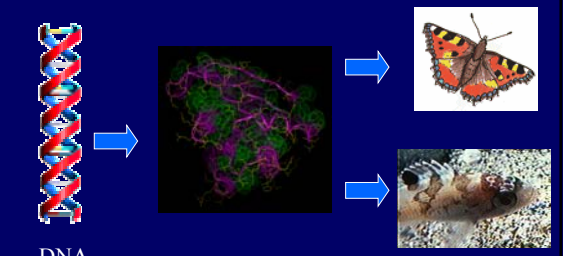
Large Scale Bioinformatics

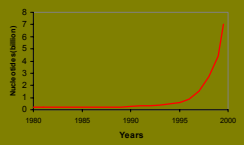
Ming Li
CRC Chair in Bioinformatics
School of Computer Science
University of Waterloo

Biology



Large Scale Bioinformatics

- The trend of genetic data growth



20 billion bases in GenBank in 2002. Doubling every 18 months

- Human Genome: 3 billion bases
- Human Proteome: trillions of cells, thousands of proteins expressed in different levels in a cell
- Sharcnet: Genome alignment, protein folding

Large-scale problems:

- Genome-Genome alignments
  - Spaced seeds.
- Protein 3D structure prediction
  - New LP mathematical formulation
- Whole genome phylogeny

1. Scalable Homology Search

- Daily task facing molecular biologists: I have a DNA/protein sequence, find something "similar" in GenBank.
- Search size: billions.
- A fraction of world's supercomputing power is consumed by homology search, using a program called Blast. (download from http://www.ncbi.nlm.nih.gov)
- Blast paper is the most referenced paper in the last decade, over 100,000 times.
- Other programs: FASTA, MegaBlast, WU-Blast, SIM
- None of these is scalable to genome scale. Blast takes 19 years to compare human and mouse genome sequences. Industry depend on supercomputers.

Blast Algorithm

- Find seeded matches
- Extent to HSP's (High Scoring Pairs)
- Extension, dynamic programming
- Report all local alignments

PatternHunter (*Nature*, Dec 5, 2002)

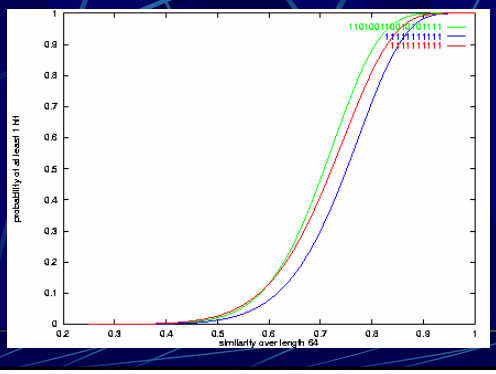
(Ma, Tromp, Li: *Bioinformatics*, 18:3, 2002, 440-445)

- Comparing mouse genome vs human genome
  - Blast: 20 years
  - PatternHunter: 19 days.
- Used in Mouse Genome Consortium as well as in hundreds of institutions and industry (Celera, deCODE, ALTANA)

New idea: Optimized Seed

- Blastn finds a match of length 11, then extend from there.
- Spaced Seed: PatternHunter looks for matches of 11 nonconsecutive matches and optimized such seeding scheme.
- That is:
  - Blast seed: 11111111111
  - PatternHunter seed: 111010010100110111
- This seemingly simple change makes a huge difference: significantly increases hit to homologous region while reducing bad hits.

Sensitivity: PH weight 11 seed vs Blast 11 & 10



Hitting the right place

Lemma: The expected number of hits of a weight W length M seed within a length L region with similarity p is

$$(L - M + 1)p^W$$

- That is, at the same seed weight:
  - BLAST and PatternHunter both have same number of hits.
  - However, PatternHunter hits good regions and BLAST hits bad regions.

Problem Partition

- Mouse Genome Consortium: comparing mouse genome and human genome.
- 3 billion bases against 3 billion bases.
- This problem can be easily split into subproblems: chromosome against chromosome --- 23 (human) against 20 (mouse) and using separate processors

2. Protein Structure Prediction

- Post-genomics era, proteomics is the new focus. A key issue: understand proteins via their 3-D structures.
- Wet lab protein structure determination is costly and low throughput. Computational methods are cheap and fast.
- Protein threading is promising. But it is NP-hard.
- New ideas are needed to do better protein structure prediction.

