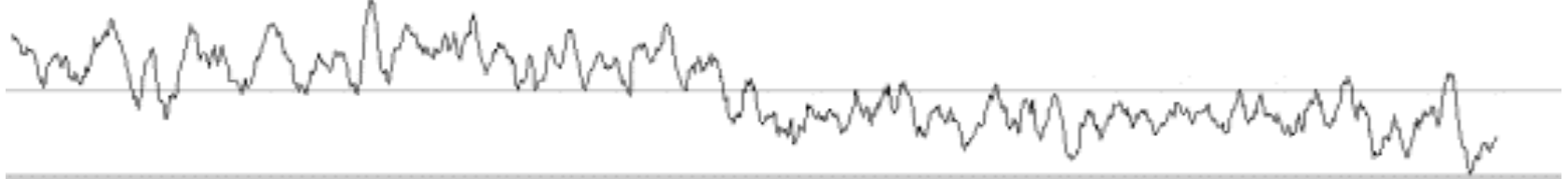


# Salmonella typhi vs. E. coli - ACT

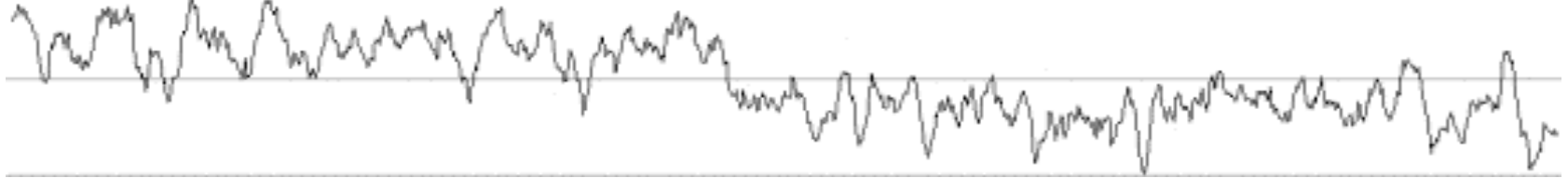


# *Neisseria meningitidis* - A vs. B comparison - ACT

GC Deviation  $(G-C)/(G+C)$  Window size: 20000



GC Deviation  $(G-C)/(G+C)$  Window size: 20000



# Proteome by proteome comparisons

---

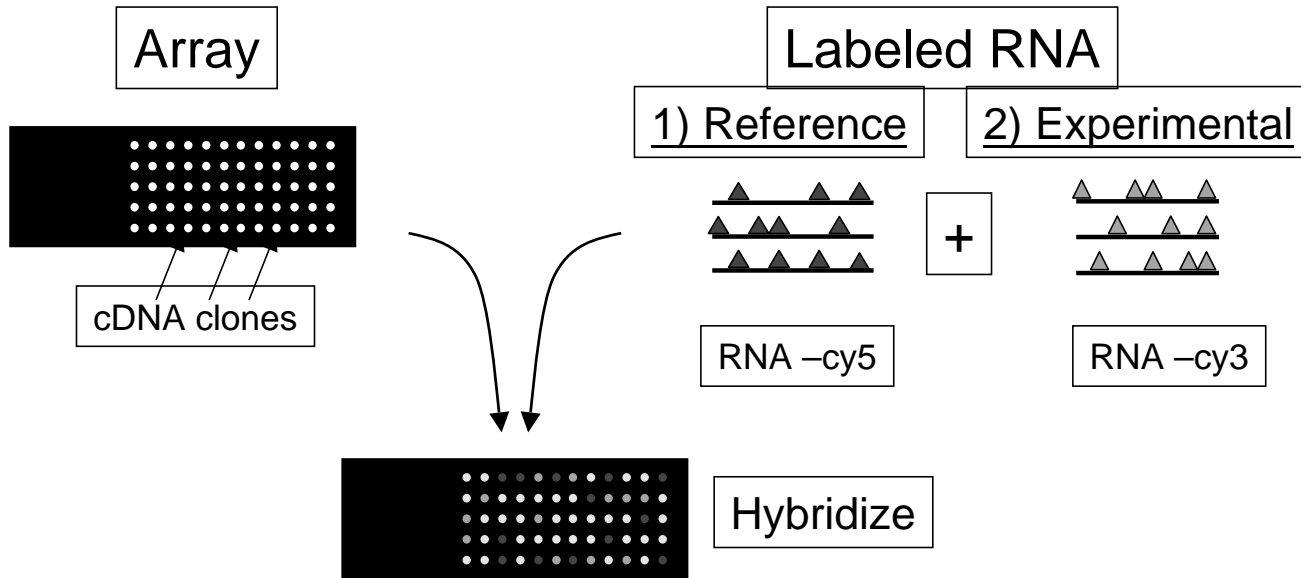
## Unique, orthologous & paralogous proteins

