DIMACS Technical Report 97-63
October 1997

A Short Course in Computational Molecular Biology[1]

by

D. Durand[2]
Computational Biology Group, University of Pennsylvania
DIMACS, Rutgers University
durandd@dimacs.rutgers.edu, http://www.cs.princeton.edu/~durand

M. Farach[3]
Department of Computer Science
Rutgers University
farach@cs.rutgers.edu, http://www.cs.rutgers.edu/~farach

R. Ravi[4]
Graduate School of Industrial Administration
Carnegie Mellon University
ravi@cmu.edu

M. Singh[5]
Department of Computer Science
Princeton University
mona@cs.princeton.edu

[1]Presented at the First International University of Buenos Aires/Dimacs Tutorial in Bioinformatics
[2]Permanent Member, Supported by NSF Grants BIR-94-13215 A01 and BIR-94-12594.
[3]Permanent Member, Supported by NSF Career Development Award CCR-95-01942, NSF Grant BIR-94-12594, an Alfred P. Sloan Research Fellowship and NATO Grant 96-0215.
[4]Permanent Member, Supported by NSF Grant BIR-94-12594 and NSF CAREER grant 96-25297.
[5]Permanent Member, Supported by NSF Grant BIR-94-12594.

DIMACS is a partnership of Rutgers University, Princeton University, AT&T Research, Bellcore, and Bell Laboratories.
DIMACS is an NSF Science and Technology Center, funded under contract STC–91–19999; and also receives support from the New Jersey Commission on Science and Technology.

ABSTRACT

The advent of recombinant DNA technology during the 1970s has led to an inundation of biological sequence data. The compilation and analysis of DNA and protein sequences is now a fundamental task in molecular biology requiring. Computational Molecular Biology is the field of computer science that has emerged to solve algorithmic problems in determining sequences and analyzing them. Specific research efforts in this area include sequencing and mapping, pairwise and multiple sequence comparison, protein structure determination and evolutionary tree reconstruction. Solutions to these problems contribute both to basic scientific research and product development in the biotechnology industry. We have designed a course to give a basic introduction to the major algorithmic research areas in computational biology.

Overview

1. General Biology *(3 hours)*
   (a) Biological sequences: DNA, RNA and proteins.
   (b) Mutations
   (c) Gene and genome structure
   (d) Introduction to alignments: what and why?
2. Sequence Analysis *(4.5 hours)*
   (a) Dynamic Programming: global and local pairwise alignment, gap penalty functions.
   (b) Pairwise alignment revisited: log-odds statistics, substitution matrices. Database searching: BLAST, FASTA.
   (c) Multiple sequence alignment.
3. Sequencing and Mapping *(3 hours)*
   (a) Recombinant DNA technology.
   (b) Sequence assembly.
   (c) Physical mapping.
4. Protein Structure *(4.5 hours)*
   (a) Introduction to structural classification.
   (b) Tertiary protein structure prediction.
   (c) Prediction of secondary structure.
   (d) Motif recognition: statistical and computational learning methods.
   (e) Protein folding and lattice models.
5. Evolutionary Trees*(3 hours)*
   (a) Molecular evolution: paralogy, gene trees, mutational models.
   (b) Multiple sequence alignment and tree reconstruction.
   (c) Phylogeny construction: maximum likelihood estimation and distance methods.

1 General Biology

Genetic material encodes the information that determines the function, development and differentiation of cells, and, hence, the appearance of the organism. This information is stored in DNA molecules and expressed through the formation of proteins. Cell development and differentiation is controlled through gene regulation, which determines when and how much of a protein is made.

