Computational Biology

Lecture #6: Haplotypes & Disease Modeling

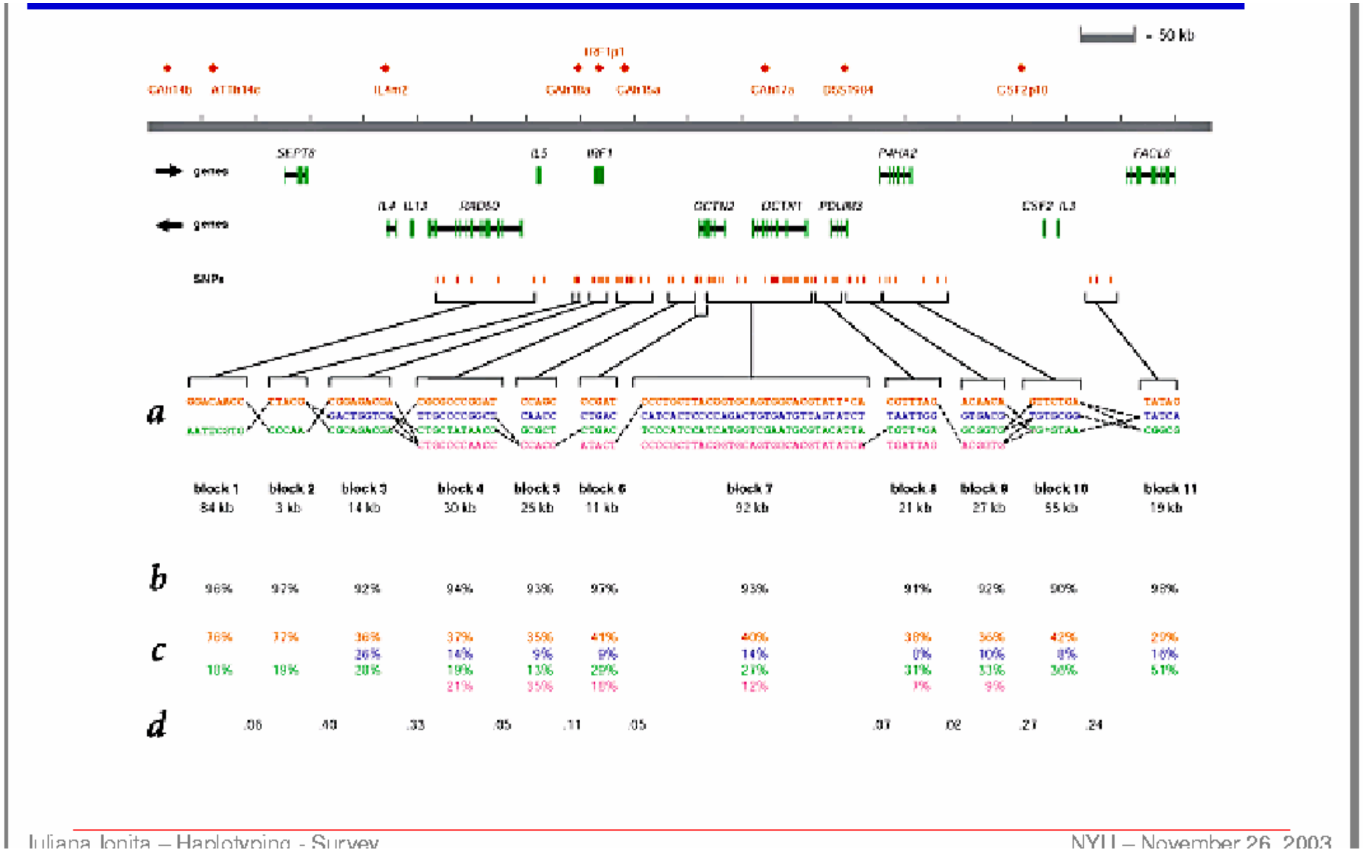
Bud Mishra

Professor of Computer Science, Mathematics, & Cell Biology

Oct 24 2005

Blocky Genome

– Picture Daly et al.



Iuliana Ionita – Haplotyping - Survey

NYU – November 26, 2003

Blocky Genome

- Daly et al.(2001) Study on haplotypes
  - A genomic region on chromosome 5
  - Found that the region can be partitioned into 11 blocks of size up to 100 kb such that in each block there is very little variation.
  - In each block only a few haplotypes (2-4) account for over 90% of the haplotypes in the sample.
  - Inside the blocks there is no or very little evidence for recombination, whereas between blocks there are hot-spots of recombination.

Blocky Genome

- Reducing the complexity of the genome.
  - Having such an extended LD is important because it means that only few sites encode the information present in the entire region. (Knowing the information at these sites gives you the entire haplotype).
  - So no need to genotype all sites.
- Motivated by these findings, several deterministic and Bayesian algorithms analyze data specifically specifically exploiting these blocks of limited diversity.

THE HAPLOTYPING PROBLEM

- Single Individual:
  - Given genomic data of one individual, determine 2 haplotypes (one per chromosome)
- Population :
  - Given genomic data of k individuals, determine (at most) 2k haplotypes (one per chromosome/indiv.)
- Under different objective functions

HAPLOTYPING

- For the individual problem, input is erroneous haplotype data, from sequencing & mapping.
- For the population problem, data is ambiguous genotype data, from screening

- Objective Function is gverened by Occam's razor:find minimum explanation of observed data under given hypothesis (Parsimony Principle, Maximum Likelihood)

Theory and Results

- Single individual
  - Polynomial Algorithms for gapless haplotyping (Lancia, Bafna, Istrail, Lippert, Schwartz 01)
  - Polynomial Algorithms for bounded-length gapped haplotyping (Bafna, Lancia, Istrail, Rizzi 02)
  - NP-hardness for general gapped haplotyping (Lancia, Bafna, Istrail, Lippert, Schwartz 01)
- Population
  - Parsimony (Gusfield 03, Lancia, Rizzi, Pinotti 02)
  - Clark's rule: APX-hardness and I.P. approach (Gusfield 00 & 01)
  - Polynomial algorithm for perfect phylogeny (Bafna, Gusfield, Lancia, Yooseph 02)
  - Formulations for Disease Detection (Lancia, Pesole 02)

Shotgun Assembly of a Chromosome

ACTGAGCCTAGAGATTTCTAGGCGTATCTATCTTACACTGCATCGATCGATCGATCGA

fragmentation

ACTGA GATTT GCCTAG CTATCTT
ATAGATA GAGATTTC TAGAAATC TGAGCCTAG
TAGAGATTTC TCCTAAAGAT CGCATAGATA

sequencing

TGAGCCTAG GATTT GCCTAG CTATCTT
ATAGATA GAGATTTCTAGAAATC ACTGA
TAGAGATTTC TCCTAAAGAT CGCATAGATA

assembly

```
ACTGCAG          ATTCTCAGA           GGCGT      TCTT
   TGCAGCCTA   GATTCTC             CTAGG   TATCTATCTT
ACTGC     CTAGAGAT        GATATTTCTAG      TATCT
```

ACTGCAGCCTAGAGATTCTCAGATATTTCTAGGCGTATCTATCTT

- [Webber and Myers, 1997]

ERROR SOURCES

Sequencing errors:

```
ACTGCCTGGCCAATGGAACGGACAAG
     CTGGCCAAT
          CATTGGAAC
           AATGGAACGGA
```

Paralogous regions:

```
ACAAACCCTTTGGGACT … CTAGTAAACCCTATGGGGA
  AAACCCTT                TAAACCCT
       CTATGGGA                CCTATGG
       CTTTGGGACT            ACCCTATGGG
```

Individual Haplotyping

  - Given errors (sequencing errors, and/or paralogous) the data may be inconsistent with exactly 2 haplotypes
  - Hence, assembler is unable to build 2 chromosomes
- PROBLEM:
- Find and remove the errors so that the data becomes consistent with exactly 2 haplotypes

The data: a SNP matrix

```
ACTGAAAGCGA       ACTAGAGACAGCATG
ACTGATAGC          GTAGAGTCA
ACTG           TCGACTAGA     CATG
ACTGA   CGATCCATCG       TCAGC
ACTGAAA   ATCGATC          AGCATG
```

```
ACTGAAAGCGA       ACTAGAGACAGCATG
ACTGATAGC          GTAGAGTCA
ACTG           TCGACTAGA     CATG
ACTGA   CGATCCATCG       TCAGC
ACTGAAA   ATCGATC          AGCATG
```

```
     X             X     O
     O             O     X
                   X
             X           X
     X       O
```