Expand proteome comparison to include other species –

possible results of cluster analysis:

- unique protein – no close relatives in another proteome
- paralogs in one proteome only – expansion of gene family
- orthologs – clearly same function in different organisms (core function) due to great similarity
- complex gene family due to gene duplication events in ancestor sequence

# Gene expression profiling



|   | Expression                  | Ratio |
|---|-----------------------------|-------|
| ● | <b>higher</b> in 1 than 2   | > 2   |
| ○ | <b>same</b> between 1 and 2 | 1     |
| ● | <b>lower</b> in 1 than 2    | < 0.5 |

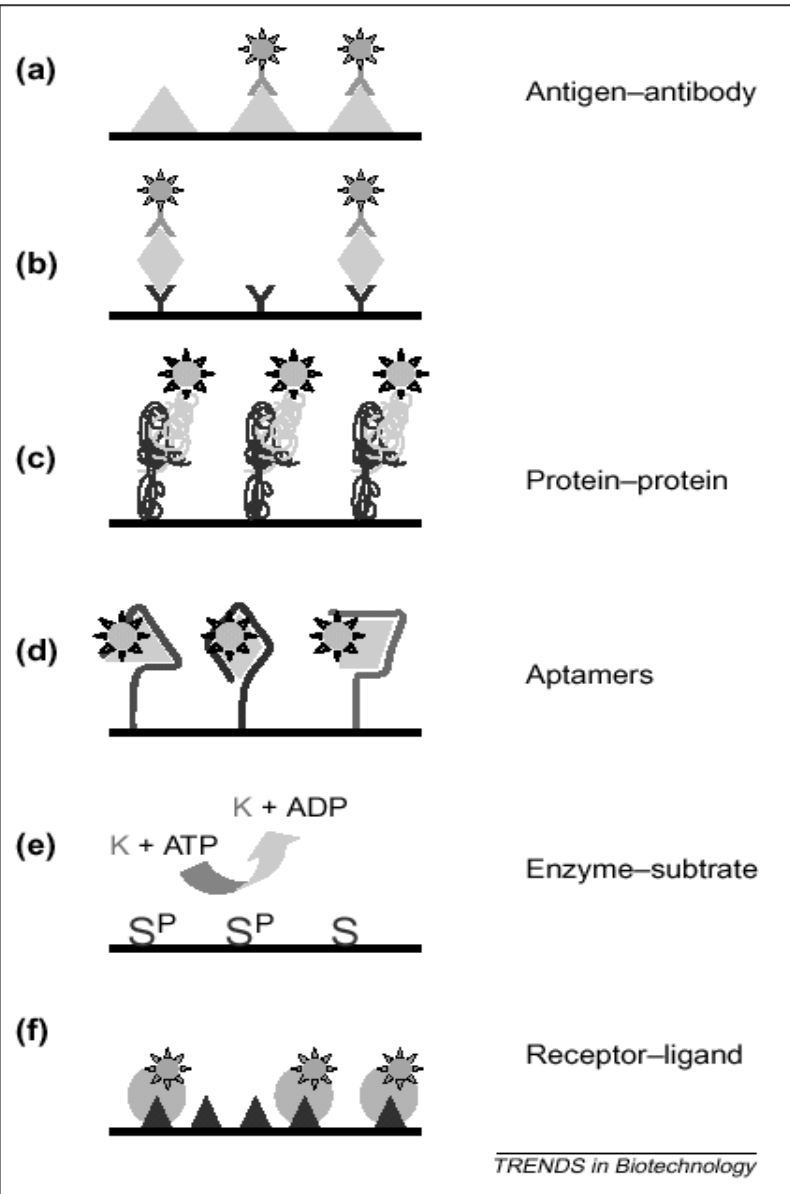
# Protein microarray technology

**Markus F. Templin, Dieter Stoll, Monika Schrenk, Petra C. Traub, Christian F. Vöhringer and Thomas O. Joos**

**Microarray technology allows the simultaneous analysis of thousands of parameters within a single experiment. Microspots of capture molecules are immobilized in rows and columns onto a solid support and exposed to samples containing the corresponding binding molecules. Readout systems based on fluorescence, chemiluminescence, mass spectrometry, radioactivity or electrochemistry can be used to detect complex formation within each microspot. Such miniaturized and parallelized binding assays can be highly sensitive, and the extraordinary power of the method is exemplified by array-based gene expression analysis. In these systems, arrays containing immobilized DNA probes are exposed to complementary targets and the degree of hybridization is measured. Recent developments in the field of protein microarrays show applications for enzyme–substrate, DNA–protein and different types of protein–protein interactions. Here, we discuss theoretical advantages and limitations of any miniaturized capture-molecule–ligand assay system and discusses how the use of protein microarrays will change diagnostic methods and genome and proteome research.**