1. Introduction
   (a) What is computational molecular biology?
   (b) What will we cover today?
2. Genes and Protein Synthesis in Bacteria (Procaryotes)
   (a) Chromosomes are the DNA molecules on which genetic information is stored. A gene is a subsequence of a chromosome that encodes a single protein.
   (b) DNA
      i. DNA is a polymer of four nucleotides (adenine, cytosine, guanine and thymine) and can be viewed as a string over a four letter alphabet (A, C, G, T).
      ii. Nucleotides are composed of a sugar, a phosphate and a basic group. The base determines the identity of the nucleotide.
      iii. DNA structure: double stranded, helical structure; base pairing (AT, GC bonds); orientation (3' – 5').
   (c) DNA replication.
   (d) Protein Synthesis
      i. Proteins are amino acid polymers. There are twenty amino acids, each composed of a carbon backbone and a residue that determines its identity and its chemical properties.
      ii. Protein synthesis is a two step process mediated by RNA. RNA is a single stranded nucleic acid. It differs from DNA in that its nucleotides contain a different sugar and the nucleotide, thiamine, is replaced with the nucleotide, uracil.
      iii. First, DNA is *transcribed* into messenger RNA (also called mRNA). Regulatory sequences (promoters, repressors) on the chromosome determine when genes are transcribed.
      iv. Second, mRNA is *translated* into the amino acid sequence it encodes, aided by tRNA molecules and ribosomes (RNA and protein complexes). As it is synthesized, the protein takes on its three-dimensional structure.
3. Genes and Protein Synthesis in Eucaryotes ("higher" organisms)

   (a) Genetic organization in eucaryotes: the nucleus, linear chromosomes, diploidy and polyploidy, nucleosomes.
   (b) DNA replication revisited: Recombination and meiosis.
   (c) Gene structure in eucaryotes: introns, exons and gene splicing.
   (d) More complex gene regulation: inducers, enhancers and transcription factors.

4. Gene and chromosome mutability
   (a) Point mutations
      i. insertions and deletions.
      ii. substitutions: transitions versus transversions; silent, neutral, nonsense and missense mutations; reverse mutation.
   (b) Genome rearrangements: duplication, deletion, inversion and translocation.
   (c) Gene families.

5. Conclusion
   (a) Summary of today's lecture.
   (b) What will we cover in this course?
      i. Introduction to sequence alignment.
      ii. An overview of problems in computational biology.

References

[1] Bruce Alberts et al. *Molecular Biology of the Cell* Garland, 1994

[2] Larry Gonick and Mark Wheelis. *The Cartoon Guide to Genetics* HarperPerennial, 1991

[3] James L. Gould and William T. Keeton *Biological Science* W. W. Norton and Co., 1996

[4] A. J. F. Griffiths, J. H. Miller, D. T. Suzuki, R. C. Lewontin and W. M. Gelbart *An Introduction to Genetic Analysis* Freeman, 1996

[5] R. C. King and W. D. Stansfield *A Dictionary of Genetics* Oxford University Press, 1990

[6] Benjamin Lewin *Genes VI* Oxford University Press, 1997

[7] Wen-Hsiung Li and Dan Graur *Fundamentals of Molecular Evolution* Sinauer Associates, 1991

2 Sequence Analysis

1. Dynamic Programming and Alignments

   (a) Definitions

      i. Edit operations: Insertions, Deletions, Substitutions
      ii. Edit distance
      iii. Global Alignment
      iv. Local Alignment

   (b) Edit operations come in a canonical ordering

   (c) Therefore, we can compute

$$Sim(A[i], B[j]) = \max \begin{cases} Sim(A[i-1], B[j-1]) + S(A[i], B[j]) \\ Sim(A[i-1], B[j]) + I(A[i]) \\ Sim(A[i], B[j-1]) + I(B[j]) \end{cases}$$

   (d) Initial conditions change behaviour:
   As an exercise, what do we do if we want deletion of prefixes to be free?

$$Sim(A[0], B[i]) = Sim(A[j], B[0]) = 0$$

   (e) How do we compute Local Alignment?

      i. Deleting prefixes is free.
      ii. Deleting suffixes is free: we are looking for maximum in the entire matrix, not just $Sim(A[n], B[m])$.
      iii. How do we allow prefixes of *both* strings to be deleted?

$$LSim(A[i], B[j]) = \max \begin{cases} LSim(A[i-1], B[j-1]) + S(A[i], B[j]) \\ LSim(A[i-1], B[j]) + I(A[i]) \\ LSim(A[i], B[j-1]) + I(B[j]) \\ 0 \end{cases}$$

      Note: the last case kicks in when prefixes are badly alignment and must be deleted.

   (f) Gaps: what if k deletions in a row cost $f(k)$, rather than $kf(1)$, that is, a gap of length k is not simply the same as k individual single character gaps?

      i. Affine gap functions
      ii. Convex/concave gap functions

   (g) Alignment in linear space.