Resolving a SNP matrix

Snips *1,..,n*

$$\begin{array}{c|ccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
\hline
1 & - & - & - & 0 & X & X & 0 & 0 & - \\
2 & - & 0 & - & 0 & X & - & - & - & X \\
3 & X & X & 0 & X & X & - & - & - & - \\
4 & 0 & 0 & X & - & - & - & - & 0 & - \\
5 & - & - & - & - & - & - & - & X & 0 \\
6 & - & - & - & - & 0 & 0 & 0 & X & -
\end{array}$$

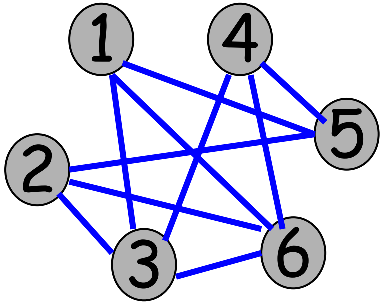
Fragments *1,..,m*

Fragment Conflict Graph

Snips *1,..,n*

Fragments *1,..,m*

	1	2	3	4	5	6	7	8	9
1	-	-	-	0	X	X	0	0	-
2	-	0	-	0	X	-	-	-	X
3	X	X	0	X	X	-	-	-	-
4	0	O	X	-	-	-	-	0	-
5	-	-	-	-	-	-	-	X	0
6	-	-	-	-	0	0	0	X	-



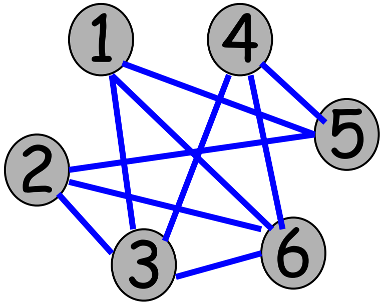
- Fragment conflict:
  - Cannot be on same haplotype
  - Summearized by a Fragment Conflict Graph GF(M)
- Theorem:
  - We have 2 haplotypes iff GF is BIPARTITE
- PROBLEM (Fragment Removal):
  - Make GF Bipartite

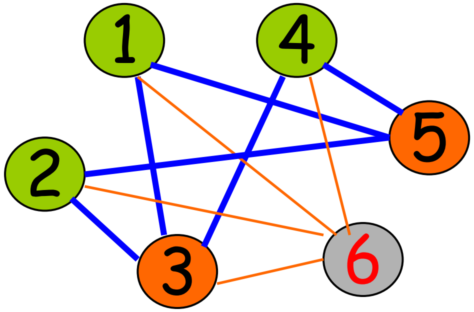
Removing Fragments

← Snips *1,..,n* →

Fragments *1,..,m*

	1	2	3	4	5	6	7	8	9
1	-	-	-	0	X	X	0	0	-
2	-	0	-	0	X	-	-	-	X
3	X	X	0	X	X	-	-	-	-
4	0	0	X	-	-	-	-	0	-
5	-	-	-	-	-	-	-	X	0
6	-	-	-	-	0	0	0	X	-





	1	2	3	4	5	6	7	8	9
1	-	-	-	0	X	X	0	0	-
2	-	0	-	0	X	-	-	-	X
4	0	0	X	-	-	-	-	0	-
	0	0	X	0	X	X	0	0	X
3	X	X	0	X	X	-	-	-	-
5	-	-	-	-	-	-	-	X	0
	X	X	0	X	X	-	-	X	0

Complexity

- Removing fewest fragments is equivalent
  - to maximum induced bipartite subgraph
  - NP-complete [Yannakakis, 1978a, 1978b; Lewis, 1978]
  - $O(|V|(\log \log |V|/\log |V|)2)$-approximable [Halldórsson, 1999]
  - not $O(|V|\varepsilon)$-approximable for some ε [Lund and Yannakakis, 1993]
- THEOREM
  - For a gapless M, the Min Fragment Removal Problem is Polynomial

Population Haplotypes

- Clark's Algorithm
- Perfect Phylogeny Solutions
- Statistical Solutions

Clark's Algorithm -1990

- This is a parsimony approach
  - It tries to solve the genotypes in the data set with as few haplotypes as possible.
  - It starts with the list of haplotypes that can by unambiguously inferred from the genotype data, i.e. the ones coming from homozygous or single-site heterozygous individuals.
  - It then tries to solve the phase ambiguous individuals by using these already determined haplotypes.

Example

- For the data set that we have, we know that the following haplotypes are present in the population: {AGT, ACT, AGA, TGA}
- Now, for each known haplotype we traverse the list of ambiguous individuals and ask whether each individual can be solved by that haplotype: e.g. {A,T} {G, C} {T, T}, can be solved as AGT and TCT.
- By doing this we also acquired a new haplotype (TCT) that we add to the end of the list. We do this process until either all individuals are resolved or we can't find any more solutions.

Problems

- There are a few problems with this algorithm.
  - It might not get started
  - It might not resolve all individuals
  - It depends on the order in which one examines the genotypes
  - It performs poorly compared to other existing algorithms when too few homozygotes are in the data.
- Simple and Popular.
  - No limit on the number of SNPs it can handle
  - Other variations (e.g. The Consensus Solution)

Perfect Phylogeny

- Gusfield (2002)
- Bafna et al. (2002)
- Eskin et al. (2002)
- New Results (2005) Gusfield et al.

Perfect Phylogeny 2001

- The Perfect Phylogeny model of haplotype evolution
  - It assumes that there is no recombination and the usual infinite-site mutation model of population genetics applies.
  - Given the existence of these blocks, the PP model seems a reasonable model when working with SNP data.

Perfect Phylogeny

- The first paper assuming this model [Gusfield (2002)].
  - The solution presented is a reduction of the haplotype inference problem to a problem in graph theory called the "graph realization problem." This problem has an optimal solution - almost linear time. But it is very difficult to implement.

Perfect Phylogeny

- A simpler solution[Bafna et al. (2002) and by Eskin et al. (2002)]
  - Uses no complex tools and is very easy to implement.
- Relatively Fast:
  - The time complexity of these algorithms is $O(ns^2)$ where n is the number of individuals and s the number of sites.
- Limited Applicability
  - They can only be applied on haplotype blocks with no recombination.

Genotypes and Haplotypes

- Each individual has two "copies" of each chromosome.
  - At each site, each chromosome has one of two alleles (states) denoted by 0 and 1 (Biallelic SNPs)
  - Diploid individuals
  - Merged haplotypes gives genotypes:
    - 0+0 = 0
    - 1+1 = 1
    - 0+1=2

0 1 1 1 0 0 1 1 0

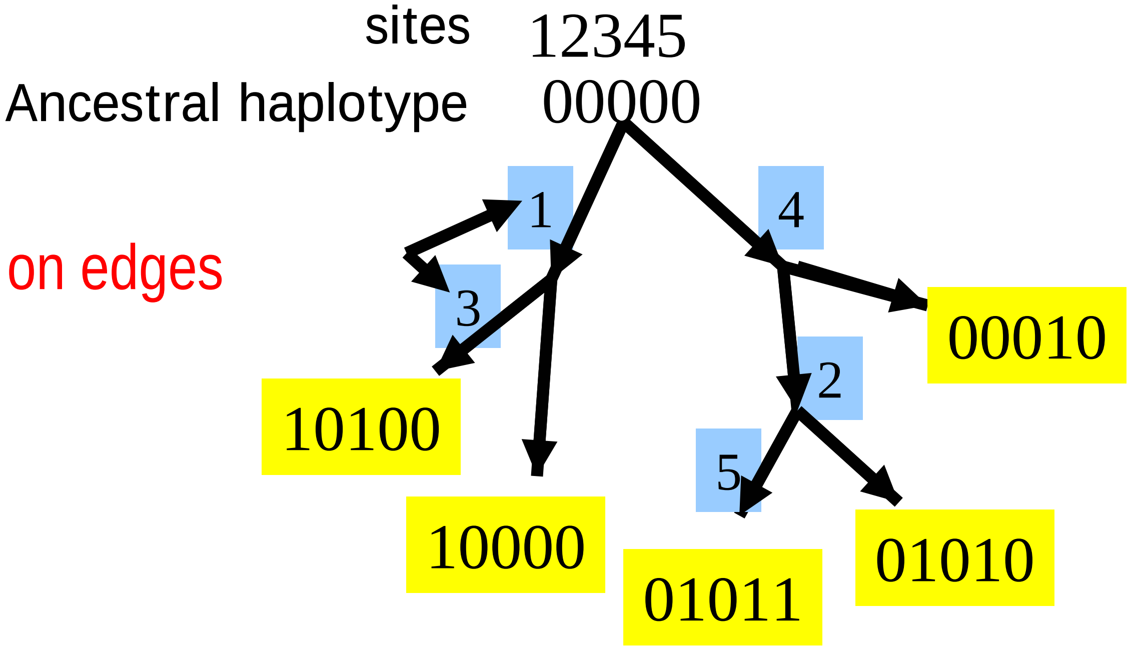
1 1 0 1 0 0 1 0 0

2 1 2 1 0 0 1 2 0

Haplotyping Problem

- Biological Problem:
  - For disease association studies, haplotype data is more valuable than genotype data, but haplotype data is hard to collect. Genotype data is easy to collect.
- Computational Problem:
  - Given a set of n genotypes, determine the original set of n haplotype pairs that generated the n genotypes. This is hopeless without a genetic model.