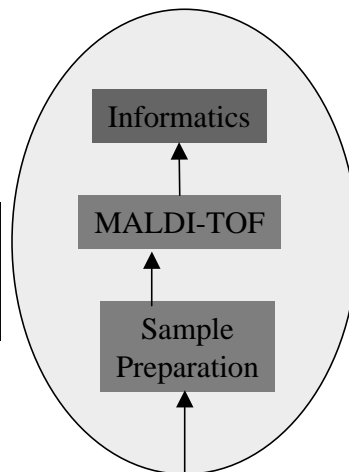
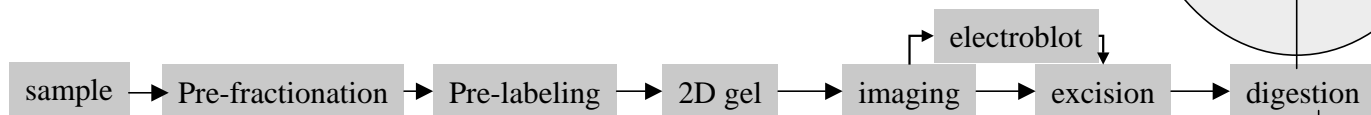
experiment (Fig. 1). Their use for the analysis of single nucleotide polymorphisms and in expression profiling has already changed pharmaceutical research, and their use as diagnostic tools will have a big impact on medical and biological research.

As known from gene expression studies, however, mRNA level and protein expression do not necessarily correlate [7–9]. Protein functionality is often dependent on post-translational processing of the precursor protein and regulation of cellular pathways frequently occurs by specific interaction between proteins and/or by reversible covalent modifications such as phosphorylation. To obtain detailed information about a complex biological system, information on the state of many proteins is required. The analysis of the proteome of a cell (i.e. the quantification of all proteins and the determination of their post-translational

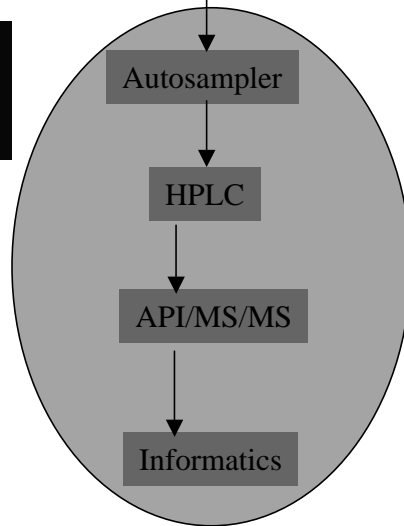


# The Proteomics Flowchart

*Proteomics Solution 1™ (MALDI-TOF)*  
high throughput protein identification

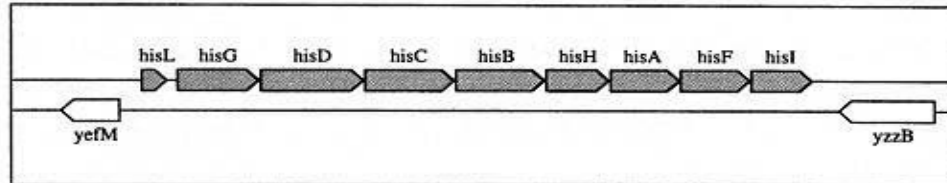


*QSTAR™ Pulsar i Hybrid LC/MS/MS System*  
protein structural characterization

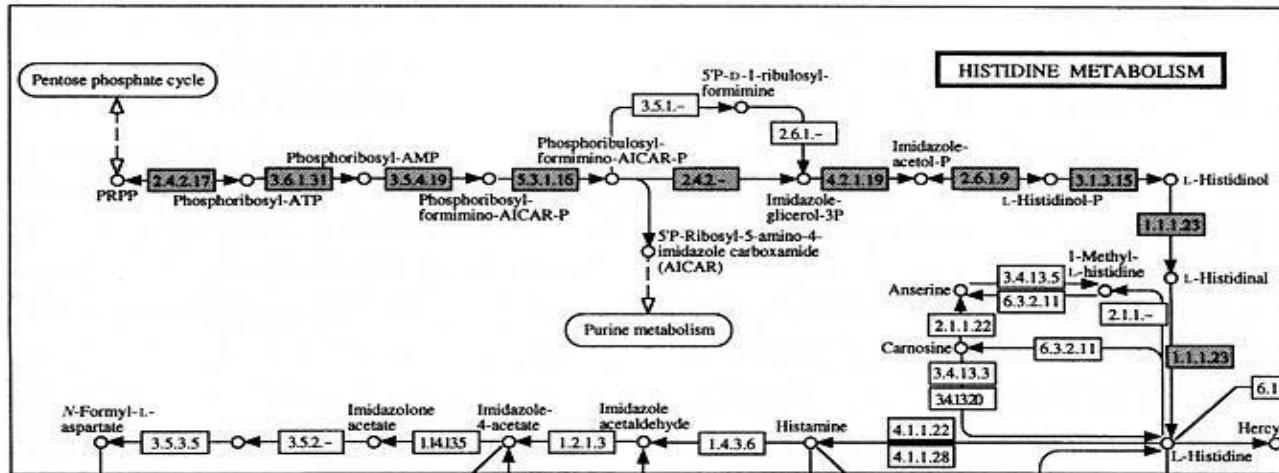


# Genome-Pathway comparison

(a) *E. coli* genome



(b) Metabolic pathway



**Fig. 4.8.** Genome-pathway comparison, which reveals the correlation of physical coupling of genes in the genome (operon structure) and functional coupling of gene products in the pathway.