2. Sequence Analysis: Statistics and Programs

(a) Searching a single sequence for maximal scoring segments

  i. Random models of arbitrary genetic sequences and target genetic sequences. What assumptions are needed to study the statistics of the maximal scoring segments?
  ii. Given a sequence and a scoring vector with a score for each character, what is the statistical significance of its maximal segment score?
  iii. What is the distribution of characters in segments with very high scores?

(b) Comparison of two sequences for maximal segment pairs

  i. Random models with occurrence frequencies, and target frequencies for aligned pairs (with no gaps). What assumptions are needed to study the statistics of the maximal segment pair (the pair of segments from the two sequences whose local gapless alignment has the maximum similarity score over all such pairs)?
  ii. Given a pair of sequences and a scoring matrix with a pairwise alignment score for every pair of characters, what is the statistical significance of the alignment score of the maximal segment pair?
  iii. What is the distribution of the aligned pairs of characters in segment pairs with very high scores?
  iv. Reasoning backwards, given a target distribution of aligned pairs of characters, how can we design a scoring matrix to best pick out such alignments as its maximal segment pairs?

(c) Database search tools: BLAST and FAST

  i. What are they?
  ii. Where are they? (Check out, e.g., `http://www.ncbi.nlm.nih.gov/BLAST` and `http://swarmer.stanford.edu/cgi-bin/fastaq-form?options=simple`).
  iii. How do they work?
  iv. How can one interpret their results?

References

[1] "Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes," S. Karlin and S. Altschul, *Proc. of the Natl. Acad. Sci. USA*, Vol. 87, pp. 2264-2268 (1990). This reference contains a description (without derivation) of the results on the statistics of maximal segment scores for single sequences and of maximal segment pairs for pairs of sequences, in terms of the scoring matrices used.

[2] Chapter 3.5, Introduction to Computational Molecular Biology, J. Setubal and J. Meidanis, PWS Publishing Company, 1997. This chapter contains an intuitive derivation of the PAM matrix scores, along with brief descriptions of the design of the FAST and BLAST programs for genetic database search.

[3] "Improved tools for biological sequence comparison," W. R. Pearson and D. J. Lipman, *Proc. Natl. Acad. Sci. USA*, Vol. 85, pp. 2444-24448 (1988). This paper contains a description of the FAST suite of programs for local similarity searches of genetic databases with a query string.

[4] "Basic Local Alignment Search Tool," S. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, *J. Mol. Biol.*, 215, 403-410 (1990). This is the main article describing BLAST, a popular program for genetic database searching for local alignment matches with a query sequence. It also contains some justification for the parameter setting used as default by the program.

[5] "Amino acid substitution matrices from an information theoretic perspective," S. Altschul, *J. Mol. Biol.*, 219, 555-565 (1991). This paper contains an interpretation of local alignment scores in units of bits of information by examing substitution matrices in terms of their entropy. From this perspective, the paper prescribes typical lengths of significant local alignments for an average search using a particular PAM matrix for a requisite level of significance.

[6] "Sequence alignment and penalty choice," M. Vingron and M. S. Waterman, *J. Mol. Biol.*, 235, 1-12 (1994). This review article contains two parts: The first part describes a parametric approach to describing optimal alignments for all possible settings of gap penalties. The second part describes a probabilistic phase transition in the behavior of optimal alignments as an expected score measure associated with the scoring matrix is increased. This threshold identifies a boundary between local and global alignments and thus helps in identifying favorable scoring schemes for these two distinct types of alignments.

3. Multiple Sequence Alignment

   (a) An introduction to Multiple Sequence Alignment (MSA)

      i. Intuitive notions of MSA as an extension of pairwise alignment.

      ii. Global versus local MSA

   (b) Applications of MSA

      i. Characterizing conserved patterns.

      ii. Phylogeny reconstruction

      iii. Structure prediction.

   (c) Global Alignment

      i. A formal definition of MSA as an optimization problem.

   ii. Scoring functions for MSA: Sum-of-Pairs (SP), tree alignment (TA), star alignment
   iii. Complexity results: hardness of global MSA

(d) Exact methods for constructing global MSA's.
   i. Sum-of-pairs using dynamic programming
   ii. Tree alignment.
   iii. Improving performance by pruning the search space.