The Perfect Phylogeny Model of Haplotype Evolution



Site mutations on edges

Extant haplotypes at the leaves

The Perfect Phylogeny Model

- We assume that
  - the evolution of extant haplotypes can be displayed on a rooted, directed tree, with the all-0 haplotype at the root, where each site
  - changes from 0 to 1 on exactly one edge, and each extant haplotype is created by accumulating the changes on a path from the root to a leaf, where that haplotype is displayed.
- In other words, the extant haplotypes evolved along a perfect phylogeny with all-0 root.

Justification for Perfect Phylogeny Model

- Recent strong evidence for long regions of DNA with no recombination.
  - Key to the NIH haplotype mapping project. (See NYT October 30, 2002)
- Mutations are rare at selected sites, so are assumed non-recurrent.
- Connection with coalescent models.

Perfect Phylogeny Haplotype (PPH)

sites

		1	2
S	a	2	2
	b	0	2
	c	1	0

Genotype matrix

- Given a set of genotypes S, find an explaining set of haplotypes that fits a perfect phylogeny. A haplotype pair explains a genotype if the merge of the haplotypes creates the genotype.
  - Example: The merge of 0 1 & 1 0 explains 2 2.
  - Solutions: 0 1, 1 0 & 0 0.
- 3-Gamete Rule:
  - not all four possible values are admissible…(1 1 is missing)

The PPH Problem

	1	2
a	2	2
b	0	2
c	1	0

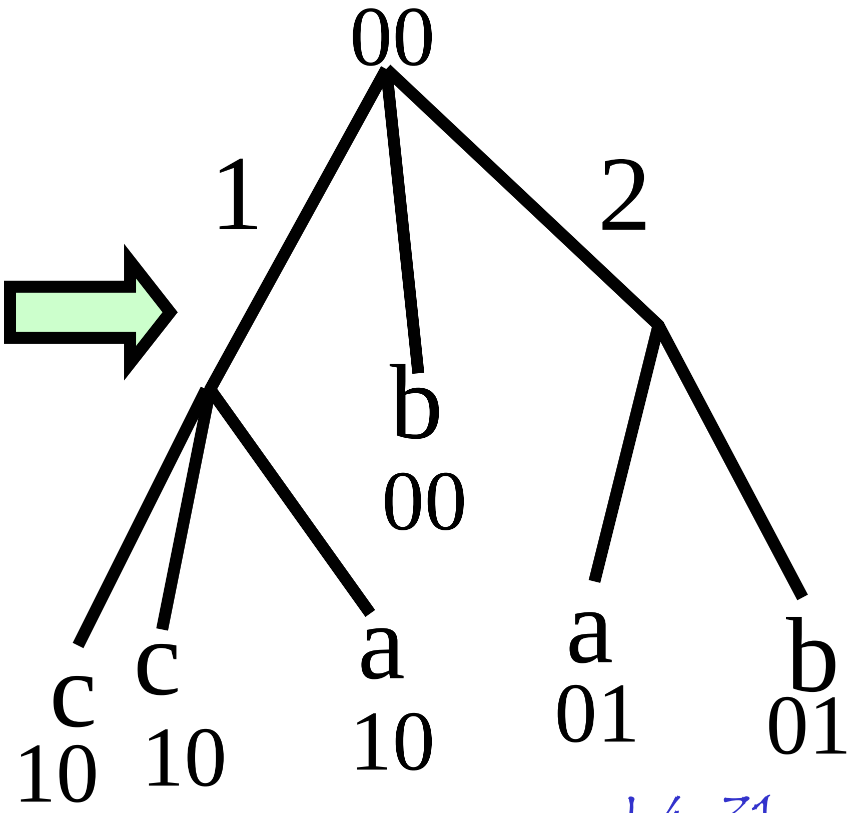
	1	2
a	1	0
a	0	1
b	0	0
b	0	1
c	1	0
c	1	0

- Given a set of genotypes, find an explaining set of haplotypes that fits a perfect phylogeny

The Haplotype Phylogeny Problem

	1	2
a	2	2
b	0	2
c	1	0

	1	2
a	1	0
a	0	1
b	0	0
b	0	1
c	1	0
c	1	0



The Alternative Explanation

	1	2
a	2	2
b	0	2
c	1	0

	1	2
a	1	1
a	0	0
b	0	0
b	0	1
c	1	0
c	1	0

No tree possible for this explanation

Efficient Solutions to the PPH problem - n genotypes, m sites

- Reduction to a graph realization problem (GPPH)
  - Build on Bixby-Wagner or Fushishige solution to graph realization O(nm α(nm)) time.
- Reduction to graph realization –
  - Build on Tutte's graph realization method O(nm^2) time.
  - Specialize the Tutte solution to the PPH problem - O(nm^2) time. Eskin et al.
- Direct, from scratch combinatorial approach
  - O(nm^2) Bafna et al.

Recognizing graphic Matroids

- The graph realization problem
  - It is the same problem as determining if a binary matroid is graphic
  - The fastest algorithm is due to Bixby and Wagner
  - Representation methods due to Cunningham et al.

The DPPH Method

- Bafna et al. $O(nm^2)$ time
  - Based on deeper combinatorial observations about the PPH problem.
- A matrix-centric approach (rather than tree-centric), although a graph is used in the algorithm.
- Key Intuition:
  - We need to understand why some sets of haplotypes have a perfect phylogeny, and some do not.

When does a set of haplotypes fit a perfect phylogeny?

- Classic NASC:
  - Arrange the haplotypes in a matrix, two haplotypes for each individual. Then (with no duplicate columns), the haplotypes fit a unique perfect phylogeny if and only if no two columns contain all three pairs:
- 0,1 and 1,0 and 1,1
- This is the 3-Gamete Test

The Alternative Explanation

	1	2
a	2	2
b	0	2
c	1	0

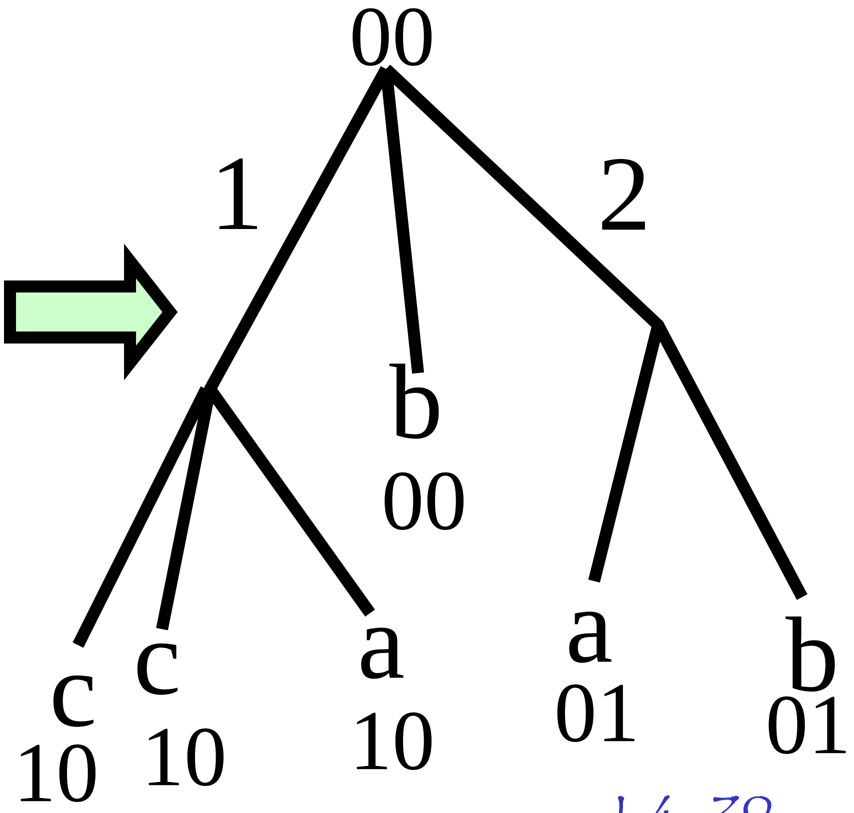
	1	2
a	1	1
a	0	0
b	0	0
b	0	1
c	1	0
c	1	0

No tree possible for this explanation

The Haplotype Phylogeny Problem

	1	2
a	2	2
b	0	2
c	1	0

	1	2
a	1	0
a	0	1
b	0	0
b	0	1
c	1	0
c	1	0



PPH: The Combinatorial Problem

- Input:
  - A ternary matrix (0,1,2) M with 2N rows partitioned into N pairs of rows, where the two rows in each pair are identical.