# Enzymes of glycolysis, the tricarboxylic acid (TCA) cycle and the pentose pathway as defined by genomic annotation in 17 microbial species

| Enzyme                                    | <i>Aquifex aeolicus</i> | <i>Archaeoglobus fulgidus</i> | <i>Bacillus subtilis</i> | <i>Borrelia burgdorferi</i> | <i>Chlamydia trachomatis</i> | <i>Escherichia coli</i> | <i>Haemophilus influenzae</i> | <i>Helicobacter pylori</i> | <i>Methanobacterium thermoautotrophicum</i> | <i>Methanococcus jannaschii</i> | <b><i>Mycobacterium tuberculosis</i></b> | <i>Mycoplasma genitalium</i> | <i>Mycoplasma pneumoniae</i> | <i>Pyrococcus horikoshii</i> | <i>Rickettsia prowazekii</i> | <i>Synechocystis</i> sp. PCC6803 | <i>Treponema pallidum</i> |
|-------------------------------------------|-------------------------|-------------------------------|--------------------------|-----------------------------|------------------------------|-------------------------|-------------------------------|----------------------------|---------------------------------------------|---------------------------------|------------------------------------------|------------------------------|------------------------------|------------------------------|------------------------------|----------------------------------|---------------------------|
| <b>TCA Cycle</b>                          |                         |                               |                          |                             |                              |                         |                               |                            |                                             |                                 |                                          |                              |                              |                              |                              |                                  |                           |
| Citrate synthase                          | +                       | +                             | +                        | -                           | -                            | +                       | -                             | +                          | +                                           | -                               | +                                        | -                            | -                            | -                            | +                            | +                                | -                         |
| Aconitase                                 | +                       | +                             | +                        | -                           | -                            | +                       | -                             | +                          | +                                           | -                               | +                                        | -                            | -                            | -                            | +                            | +                                | -                         |
| Isocitrate DHase                          | +                       | +                             | +                        | -                           | -                            | +                       | -                             | +                          | +                                           | -                               | +                                        | -                            | -                            | -                            | +                            | +                                | -                         |
| $\alpha$ -Ketoglutarate DHase             | -                       | -                             | +                        | -                           | +                            | +                       | +                             | -                          | -                                           | -                               | +                                        | -                            | -                            | -                            | +                            | +                                | -                         |
| Succinyl-CoA synthetase                   | +                       | +                             | +                        | -                           | +                            | +                       | +                             | -                          | +                                           | +                               | +                                        | -                            | -                            | -                            | +                            | +                                | -                         |
| Succinate DHase                           | +                       | +                             | +                        | -                           | +                            | +                       | +                             | +                          | +                                           | +                               | +                                        | -                            | -                            | -                            | +                            | +                                | -                         |
| Fumarase                                  | +                       | +                             | +                        | -                           | +                            | +                       | +                             | +                          | +                                           | +                               | +                                        | -                            | -                            | -                            | +                            | +                                | -                         |
| Malate DHase                              | +                       | +                             | +                        | -                           | +                            | +                       | +                             | -                          | +                                           | +                               | +                                        | -                            | -                            | +                            | +                            | +                                | -                         |
| <b>Glycolysis</b>                         |                         |                               |                          |                             |                              |                         |                               |                            |                                             |                                 |                                          |                              |                              |                              |                              |                                  |                           |
| Hexokinase                                | -                       | -                             | -                        | -                           | -                            | +                       | -                             | +                          | -                                           | -                               | -                                        | -                            | -                            | -                            | -                            | +                                | +                         |
| Phosphoglucose isomerase                  | +                       | -                             | +                        | +                           | +                            | +                       | +                             | +                          | -                                           | +                               | -                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Phosphofructokinase                       | +                       | -                             | +                        | +                           | +                            | +                       | +                             | -                          | -                                           | -                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Fructose-1,6 bis-PO <sub>4</sub> aldolase | +                       | -                             | +                        | +                           | -                            | +                       | +                             | +                          | -                                           | -                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Triose-PO <sub>4</sub> isomerase          | +                       | +                             | +                        | +                           | +                            | +                       | +                             | +                          | +                                           | +                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Glyceraldehyde-3-PO <sub>4</sub> DHase    | +                       | +                             | +                        | +                           | +                            | +                       | +                             | +                          | +                                           | +                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Phosphoglycerate kinase                   | +                       | +                             | +                        | +                           | +                            | +                       | +                             | +                          | +                                           | +                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Phosphoglyceromutase                      | +                       | -                             | +                        | +                           | +                            | +                       | +                             | +                          | -                                           | -                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Enolase                                   | +                       | +                             | +                        | +                           | +                            | +                       | +                             | +                          | +                                           | +                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Pyruvate kinase                           | +                       | -                             | +                        | +                           | +                            | +                       | +                             | -                          | -                                           | +                               | +                                        | +                            | +                            | +                            | +                            | +                                | +                         |
| <b>Pentose PO<sub>4</sub></b>             |                         |                               |                          |                             |                              |                         |                               |                            |                                             |                                 |                                          |                              |                              |                              |                              |                                  |                           |
| Glucose-6-PO <sub>4</sub> DHase           | +                       | -                             | +                        | +                           | +                            | +                       | +                             | +                          | -                                           | -                               | +                                        | -                            | -                            | -                            | -                            | +                                | +                         |
| Lactonase                                 | -                       | -                             | -                        | -                           | -                            | -                       | -                             | -                          | -                                           | -                               | -                                        | -                            | -                            | -                            | -                            | -                                | -                         |
| 6-PO <sub>4</sub> -gluconate DHase        | +                       | -                             | +                        | +                           | +                            | +                       | +                             | +                          | -                                           | -                               | +                                        | +                            | -                            | -                            | -                            | +                                | +                         |
| Ribose-5-PO <sub>4</sub> isomerase        | +                       | +                             | +                        | +                           | +                            | +                       | +                             | -                          | +                                           | +                               | +                                        | -                            | -                            | +                            | -                            | +                                | +                         |
| Ribulose-PO <sub>4</sub> -3-epimerase     | +                       | -                             | +                        | -                           | +                            | +                       | +                             | +                          | -                                           | +                               | +                                        | +                            | -                            | -                            | -                            | +                                | +                         |
| Transketolase                             | +                       | -                             | +                        | -                           | +                            | +                       | +                             | +                          | -                                           | +                               | +                                        | +                            | +                            | -                            | -                            | +                                | +                         |
| Transaldolase                             | +                       | -                             | +                        | -                           | +                            | +                       | +                             | +                          | -                                           | +                               | +                                        | -                            | -                            | -                            | -                            | +                                | +                         |