(e) Approximation algorithms for global MSA

(f) Biological measures of MSA quality
   i. Using structural information to construct or validate alignments.
   ii. Experimental comparisons of MSA algorithms.

(g) A sampling of heuristic methods.

References

[1] S.C. Chan, A.K.C. Wong and D.K.Y. Chiu. "A Survey of Multiple Sequence Comparison Methods" Bulletin of Mathematical Biology (1992) 54:563–598

[2] Adam Godzik. "The structural alignment between two proteins: Is there a unique answer?" Protein Science (1996) 5:1325–1338

[3] M.A. McClure, T.K. Vasi and W.M. Fitch. "Comparative Analysis of Multiple Protein-Sequence Alignment Methods" Mol. Biol. Evol. (1994) 11:571 – 592

3 Sequencing and Mapping

1. Recombinant DNA Technology

   (a) Cut, paste and copy

      i. Vectors: plasmids, phages, cosmids and bacteria.
      ii. Cutters: restriction endonucleases; constructing restriction maps; RFLPs.
      iii. Pasters: Ligases.
      iv. Cloning: cut and paste into a vector, then use bacteria to produce several copies; Methods for recognition of successfully cloned copies.
      v. Copy: Polymerase Chain Reaction (PCR): Use polymerase and primers flanking the DNA region of interest to produce several copies without cloning.

   (b) Basic sequencing

      i. Gel Electrophoresis.
      ii. Chain-terminated PCR or Sanger's method.
      iii. (time permitting) Sequencing by hybridization (SBH).

References

[1] Understanding DNA and gene cloning (second ed.), K. Drlica, John Wiley & Sons, Inc. (1992). This book contains a very readable account of the various laboratory methods in recombinant DNA technology.

[2] "Towards DNA sequencing chips," P. Pevzner and R. Lipshutz, in *Proc. MFCS '94*, Springer-Verlag LNCS 841, pp. 143-158 (1994). A good survey on sequencing by hybridization.

2. Sequence Assembly

   (a) Biology

      i. Shotgun sequencing.
      ii. Ideal case - consensus sequence.
      iii. Complications: chimerism, unknown orientation, repeated regions, lack of coverage.

   (b) Models

      i. Coverage estimation by statistical model.
      ii. Shortest Common Superstring.
      iii. A weaker reconstruction model incorporating orientation.

   (c) Methods

   i. Greedy algorithm for SCS and theoretical embellishments.
   ii. Heuristic methods to aid Greedy: find overlaps, build layout (use statistics to extend overlapping matches), compute alignment for consensus to fix errors.

References

[1] "Genomic mapping by fingerprinting random clones: a mathematical analysis," E. S. Lander and M. S. Waterman, *Genomics* 2, 231-239 (1988). This article contains a probabilistic analysis of the number of contigs and oceans (gaps) in a large scale sequencing project as a function of the number of clones used (or alternatively, the coverage of the genome by the clones used).

[2] "Exact and approximate algorithms for the sequence reconstruction problem," J. D. Kececioglu and E. W. Myers, *Algorithmica* 13 (1-2), 7-51 (1995). This paper gives algorithms for the various subproblems arising in sequence assembly.

[3] "A quantitative comparison of DNA sequence assembly programs," M. J. Miller and J. I. Powell, *J. Comput. Biol.*, 1(4), 257-269 (1994). This paper presents a comparison of nearly a dozen sequence assembly programs for their accuracy and reproducibility of DNA fragments.

3. Physical Mapping

   (a) Biology
      i. Hybridization mapping; non-unique probes versus Sequence Tagged Sites (STS) as unique probes;
      ii. Types of common errors: false positives and negatives, chimerism.

   (b) Models
      i. Ideal case: interval graph recognition.
      ii. Modeling errors: a Hamming distance Traveling Salesperson Problem (TSP).

   (c) Methods
      i. Exact algorithms for testing consecutive-ones property are useful in recognizing interval graphs; Heuristic extensions to allow errors;
      ii. Heuristics for screening chimeric clones; Local improvement algorithms for finding good probe orderings by solving the Hamming TSP.

   (d) Other methods (time permitting)
      i. RH (Radiation Hybrid) mapping - the biology; Error types: false positives and false negatives; Formulation as finding the ordering and placement of markers; Greedy and local improvement algorithms.