Structural Fold Space and Structure Prediction

- Basic premise

  The number of unique structural (domain) folds in nature is fairly small (possibly a few thousand)

- Statistics from PDB

  90% of new structures submitted to PDB in the past three years have similar structural folds in PDB

- Proteins with similar structural folds could be homologues or analogues
- Protein threading: a protein structure prediction technique based on known protein structures

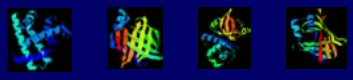
Protein Threading

- Make a (backbone) structure prediction through finding an optimal placement (threading) of a protein sequence onto each known structure (structural template)
  - "placement" quality is measured by some statistics-based energy function
  - best overall "placement" among all templates may give a structure prediction -- also depending on additional criteria

target sequence

MTYKLILNGKTKGETTTEAVDAATAEKVFQYANDNGVDGEWTYTE

template library



Threading Energy and Energy Optimization

MTYKLILNGKTKGETTTEAVDAATAEKVFQYANDNGVDGEWTYTE

how preferable to put two particular residues nearby: E_p

how well a residue fits a structural environment: E_s

alignment gap penalty: E_g

sequence similarity between query and template proteins: E_m

total score: E = E_p + E_s + E_m + E_g + E_ss

Find a sequence-structure alignment to optimize this function

The sequence-structure alignment problem is much more difficult than a sequence-sequence alignment problem

Most threading algorithms do not treat pair-contact energy rigorously to avoid high computational cost

RAPTOR

Xu, Li, et al, CASP5 and PSB 2003

- We implemented RAPTOR using a new approach of Linear programming.
- Raptor was ranked ***number 1*** in CAFASP3/CASP5 among non-meta automatic protein 3D dimensional structure prediction programs (for fold recognition), Dec. 2002
- CASP is the leading international conference for protein structure prediction competition.
- We used Flexor at Waterloo.

New Idea: Linear Programming

- Formulate the problem of minimizing E as an Integer Programming problem, as seen on the right.
- But Integer Programming is NP-hard.
- So we relax it to Linear Programming, allowing solutions to be non-integral, changing x in {0,1} to 0<=x<=1 in last line.
- LP is not NP-hard. Can be solved by well-known polynomial time elipsoid methods, or heuristic (fast) Simplex method.
- Then we worry about how to convert a non-integral solution to an integer solution.
- In practice, almost all solutions are integral already.

$\min \{w_1 E_{single} + w_2 E_{mutate} + w_3 E_{pair} + w_4 E_{gap}\}$

$E_{single} = \sum_{i=1..d;\ j=1..n} x_{i,j}$ Fitness(i,j)

$E_{mutate} = \sum_{i=1..d;\ j=1..n} x_{i,j}$ Mutation(i,j)

$E_{pair} = \sum_{i,j,i',j'} y_{i,i',j,j'}$ Pair(i,i',j,j')

$E_{gap} = \sum_{i,i',(i+1)'} y_{i,i',i+1,(i+1)'}$ Gap(i,i',i+1,(i+1)')

subject to:

$\sum_{j=1..n} x_{i,j} <= 1, \quad i = 1, 2, .., d;$

$\sum_j x_{i2,j} <= \sum_j x_{i1,j} + \sum_j x_{i3,j} - 1$

for all i1 <i2 <i3, i2=(i1 + i3) / 2

$x_{i,j0} + \sum_{j <= j0} x_{i+1,j} <= 1,$ for all i, j0

$y_{i,i',j,j'} <= x_{i,i'},$ for all i, i', j, j'

$y_{i,i',j,j'} <= x_{j,j'},$ for all i, i', j, j'

$y_{i,i',j,j'} >= x_{i,i'} + x_{j,j'} - 1,$ for all i, i', j, j'

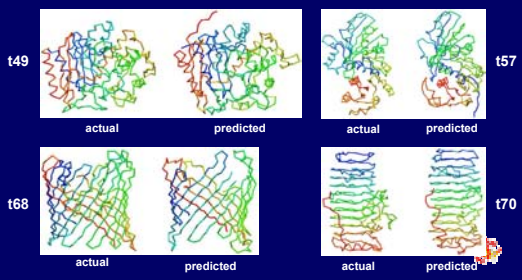
$y_{i,i',j,j'}, x_{i,j}$ in {0, 1}.

RAPTOR performance

(on Lindahl dataset, 976 proteins)

% correct	Family Only		Superfamily Only		Fold Only	
Methods	top 1	top 5	top 1	top 5	top 1	top 5
RAPTOR	84.8	87.1	47.0	60.0	31.3	54.2
PROSPECT	84.1	88.2	52.6	64.8	27.7	50.3
FUGUE	82.2	85.8	41.9	53.2	12.5	26.8
Phi-Blast	71.2	72.3	27.4	27.9	4.0	4.7
HMMER-PHIBlast	67.7	73.5	20.7	31.3	4.4	14.6
SAMT98-PHIBlast	70.1	75.4	28.3	38.9	3.4	18.7
BlastLink	74.6	78.9	29.3	40.6	6.9	16.5
SSearch	68.6	75.7	20.7	32.5	5.6	15.6
Threader	49.2	58.9	10.8	24.7	14.6	37.7

Some predictions



3. Whole Genome Phylogeny

Who are our closer relatives?

(primates, ferungulates), rodents

or

(primates, rodents), ferungulates?