# Comparison of pathways in different genomes

---

## What do we look for?

- **Alternative pathways - how do they compare?**
- **Which enzyme sets are involved?**
- **Organism-specific adaptations**

## What can we gain?

- **Genome evolution**
- **Biotechnology**
  - **identification of alternative enzymes**
- **Pharmacology**
  - **non-homologous gene displacement; species-specific drug targets**
- **Identification of previously unknown genes**

## Editorial

### TOWARDS COMPUTER AIDED DESIGN (CAD) OF USEFUL MICROORGANISMS

Utilizing microorganisms in industrial processes is one of the most important achievements in the 21st century. Conventional industrial processes usually require energy-consuming conditions such as high temperature and high pressure, and often use hard-to-recycle chemical resources, resulting in harmful impact on the environment such as exhaustion of natural resources and accumulation of industrial waste. Replacing these processes with biology-based technologies could drastically reduce cost, energy, and the destruction of the environment.

At present, microorganisms used in bioprocessing are selected from the existing pool in nature. Research has shown that many of the genes of these microorganisms are unnecessary for application in industrial environments, which, as opposed to the natural environment, are well controlled and constant. For industrial applications, such genes are not only redundant but also often obstructive when expressed, because they waste energy and sometimes even hinder the objective. Trimming microorganisms genetically to optimize their productivity is therefore a key technology of immense industrial importance.

#### TRENDS IN JAPAN

The New Energy and Industrial Technology Development Organization (NEDO) in Japan has launched an innovative research and development project entitled 'Development of Technological Infrastructure for Industrial Bioprocesses'. Its goal is to develop highly-efficient, general-purpose microorganisms with direct applications in industrial bioprocesses. Funds have been provided by the Ministry of Economy, International Trade and

providing raw material for designing a novel genome. The entire data would be fed into E-CELL for computer simulation, enabling creation of novel real cells by genome engineering. Such cells could then be custom-made for specific scientific and industrial applications.