References

[1] "Physical mapping of chromosomes using unique probes," F. Alizadeh, R. M. Karp, D. K. Weisser and G. Zweig, *J. Comput. Biol.* 2(2), 159-184 (1995). This paper describes combinatorial methods for constructing physical maps with STS probes, including techniques for the Hamming TSP solution such as simulated annealing, and screening methods for errors in the data.

[2] "Physical mapping of chromosomes: a combinatorial problem in molecular biology," F. Alizadeh, R. M. Karp, L. A. Newberg and D. K. Weisser, *Algorithmica* 13 (1-2), 52-76 (1995). This paper addresses the clone ordering problem given hybridization fingerprints with non-unique probes by solving an approximation to a likelihood function using overlap information. This paper also argues the statistical consistency of this method.

[3] "Radiation hybrid mapping: a somatic cell genetic method for constructing high-resolution maps of mammalian chromosomes," D. Cox et al., *Science* 250, 245-250 (1990). This paper introduces the RH mapping technique.

4 Protein Structure

1. Introduction

   (a) Proteins play a key role in almost all biological processes.

   (b) A protein is a linear chain of amino acid residues.

   (c) Amino acid sequence specifies three-dimensional structure.

   (d) The functional properties of proteins depend on their 3D structures.

   (e) Protein structures can be determined via experimental methods such as X-ray crystallography and NMR, but such methods are time consuming.

   (f) The protein structure prediction problem is: given the amino acid sequence which specifies a protein, determine the three-dimensional structure of the protein.

2. Levels of Structure in Protein Architecture

   (a) The one-dimensional amino acid sequence of a protein's polypeptide chain is called its primary structure.

   (b) A protein structure can be described in terms of its secondary structure, which are local regular structures such as α-helices and β-sheets.

   (c) The tertiary structure of a protein is the complete 3D structure of the protein.

   (d) Quartenary structure consists of several polypeptide chains arranged together.

3. Tertiary Structure Prediction

   (a) Energy minimization methods.

      i. Model principal forces in protein folding.

      ii. Search conformational space.

      iii. Current limitations of these approaches.

   (b) Threading

      i. Threading approaches are based on the assumption that there are a limited number of protein folds.

      ii. Formal definition of problem.

      iii. The threading problem is NP-complete.

      iv. Approximation algorithms for simpler versions of the threading problem.

      v. Heuristics.

      vi. Current limitations of threading.

4. Secondary Structure Prediction

   (a) The secondary structure problem is: given an amino acid sequence, label each amino acid residue as either alpha helix, beta sheet or other.
   (b) Certain amino acid residues show (modest) preference for particular secondary structures.
   (c) A variety of approaches - from neural nets to other statistical methods - have been tried, with overall accuracy still below 70%.

5. Motif Recognition
   (a) Structural motifs are local three-dimensional folding patterns that are commonly occurring in protein structures, and are made up of particular secondary structure units (e.g., EF-hand motif, coiled coils).
   (b) The structural motif recognition problem is: given a known local 3D structure, or *motif*, determine whether this motif occurs in a given amino acid sequence, and if so, in what positions.
   (c) The general framework for most approaches to structural motif recognition is:
      i. Build a database of subsequences which take part in a motif.
      ii. Determine whether new sequences share enough distinguishing features with the known examples of the motif to be considered a good candidate of the motif.
   (d) Probabilistic framework for motif recognition
      i. Application to coiled coils.
      ii. Window based algorithm.
   (e) Hidden Markov model approaches
      i. Introduction to HMMs.
      ii. Applying HMMs to recognizing EF-hand motifs and globins.
   (f) Limitations of current approaches to motif recognition.
      i. Limited number of known examples for a particular motif.
      ii. Differentiating closely related motifs.
      iii. Iterative learning algorithms as a possible way to overcome limited data problems.

6. Lattice Models
   (a) Proteins are represented as self avoiding walks on lattices.
   (b) A protein is modeled as a specific sequence of hydrophobic (H) and polar (P) residues.
   (c) Based on the assumption that the hydrophobic effect is the dominant force in protein folding, a simplified energy function favors H-H contacts.

(d) The protein structure prediction problem in the HP lattice model is thought to be NP-complete; however, there are some approximation algorithms for this problem.

(e) Does it make sense to use the HP-lattice model to try to solve the protein structure prediction problem?