Problem Partition

- RAPTOR computes alignment of target against about 3000 templates.
- This can be split into 3000 separate processes and compute separated on different processors

Evolution of Chain Letters

- Charles Bennett collected 33 chain letters 1980--1997. Wanted to know their evolutionary history. Our work appeared in *Scientific American, June, 2003.*
- Like a gene, they are about 2000 characters; like a virus, they have infected billions of people, they mutate just like a genome, even has horizontal transfer. Traditional phylogeny methods don't work: (a) multiple alignment fail--translocated sentences; (b) no model of evolution.

A sample letter:

Other related questions

- Program plagiarism.
- Language evolution
- Information clustering
- Internet similar document search

New Approach

- Shared Information between any two sequences:

$$d(x,y) = 1 - \frac{K(x) - K(x|y)}{K(xy)}$$

- $K(x/y)$ is Kolmogorov complexity of x condition on y, defined as the length of shortest program that outputs x on input y.
- $K(x) – K(x|y)$ is mutual information

Mathematically sound:

- It can be proved, for any other computable normalized measure D(x,y) that satisfies some reasonable neighborhood density property, then we have: there is a constant c such that for all x,y, we have

$$d(x,y) < cD(x,y)$$

- Informally speaking: any similarity detected by D is also detected by d !

Traditional Phylogenetic Methods

- Consider a gene seq. from each of k species. Build a "gene tree" based on one of the following methods.
- Typical algorithms:
  - Max likelihood method – fastNDAml/PROTml (Phylip, MOLPHY)
  - Neighbor Joining
  - Parsimony (PULP)
  - UPGMA
  - Quartet Puzzle
  - Quartet cleaning (SODA'00)
  - PTAS – finding the most consistent tree (FOCS98).
- All these methods require alignments, not scalable to genome scale. Gene trees give conflicting results.

Whole Genome Phylogeny

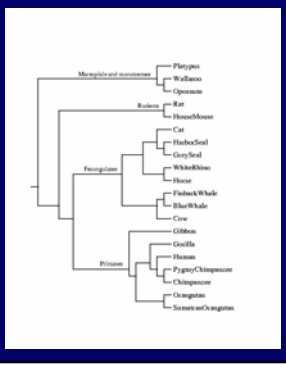
The new distance provides a new approach. It

- uses all the information in a genome;
- completely automated;
- needs no alignment; robust, fast, accurate;
- no model -- simply universal;
- generalizes all other distances

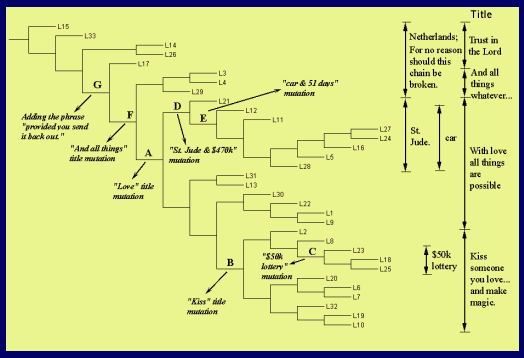
Eutherian Orders:

- It has been a disputed issue which two groups of placental mammals are closer: Primates, Ferungulates, Rodents.
- Hasegawa's group concatenated 12 proteins from: rat, house mouse, grey seal, harbor seal, cat, white rhino, horse, finback whale, blue whale, cow, gibbon, gorilla, human, chimpanzee, pygmy chimpanzee, orangutan, sumatran orangutan, with opossum, wallaroo, platypus as out group. They used Max likelihood method
- We used our new approach and constructed the whole genome phylogeny on the same dataset, obtained the tree and verified: ((primates, ferungulates), rodents)

Evolutionary Tree of Mammals:

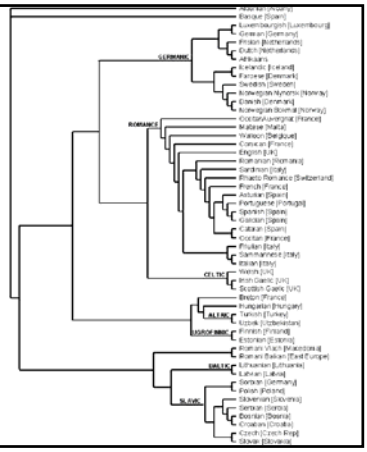


Phylogeny of 33 Chain Letters



Confirmed by VanArsdale's study, answers an open question

Another Application: A language tree created using UN's The Universal Declaration Of Human Rights



Program Plagiarism Test

- Same method was used in SID (http://genome.uwaterloo.ca/SID/) to detect program plagiarism.
- SID parses the program, unify variable names, delete comments, computes shared amount of information between each pair of programs.
- Because our measure is universal, theoretically, SID is not cheatable.

Acknowledgements

- Many people have contributed to the work presented here: B. Ma, J. Tromp, J. Xu, Y. Xu, D. Xu, G.H. Lin, D. Kim, D. Kisman, C.H. Bennett, P. Vitanyi, X. Chen