- Def:
  - If a pair of rows (r,r') in the partition have entry values of 2 in a column j then positions (r,j) and (r',j) are called **Mates**.

PPH: The Combinatorial Problem

- Output:
  - A binary matrix M′ created from M by replacing each 2 in M with either 0 or 1, such that

- A position is assigned 0 iff its Mate is assigned 1.
- M′ passes the 3-Gamete Test, i.e., does not contain a 3x2 submatrix (after row and column permutations) with all three combinations 0,1; 1,0; and 1,1

Initial Observations

- If two columns of M contain the following rows

2 0

2 0 mates

0 2

0 2 mates

- then M′ will contain a row with 1 0 and a row with 0 1 in those columns.

◇ This is a forced expansion.

Initial Observations

- Similarly, if two columns of M contain the mates

2 1

2 1

- then M′ will contain a row with 1 1 in those columns.

◇ This is a forced expansion.

Further Observations

- If a forced expansion of two columns creates

0 1 in those columns, then any 2 2
1 0 2 2

in those columns must be set to be

0 1
1 0

- We say that two columns are forced out-of-phase.

Further Observations

- If a forced expansion of two columns creates 1 1 in those columns, then any

2 2 in those columns must be set to be 1 1

2 2 0 0

- We say that two columns are forced in-phase.

Example

	1	2	3
a	1	2	2
a	1	2	2
b	2	0	2
b	2	0	2
c	1	2	2
c	1	2	2
d	1	2	2
d	1	2	2
e	2	2	0
e	2	2	0

- Columns 1 and 2, and 1 and 3 are forced in-phase.
- Columns 2 and 3 are forced out-of-phase.

Immediate Failure

- It can happen that the forced expansion of cells creates a 3x2 submatrix that fails the 3-Gamete Test. In that case, there is no PPH solution for M.

Example:

20
20
11
11
02
02

Will fail the 3-Gamete Test

An $O(nm^2)$-time Algorithm

- Find all the forced phase relationships by considering columns in pairs.
- Find all the inferred, invariant, phase relationships.
- Find a set of column pairs whose phase relationship can be arbitrarily set, so that all the remaining phase relationships can be inferred.

- Result:
  - An implicit representation of all solutions to the PPH problem.

An Example

	1	2	3	4	5	6	7
a	1	2	2	2	0	0	0
a	1	2	2	2	0	0	0
b	2	0	2	0	0	0	2
b	2	0	2	0	0	0	2
c	1	2	2	2	0	2	0
c	1	2	2	2	0	2	0
d	1	2	2	0	2	0	0
d	1	2	2	0	2	0	0
e	2	2	0	0	0	2	0
e	2	2	0	0	0	2	0

Overview of Bafna et al. algorithm

- First, represent the forced phase relationships, and the needed decisions, in a graph G.
  - Each node represents a column in M, and each edge indicates that the pair of columns has a row with 2's in both columns.
  - The algorithm builds this graph, and then checks whether any pair of nodes is forced in or out of phase.

PPH

- Color the edges to create Gc:
    - Each Red edge indicates that the columns are forced in-phase.
    - Each Blue edge indicates that the columns are forced out-of-phase.
- Let Gf be the subgraph of Gc defined by the red and blue edges.

PPH

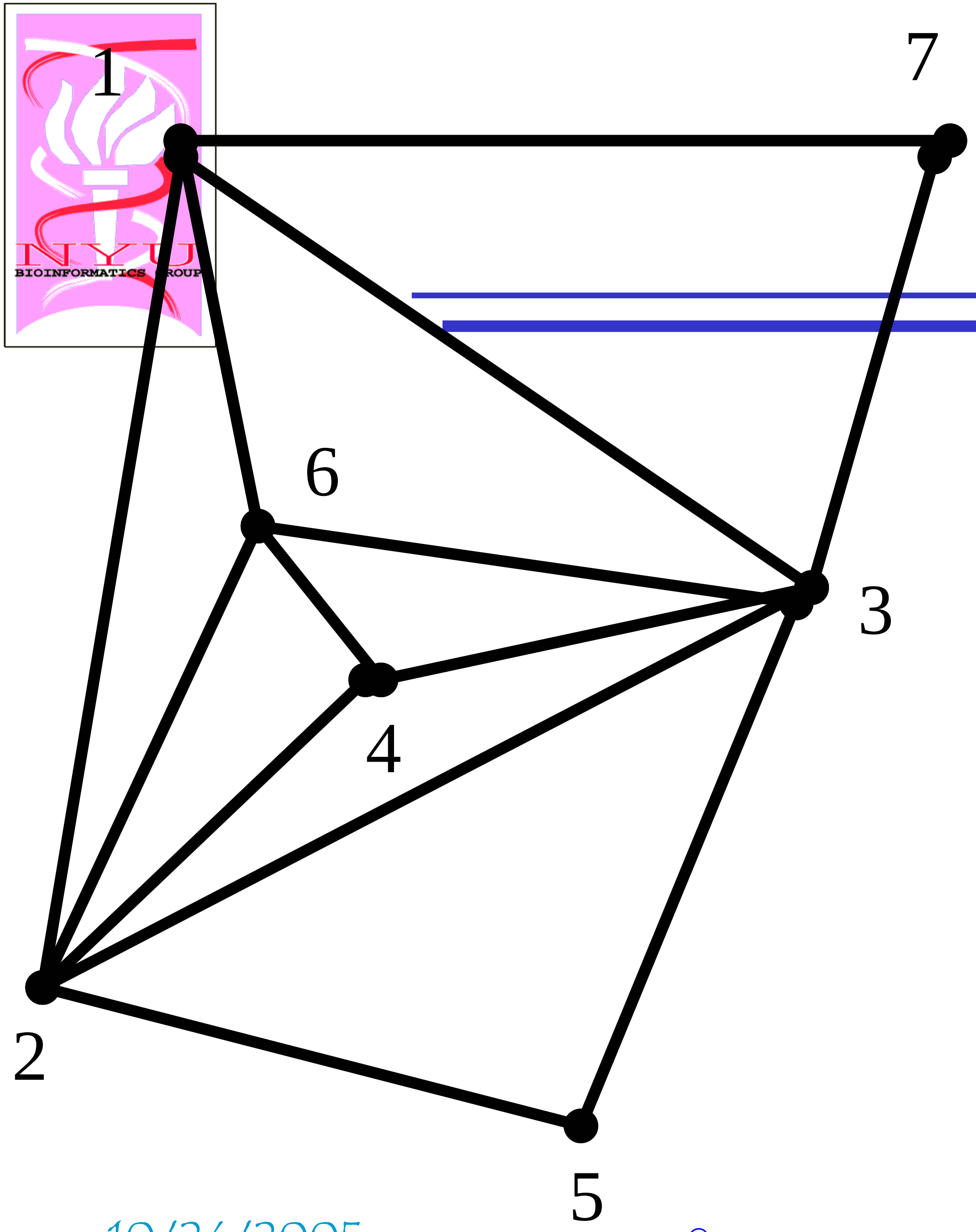
- There is a solution to the PPH problem for M if and only if there is a coloring of the dashed edges of G_c with the following property:

- For any triangle (i,j,k) in G_c, where there is one row containing 2's in all three columns i,j and k (any triangle containing at least one dashed edge will be of this type), the coloring makes either 0 or 2 of the edges blue (out-of-phase).