To achieve this highly complex and coordinated task of creating useful microorganisms through CAD (computer aided design), many different technologies must be combined together. These include:

- (1) Enzyme engineering: to refine enzymes and to analyse kinetic parameters *in vitro*.
- (2) Metabolic engineering: to analyse flux rates *in vivo*.
- (3) Analytical chemistry: to determine and analyse the quantity of metabolites efficiently.
- (4) Genetic engineering: to cut-and-paste genes on demand, for modifying metabolic pathways.
- (5) Simulation science: to efficiently and accurately simulate a large number of reactions.
- (6) Knowledge engineering: to construct, edit, and maintain large metabolic knowledge bases.
- (7) Mathematical engineering: to estimate and tune unknown parameters.

#### WHOLE CELL SIMULATION

Computer modeling and simulation is an essential part of this new technology. A major challenge is to construct a computer model of the whole microbial cell. The E-CELL Project (<http://www.e-cell.org>) was launched in 1996 at Keio University to model and simulate various cellular processes with the ultimate goal of simulating the cell as a whole. In 1997, we successfully constructed a virtual cell with 127 genes sufficient for 'self-support'. The gene set was selected from the genome of *Mycoplasma genitalium*. Processes included transcription, translation,

models of the whole cell, various cellular processes have nevertheless been modeled by many different research groups. Our E-CELL project team, among others, has been working on bacterial chemotaxis, circadian rhythms, photosynthesis, cell cycle and cell division. For modeling and simulating gene expression, we are working on general quantitative models and their application to gene regulation network, especially *E.coli* lactose operon and lambda phage genetic switch. For organelles, a quantitative model of mitochondria is nearly complete. For human cells, we have already developed a quantitative model of erythrocytes, and is currently being used in modeling and simulating enzyme deficiencies causing anemia. Other human cells in the E-CELL system pipeline are myocardial cells, neural cells, and pancreatic beta-cells.

One of the major problems in constructing large-scale cell models is lack of quantitative data. Most of the biological knowledge available in the literature is qualitative, e.g. gene function, pathway maps, protein-protein interaction, etc. For simulation, quantitative data such as concentrations of metabolites and enzymes, flux rates, kinetic parameters and dissociation constants are needed. The limited availability of precise quantitative data from the published literature makes metabolome projects indispensable for cell modeling, for it enables measurement of various components systematically and accurately.

## **METABOLOME RESEARCH**

The Metabolome group of our institute is developing methodologies for mass-production of quantitative metabolic data. We analyse metabolic flux distributions

More than 30 positively charged metabolites including amino acids and amines have been roughly separated by CE and selectively detected by MS. Additionally, many negatively charged metabolites such as organic acids have been simultaneously analysed in other conditions. Our method is simple, rapid, and sensitive and can be readily applied to the real samples. In the future, we extend this method to enable us to measure hundreds of metabolites at once.

Thus, it is possible to collect large amounts of data for a variety of cell states to construct quantitative models. These models can be refined iteratively until simulation results match a given set of data.

## **EPILOGUE**

For long, bioinformatics has played a role of technical infrastructure for research in experimental biology. Now the time has come for a 'paradigm shift' to look the other way around. Many bioinformatics projects, such as computer modeling and simulation of cellular processes, recently need a large amount of biological data, and thus well-designed, systematic experimentation is becoming an important infrastructure for those bioinformatics projects. Perhaps, these days it makes little sense to make a distinction between bioinformaticians and experimental biologists. After all, biology cannot stand without informatics, and bioinformatics itself is, in my view, a new type of biology.

Masaru Tomita  
Institute for Advanced Biosciences  
Keio University



日本慶應大學  
YM-Genetics

## **E-CELL: software environment for whole-cell simulation**

Masaru Tomita<sup>1</sup>, Kenta Hashimoto<sup>1</sup>, Kouichi Takahashi<sup>1</sup>,  
Thomas Simon Shimizu<sup>1,3</sup>, Yuri Matsuzaki<sup>1</sup>, Fumihiko Miyoshi<sup>1</sup>,  
Kanao Saito<sup>1</sup>, Sakura Tanida<sup>1</sup>, Katsuyuki Yugi<sup>1</sup>, J. Craig Venter<sup>2</sup>  
and Clyde A. Hutchison III<sup>2</sup>

<sup>1</sup>Laboratory for Bioinformatics, Keio University, 5322 Endo, Fujisawa, 252, Japan and  
<sup>2</sup>The Institute for Genomic Research, 9712 Medical Center Drive, Rockville, MD 20850,  
USA

Received on May 1, 1998; revised on October 16, 1998; accepted on November 2, 1998

### **Abstract**

**Motivation:** Genome sequencing projects and further systematic functional analyses of complete gene sets are producing an unprecedented mass of molecular information for a wide range of model organisms. This provides us with a detailed account of the cell with which we may begin to build models for simulating intracellular molecular processes to predict the dynamic behavior of living cells. Previous work in biochemical and genetic simulation has isolated well-characterized pathways for detailed analysis, but methods for building integrative models of the cell that incorporate gene regulation, metabolism and signaling have not been established. We, therefore, were motivated to develop a software environment for building such integrative

580 kb genome sequence was determined at TIGR in 1995. We discuss future applications of the E-CELL system with special respect to genome engineering.