(f) Simulations of protein folding using these simplified models can capture some of the qualitative features of protein folding.

References

[1] C. Brandon and J. Tooze. *Introduction to Protein Structure.* Garland Publishing, Inc., 1991.

[2] F. Eisenhaber, B. Persson and P. Argos. "Protein structure prediction: recognition of primary, secondary and tertiary structural features from amino acid sequence." Critical Reviews in Biochemistry and Molecular Biology (1995) 30(1):1–94.

[3] T. Defay and F. Cohen. "Evaluation of Current Techniques for Ab Initio Protein Structure Prediction." PROTEINS: Structure, Function and Genetics (1995) 23:431–445.

[4] C. Lemer, M. Rooman, and S. Wodak. "Protein structure prediction by threading methods: evaluation of current techniques." PROTEINS: Structure, Function and Genetics (1995) 23:337–355.

[5] T. Akutsu and S. Miyano. On the approximation of protein threading." In 1st Annual Conference on Computational Molecular Biology, January 1997.

[6] R. Lathrop. "Protein threading problem with sequence amino-acid action preferences is NP-complete." Protein Engineering (1994) 7:1059-1068.

[7] B. Berger. "Algorithms for protein structural motif recognition." Journal of Computational Biology (1995) 2:125–138.

[8] B. Berger, D. B. Wilson, E. Wolf, T. Tonchev, M. Milla and P. S. Kim. "Predicting coiled coils using pairwise residue correlations." Proceedings of the National Academy of Sciences (1995) 92:8259–8263.

[9] B. Berger and M. Singh. "An iterative method for improved protein structural motif recognition." In 1st Annual Conference on Computational Molecular Biology, January 1997. Journal of Computational Biology, in press.

[10] L. R. Rabiner and B. H. Juang. "An introduction to Hidden Markov models." IEEE ASSP Magazine (1986) 3(1):4-16.

[11] A. Krogh, M. Brown, S. Mian, K. Sjölander and D. Haussler. "Hidden Markov models in computational biology: Applications to protein modeling." Journal of Molecular Biology (1994) 235:1501–1531.

[12] H. S. Chan and K. Dill. "The protein folding problem." Physics Today, February 1993.

[13] H. Li, R. Helling, C. Tang and N. Wingreen. "Emergence of preferred structures in a simple model of protein folding." Science (1996) 273:666–669.

[14] G. Crippen. "Failures of inverse folding and threading with gapped alignment." PROTEINS: Structure, Function and Genetics (1996) 26:167–171.

5 Evolutionary Trees

1. How does Darwinian evolution work?
   (a) What is selection?
   (b) What are mutations?
2. What is a species?
   (a) How are species defined?
   (b) How are extant species related?
   (c) How are extant species related to extinct species?
   (d) What is a speciation event?
3. How is DNA related to evolution?
   (a) Evolution of DNA proceeds along a binary tree. How does Recombination violate this assumption?
   (b) Sequences are related by *homology* or *paralogy*
4. What is the difference between a *Gene Tree* and a *Species Tree*?
5. What observables can be used to build trees from extant species?
   (a) Morphology – for species tree
   (b) Genomics – for gene tree (and maybe for species tree)
6. Given observations, how do we find trees?
   (a) Parsimony
      i. Steiner tree in hamming space
      ii. Assumes that mutations are rare (developed for morphology)
      iii. MAX-SNP hard
      iv. Unstable
      v. Popular
   (b) Maximum Likelihood Estimation (MLE)
      i. What is stochastic model of evolution?
      ii. Which models are tractable?
      iii. What (unreasonable) assumptions do the models make?
      iv. How can we solve such models without solving entire MLE problem (i.e. are there good hacks?)?

(c) Distance Methods

  i. What is an additive metric?
  ii. What is an ultrametric?
  iii. Ultrametrics have subdominance property.
  iv. Optimal Algorithm for L_∞ for Ultrametrics
  v. Pivot relationship between Ultrametrics and Additive metrics
  vi. Heuristics

Problem Set 1

1. (a) Given only the first two nucleotides of a codon, in how many cases in the genetic code would you *fail* to know the amino acid specified by that codon?

   (b) If you knew the amino acid specified by a codon, in how many cases would you be unable to determine its first two nucleotides?