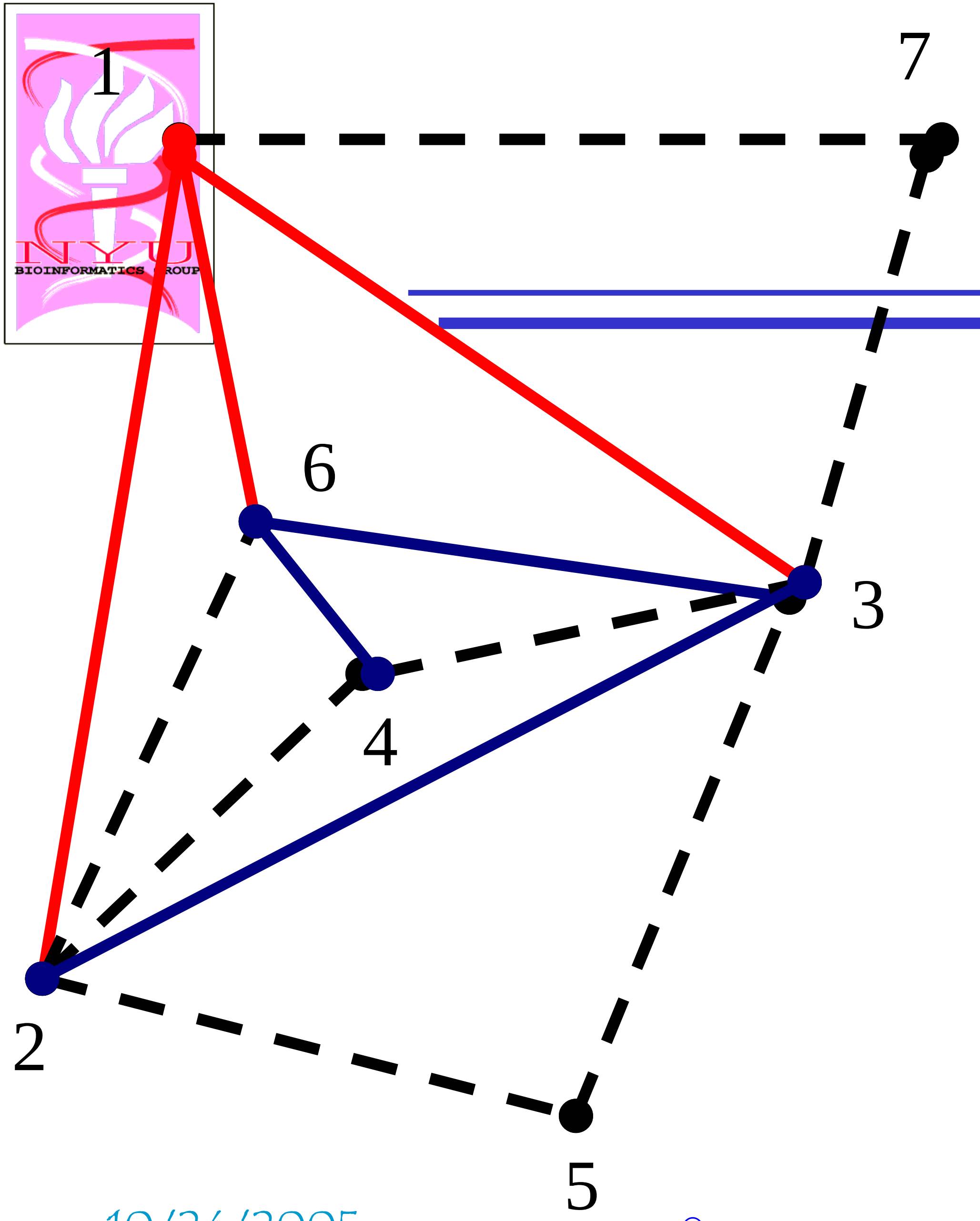
Triangle Rule

- Theorem 1:
  - If there are any dashed edges whose ends are in the same connected component of G_f, at least one edge is in a triangle where the other edges are not dashed, and in every PPH solution, it must be colored so that the triangle has an even number of Blue (out of Phase) edges.
- This is an "inferred" coloring.