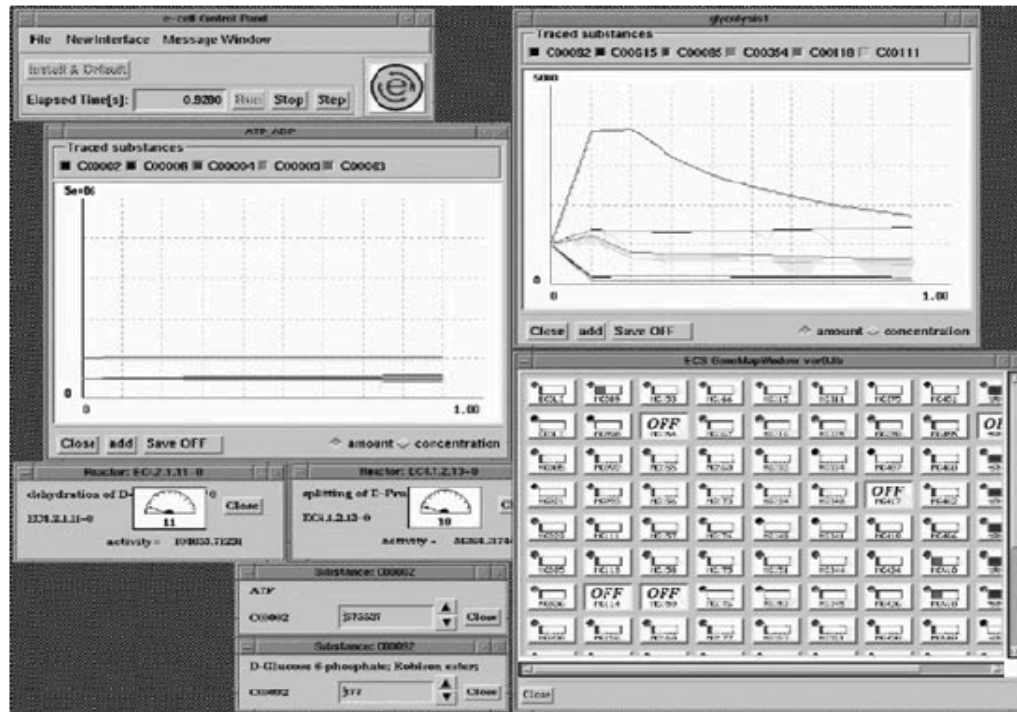
**Availability:** The E-CELL software is available upon request.

**Supplementary information:** The complete list of rules of the developed cell model with kinetic parameters can be obtained via our web site at: <http://e-cell.org/>.

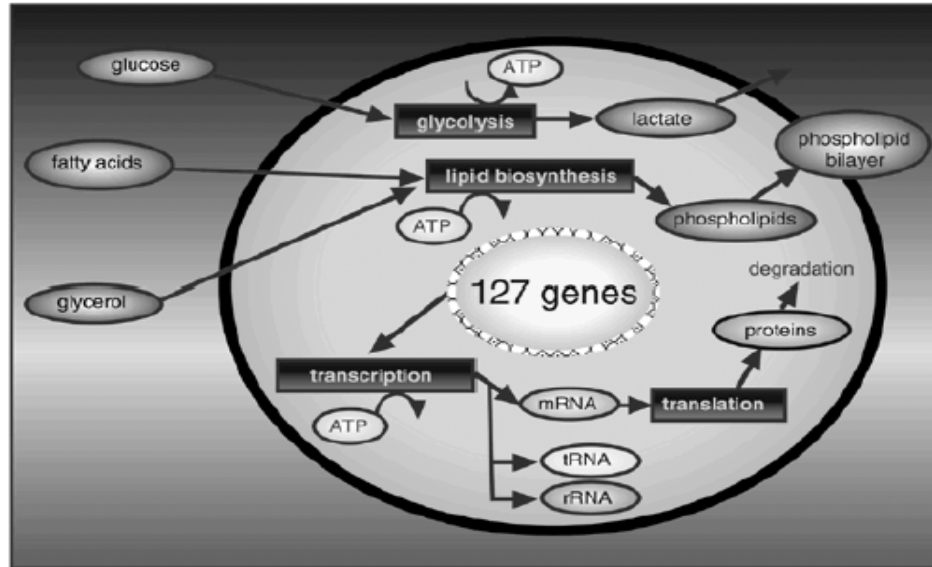
**Contact:** [mt@sfc.keio.ac.jp](mailto:mt@sfc.keio.ac.jp)

### **Introduction**

The complete genomes of more than 18 microorganisms have been sequenced. The availability of this new information on the gene content of organisms has led to the emergence of a



**Fig. 1.** A snapshot of user interfaces of the E-CELL system. The tracer window for ‘glycolysis1’ (upper right) shows dynamic changes in quantities of glycolytic metabolites: D-glucose 6-phosphate (C00092), protein histidine (C00615), D-fructose 6-phosphate (C00085), D-fructose 1,6-biphosphate (C00354), D-glyceraldehyde 3-phosphate (C00118) and glycerone phosphate (C00111). The other tracer window (left) shows changes in quantities of ATP (C00002), ADP (C00008), NADH (C00004), NAD<sup>+</sup> (C00003) and CTP (C00063). Two reactor windows (lower left) show activities of phosphopyruvate hydratase (EC 4.2.1.11) and fructose-biphosphate aldolase (EC 4.1.2.13). Two substance windows (bottom left) show precise quantities of ATP (C00002) and D-glucose 6-phosphate (C00092). The GeneMapWindow (bottom right) shows current activities (the number of mRNA molecules) of all genes in the cell. Different colors indicate an increase or decrease of activities. Knocked-out genes are marked ‘OFF’.



**Fig. 2.** Metabolism overview of the model cell. It has pathways for glycolysis and phospholipid biosynthesis, as well as transcription and translation metabolisms.

# Computerized role models

Japan's push to create a virtual cell signals a new approach to research, says Robert Triendl.

In many ways, the year-old Institute for Advanced Biosciences at Keio University in Tsuruoka is an anomaly of Japanese science. It is run by a relatively young scientist, uses short-term appointments, and it gives its junior faculty members a great deal of academic freedom.

The institute was established last year with a grant from the Yamagata prefecture with additional funding from Keio University, Japan's top-ranked private university. It is built around E-CELL, an ambitious idea by its director, Masaru Tomita, to develop a realistic computer simulation of an entire living cell.

Tomita trained as a computer scientist at Keio and at Carnegie Mellon University in Pittsburgh, where he



are renewable after three years.

Although this policy is still unusual in a country with an extremely tight academic labour market, Tomita says he voted for it despite the university's reservations. "Three years is a short period of time,

**Virtual world: researchers at the Institute for Advanced Biosciences aim to generate accurate computer models of living cells.**

# Silicon dreams in the biology lab

## Beyond Bioinformatics

division.

### IMPROVING ACCESSIBILITY

These types of initiative are providing a few examples of high-profile, but relatively isolated, activity. If computational cell biology is to be accepted by more mainstream molecular and cell biologists, "we have to have some user-friendly interfaces that lower the activation barrier for experimental biologists to start thinking about mathematical models", says Tyson.

Efforts such as the Virtual Cell project, developed as part of the University of Connecticut's National Resource for Cell Analysis and Modeling (NRCAM) and funded by the National Institutes of Health (NIH), have helped. Virtual Cell, which is available free on the Internet for non-commercial use, provides a generalized modelling and simulation framework that biologists without a mathematical background — and with a little training — can use.

Other examples include the E-CELL simulation software, developed at Keio University in Japan (see "Computerized role models", page 7), and the Gepasi biochemical kinetics simulator, developed at the University of Wales, Aberystwyth, UK.

The US defence department's Defense Advanced Research Projects Agency also began a major biocomputing initiative last year. The agency plans to

Virtual Cell modelling and simulation framework

▶ [www.nrcam.uchc.edu](http://www.nrcam.uchc.edu)

Gepasi biochemical kinetics simulation package

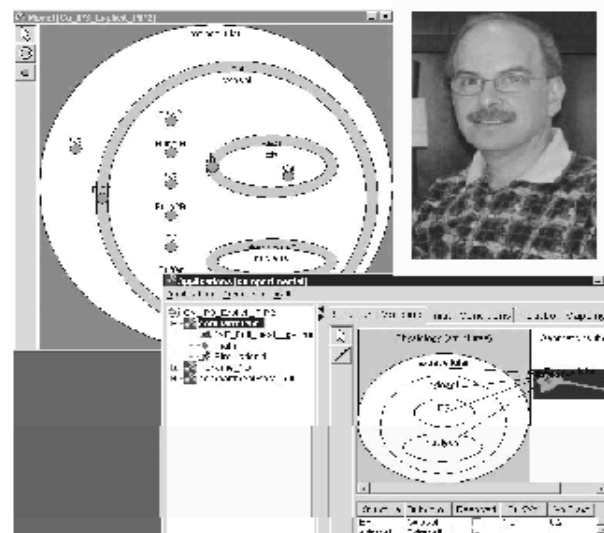
▶ [www.gepasi.org](http://www.gepasi.org)

Alliance for Cellular Signaling

▶ [www.cellularsignaling.org](http://www.cellularsignaling.org)

Cell Migration Consortium

▶ [www.cellmigration.org](http://www.cellmigration.org)



Leslie Loew believes computational efforts such as the Virtual Cell project have a lot to offer experimental biologists.