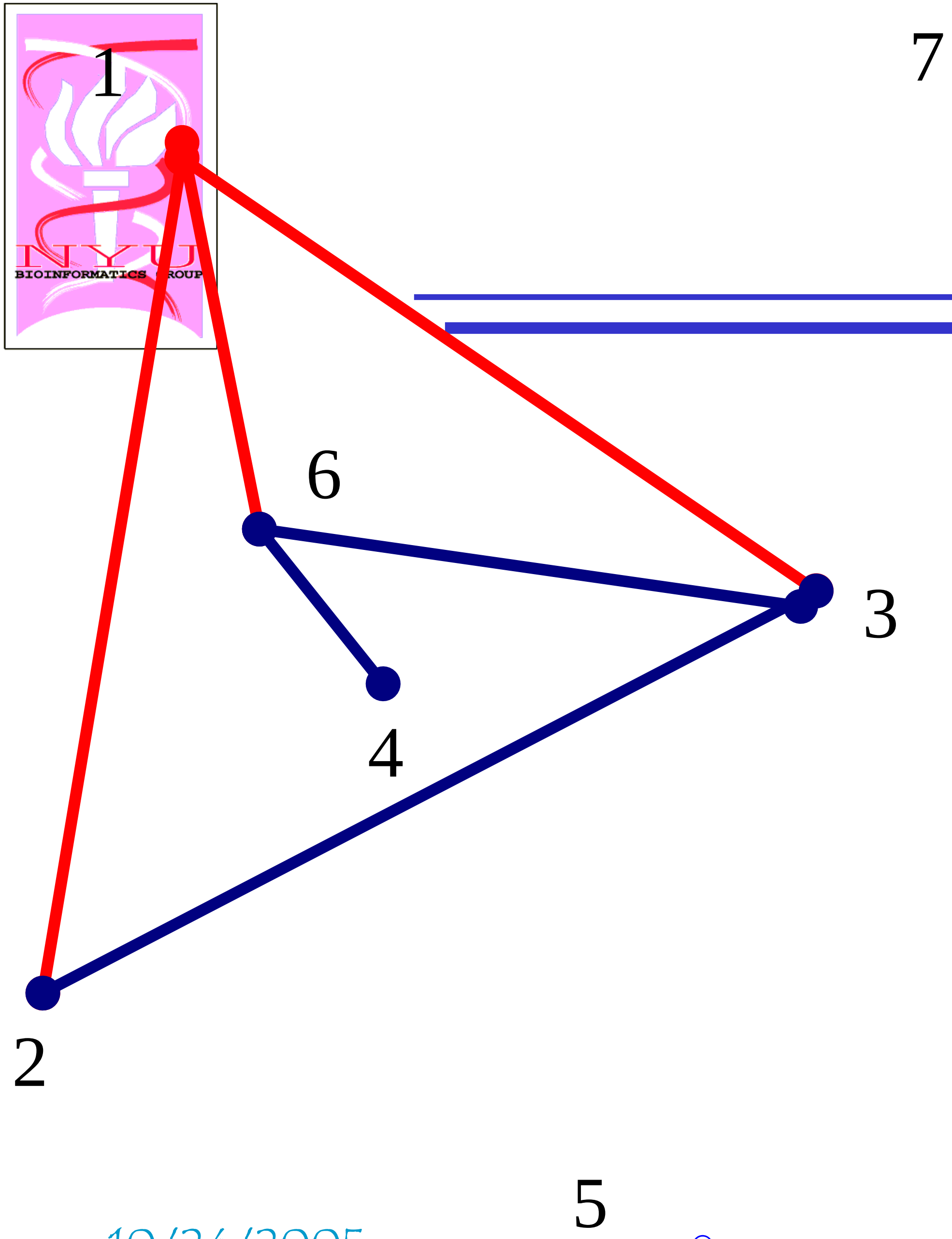
Graph G



Graph G_c

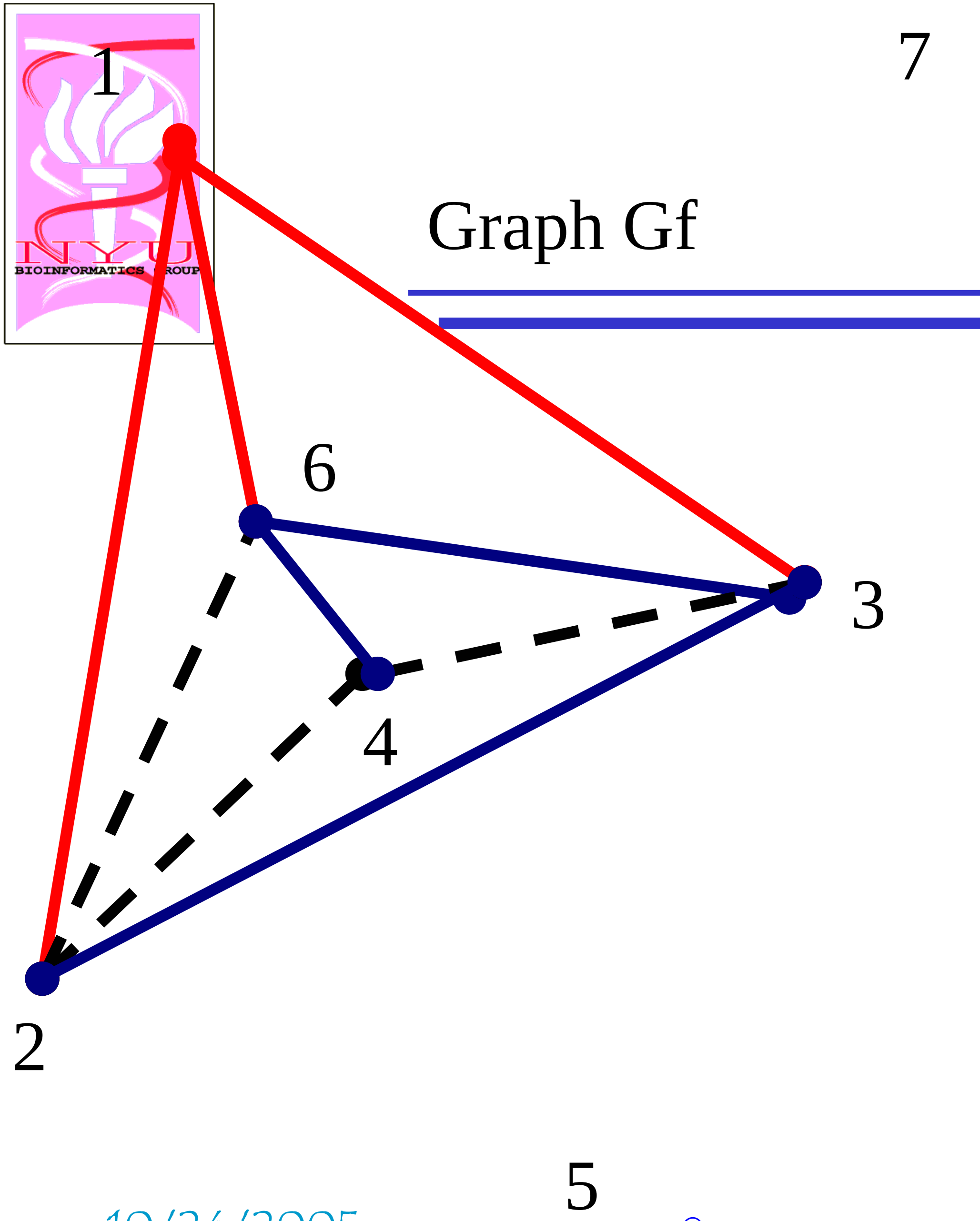


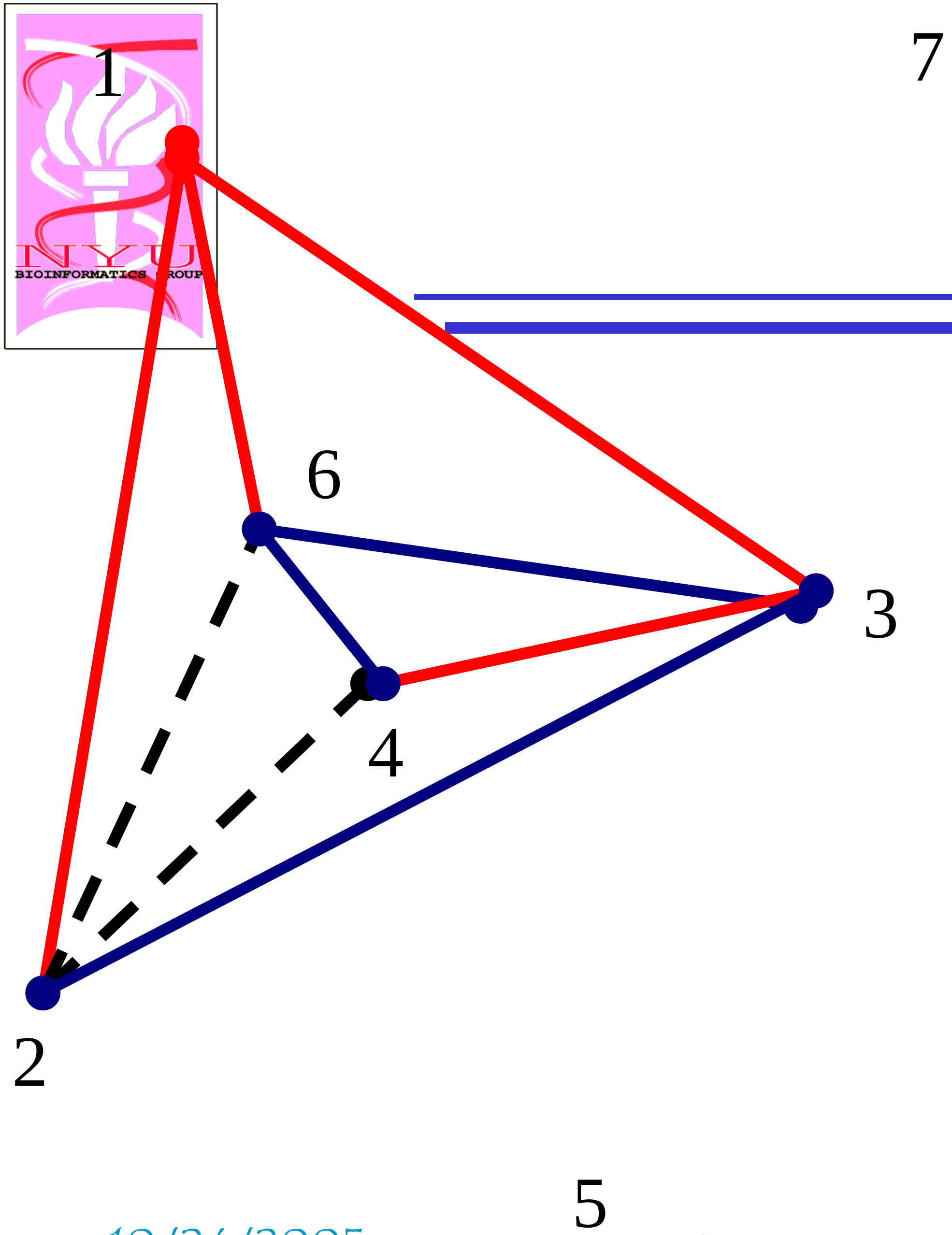
Graph G_f has three connected components.

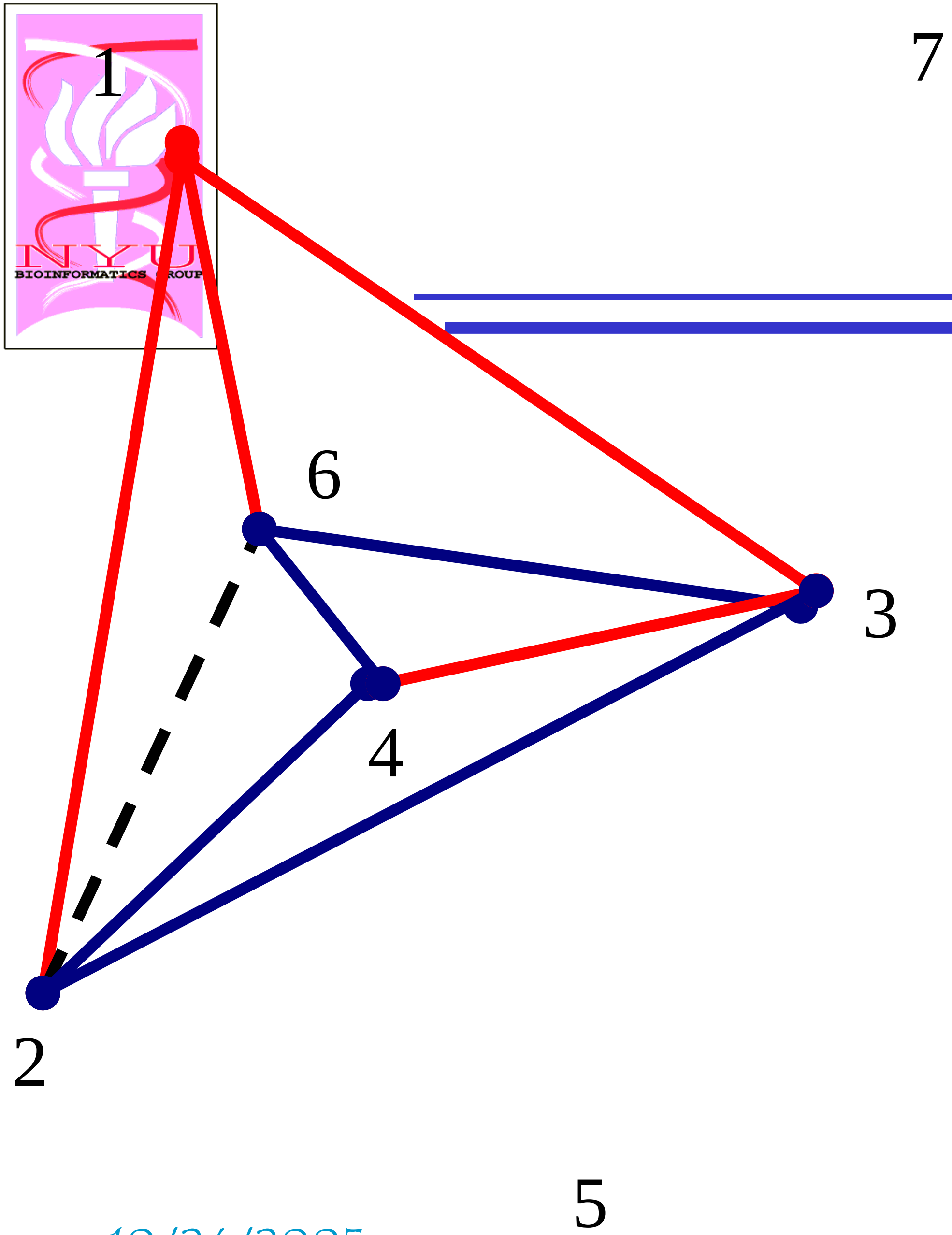


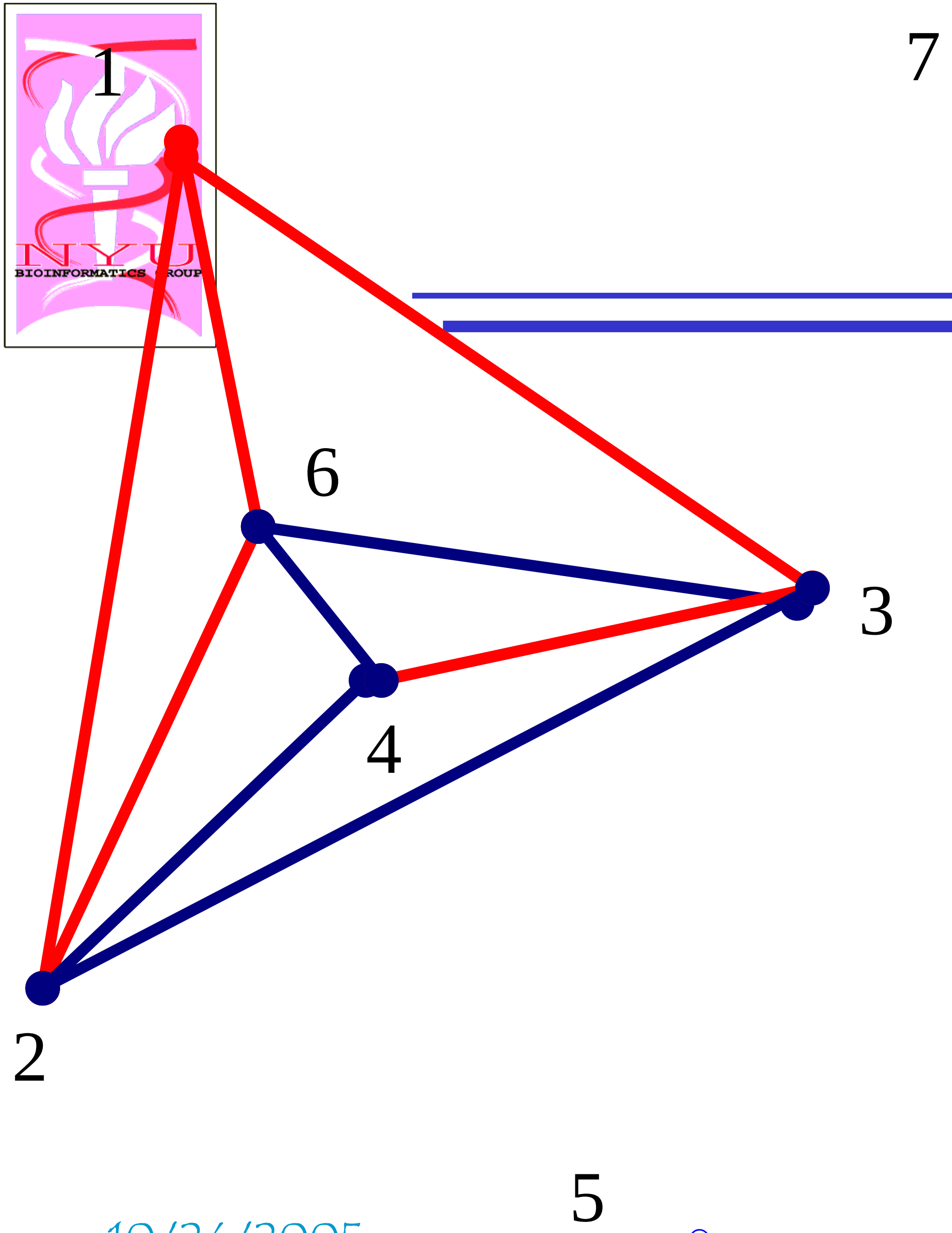
Triangle Rule

Graph Gf




7
1
6
3
4
2
5


7
1
6
3
4
2
5


7
1
6
3
4
2
5

Statistical Methods

- Maximum Likelihood Estimation
- Bayesian Estimation

Maximum Likelihood Estimation 1995

- Excoffier and Slatkin 1995
  - Their method tries to estimate the haplotype frequencies by maximizing the likelihood of the data.
  - They do this using the EM algorithm. Intuitively, you start with some initial haplotype frequencies guess, and then by an iterative method you update these haplotype frequencies until convergence is attained.

MLE

- In the E-step you compute for each genotype the probability of resolving it into each possible haplotype pair: $P(h_1, h_2 | g)$, where h_1, h_2 are two haplotypes and g is a genotype.
- In the M-step you update the haplotype frequencies using the estimates obtained in the E-step. (similar to gene counting)

$$P_h = (1/2n)\ \Sigma_{j=1}^{m}\ n_j\ \Sigma_{l=1}^{c_j}\ \delta_{ih}\ P(h_{i1}, h_{i2} | g_j)$$

- Where n_j is the number of genotypes of type j, c_j is the number of possible haplotype explanations for genotype g_j (exponential in the number of heterozygous sites) and δ_{ih} is an indicator equal to the number of times haplotype h is present in the pair h_{i1}, h_{i2}

MLE

- This algorithm has been shown to be accurate, especially in large sample sizes. The result is an estimation of the haplotype frequencies. From these one can reconstruct the haplotype themselves by taking the most probable assignment.
- The main drawback of this algorithm is that it is exponential in the number of heterozygous loci. Consequently, the maximum number of loci it can handle is around 15.

Bayesian Estimation

- The Bayesian methods
  - They treat the unknown haplotypes as random quantities from an unknown distribution that they try to estimate using the known genotype data.
- There are two ingredients in each Bayesian algorithm:
  - Prior beliefs about the haplotypes in the population
  - The Computational part

Bayesian Estimation

- Posteriori
  - What you really want is the most probable a posteriori solution given the genotype data. Unfortunately the posterior distribution cannot be calculated exactly and one has to apply MCMC methods to obtain samples from this distribution.
- The choice of prior or computational algorithm
  - affect the estimation process and the existing algorithms differ in either one or both components.

Bayesian Estimation

- Stephens et al.
  - Two Bayesian algorithms were proposed by Stephens et al. Both use a Gibbs sampler, but different priors.
  - The Gibbs sampler is an MCMC algorithm that constructs a MC whose stationary distribution is P(H|G).

Bayesian Estimation

- It starts with an initial guess of haplotypes H^0 and then repeatedly chooses an individual at random from the ambiguous individuals and estimates its haplotypes given the haplotypes of the other individuals:
  - Sample (h_{i1}, h_{i2}) from $P((h_1, h_2) \mid G, H_{-i})$ where H_{-i} are the estimated haplotypes for the other individuals.
  - Repeat this process until convergence.
- These conditional distributions are influenced by the priors assumed. The first one assumes a Dirichlet prior on the haplotype frequencies, while the second one assumes a better prior based on the coalescent

Bayesian Estimation

- The Bayesian methods
  - .. are very promising for this challenging problem because of their ability to provide accurate solutions, to incorporate prior information, missing genotype data, and genotyping error.
  - Another good feature of all statistical methods is that it gives an estimation of the uncertainty in the estimation and hence for those individuals for which the algorithms are not that sure, subsequent molecular techniques can further be used to find the haplotypes.

Bayesian Estimation

- Blockiness:
  - Designing statistical methods that take into account the blocky structure of the genome.
- Time efficiency is important,
  - … but only secondary to the other issues. After all it takes such a long time just to gather the data and do the genotyping experiments, and so if one can predict the haplotypes accurately in a reasonable time, this is what is important.

Genetic Diseases

Genetic Diseases

- Classified into three types:
  - Single Gene disorder
    - Mutations in autosomes, sex chromosomes or mitochondrial DNA
  - Chromosomal Abnormalities
    - Excess, Deficiency or Translocation of part or all of a chromosome
  - Polygenic Diseases
- Pedigree patterns
  - Dominant or Recessive

Polygenic Disorders

- Do not show characteristic pedigree patterns
- Severity is influenced by lifestyle factors
  - They often were not recognized as genetic diseases.
- They are result of small variations in a number of genes.
  - Not due to a single mutation in any single dominant gene.
  - Together they predispose an individual to a serious defect.
- Affect about 5% in children & > 60% in the total population.

Treatment

- Therapeutic intervention for patients with single gene defect:
  - Therapeutic proteins
  - Antisense technology
  - Gene therapy
  - Gene repair
  - RNAi
- Personalized Medicine
  - Dosage and choice of medication, based on genome-wide SNP profile.

Single Gene Disorder

Genetic Change	Example
Point Mutation	
Missense mutation (Substituted Protein)	Sickle Cell Anemia (β globin gene)
Nonsense mutation (Premature Stop Codon)	β^{0}-Thalassemia (β globin gene)
RNA processing mutation (Abnormal Splicing)	β-Thalassemia (RNA splicing mutant)
Regulatory Mutation (Affect TFs)	Hereditary Persistenc of Fetal hemoglobin (Y-globin gene promoter mutation)
INDEL Mutation	
3-base Mutation (no frameshift)	Cystic Fibrosis (removes phenylalanine residue)
Small indel causing frameshift	Tay-Sachs Disease (4 base insertio in hexosaminidase A gene)
Line or Alu insertion	
Expansion of trinucleotide repeat	Huntington disease

Bioinformatics Databases of Interest

Bioinformatics DataSources

- Database interfaces
  - Genbank/EMBL/DDBJ, Medline, SwissProt, PDB, …
- Sequence alignment
  - BLAST, FASTA
- Multiple sequence alignment
  - Clustal, MultAlin, DiAlign
- Gene finding
  - Genscan, GenomeScan, GeneMark, GRAIL
- Protein Domain analysis and identification
  - pfam, BLOCKS, ProDom,
- Pattern Identification/
- Characterization
  - Gibbs Sampler, AlignACE, MEME
- Protein Folding prediction
  - PredictProtein, SwissModeler

Five Important Websites

- NCBI (The National Center for Biotechnology Information;
  - http://www.ncbi.nlm.nih.gov/
- EBI (The European Bioinformatics Institute)
  - http://www.ebi.ac.uk/
- The Canadian Bioinformatics Resource
  - http://www.cbr.nrc.ca/
- SwissProt/ExPASy (Swiss Bioinformatics Resource)
  - http://expasy.cbr.nrc.ca/sprot/
- PDB (The Protein Databank)
  - http://www.rcsb.org/PDB/

NCBI (http://www.ncbi.nlm.nih.gov/)

- Entrez interface to databases
  - Medline/OMIM
  - Genbank/Genpept/Structures
- BLAST server(s)
  - Five-plus flavors of blast
- Draft Human Genome
- Much, much more…

EBI (http://www.ebi.ac.uk/)

- SRS database interface
  - EMBL, SwissProt, and many more
- Many server-based tools
  - ClustalW, DALI, …

SwissProt (http://expasy.cbr.nrc.ca/sprot/)

- Curation…
  - Error rate in the information is greatly reduced in comparison to most other databases.
- Extensive cross-linking to other data sources
- SwissProt is the 'gold-standard' by which other databases can be measured, and is the best place to start if you have a specific protein to investigate

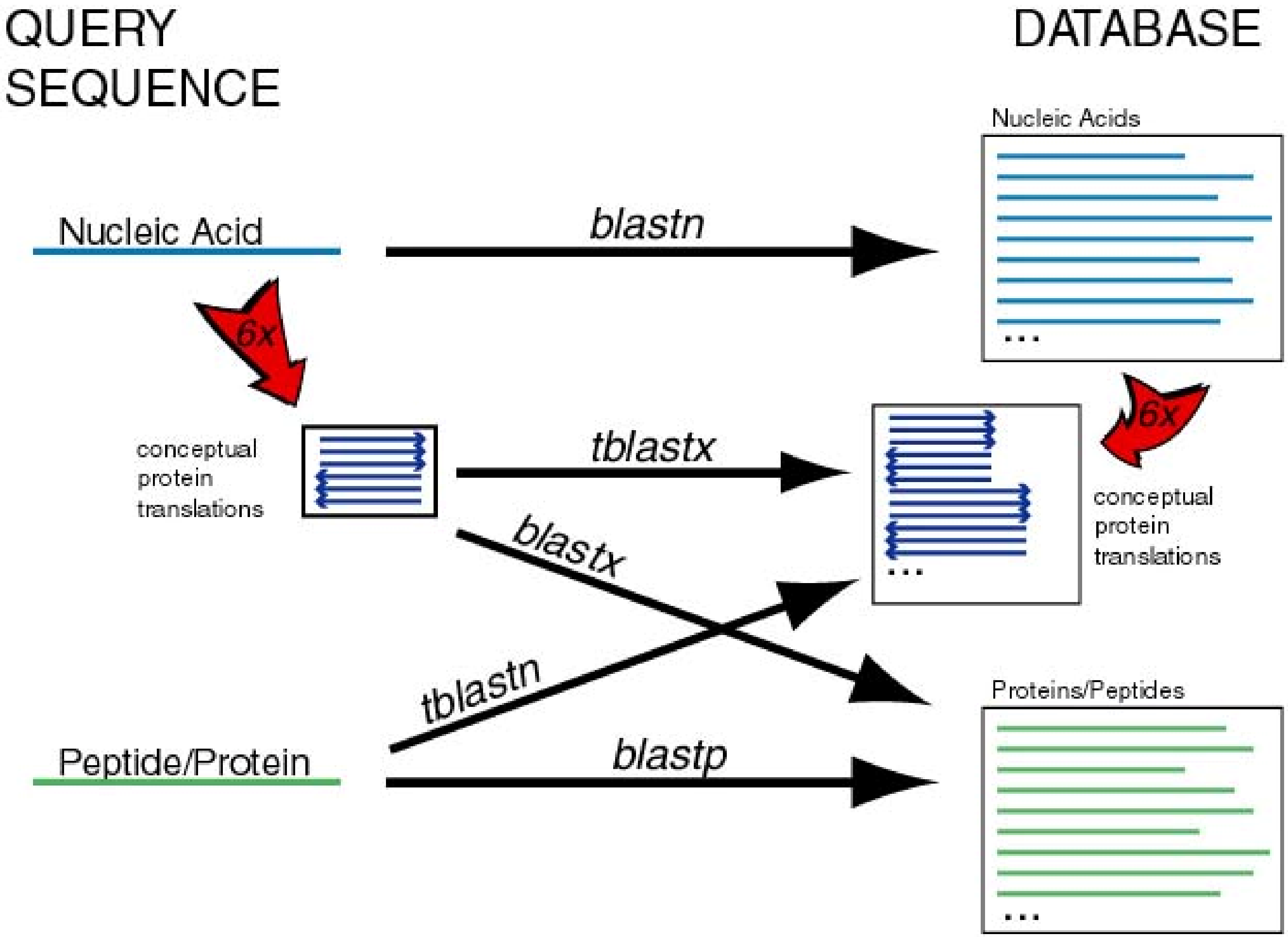
A few more resources

- Human Genome Working Draft
  - http://genome.ucsc.edu/
- TIGR (The Institute for Genomics Research)
  - http://www.tigr.org/
- Celera
  - http://www.celera.com/
- (Model) Organism specific information:
  - Yeast: http://genome-www.stanford.edu/Saccharomyces/
  - Arabidopis: http://www.tair.org/
  - Mouse: http://www.jax.org/
  - Fruitfly: http://www.fruitfly.org/
  - Nematode: http://www.wormbase.org/
- Nucleic Acids Research Database Issue
  - http://nar.oupjournals.org/

Example 1:

- Searching a new genome for a specific protein
- Specific problem:
  - We want to find the closest match in *C. elegans* of *D. melanogaster* protein NTF1, a transcription factor
- First- understanding the different forms of blast

The different versions of BLAST



Some possible methods

- If the domain is a known domain:
- SwissProt
  - text search capabilities
  - good annotation of known domains
  - crosslinks to other databases (domains)
- Databases of known domains:
  - BLOCKS (http://blocks.fhcrc.org/)
  - Pfam (http://pfam.wustl.edu/)
  - Others (ProDom, ProSite, DOMO,...)

Nature of conservation in a domain

- For new domains, multiple alignment is your best option
  - Global: clustalw
  - Local: DiAlign
  - Hidden Markov Model: HMMER
- For known domains, this work has largely been done for you
  - BLOCKS
  - Pfam

Protein Tools

- Search/Analysis tools
  - Pfam
  - BLOCKS
  - PredictProtein (http://cubic.bioc.columbia.edu/predictprotein/predictprotein.html)

Different representations of conserved domains

- BLOCKS
  - Gapless regions
  - Often multiple blocks for one domain
- PFAM
  - Statistical model, based on HMM
  - Since gaps are allowed, most domains have only one pfam model

To be continued…

…