2. You are studying a gene in *E. Coli* that specifies a protein. A part of its sequence is:

   ALA PRO TRP SER GLU LYS CYS HIS

   You recover a series of mutants for this gene that show no enzymatic activity. Isolating the mutant enzyme products, you find the following sequences:

   Mutant 1: ALA PRO TRP ARG GLU LYS CYS HIS

   Mutant 2: ALA PRO

   Mutant 3: ALA PRO GLY VAL LYS ASN CYS HIS

   Mutant 4: ALA PRO TRP PHE PHE THR CYS HIS

   What is the molecular basis for each mutation? What is the DNA sequence that specifies this part of the protein?

3. A double-stranded DNA sequence, shown below, produces, in vivo, a polypeptide that is five amino acids long.

   TAC ATG ATC ATT TCA CGG AAT TTC TAG CAT GTA
   ATG TAC TAG TAA AGT GCC TTA AAG ATC GTA CAT

   (a) Which strand of DNA is transcribed, and in which direction?

   (b) Label the 5' and 3' ends of each strand.

   (c) If an inversion occurs between the second and third triplets from the left and right ends, respectively, and the same strand of DNA is transcribed, how long will the resultant polypeptide be?

   (d) Assume that the original molecule is intact and that transcription occurs on the bottom strand from left to right. Give the base sequence, and label the 5' and 3' ends of the anticodon that inserts the fourth amino acid into the nascent polypeptide. What is this amino acid?

Problem Set 2

1. Suppose we want to compute the LCS of two strings, as long as that LCS is of length at least $n-k$, but if it is less than this length, we don't care how long it is. Give an algorithm which solves this problem in time $O(nk)$ for two strings of length n.

2. (a) Suppose we are given two sequences and a scoring matrix which we use to find a maximum similarity global alignment, with explicit scores for different indels (insertions or deletions that align different characters against a blank). Suppose now that we add a fixed number a to the score for aligning any pair of characters. What quantity (in terms of a) must we add to the score of any indel, so that we preserve the relative scores of different global alignments (i.e., so that the largest scoring alignment continues to stay the largest under the new scoring scheme, the second largest is the second largest in the new scheme and so on)? Why?
(b) For global alignments, is there likely to be a log-odds interpretation for scoring matrices analogous to that for local alignments? Why or why not?

3. We saw in class that a typical score entry s_{ij} for a pair of characters i and j in the PAM-1 matrix is of the form $\frac{1}{\lambda}\log_e(\frac{q(1)_{ij}}{p_i p_j})$ for some constant λ. Here the term $q(1)_{ij}$ represents the transition probability of the (undirected) transition between i and j in one unit of evolutionary time. Write out a formula for $q(k)_{ij}$, the transition probability of changing from i to j in k units of time. Note that the i,j-th entry in the PAM-k matrix is of the form $\frac{1}{\lambda'}\log_e(\frac{q(k)_{ij}}{p_i p_j})$ for some other constant λ'. Use this formula, and the fact that $q(1)_{ij} = q(1)_{ji}$ for every pair i, j to show that PAM-k is a symmetric matrix for all $k \geq 1$.

Problem Set 3

1. Show that an optimal alignment of k species can be obtained using dynamic programming in $O(2^k N^k)$ evaluations of a cost function, d, using $O(N^k)$ space.

2. For this problem, use the sum-of-pairs metric and follow the "once a gap, always a gap" rule. Consider the following three sequences.

   (1) ACGTC
   (2) TCCT
   (3) ACGTCCT

   (a) Compute all three optimal pairwise alignments assuming a cost of 2 for each deletion and 3 for each substitution. Give the cost of each alignment.

   (b) Compute a progressive multiple alignment starting with the pairwise alignment (1,3). Now use the pairwise alignment (2,3) to merge sequence 2 into the multiple alignment. Show the resulting alignment and give its cost.

   (c) Repeat problem 2, but this time use the pairwise alignment (1,2) to merge sequence 2 into the multiple alignment. Show the resulting alignment and give its cost. Are the two alignments the same? Which has a lower cost?

   (d) What is the optimal multiple alignment?

   (e) Suppose you charge a cost of 1 for each deletion and 1 for each substitution. What is the optimal alignment? Is it unique?

3. Suppose you are studying a new plasmid (with circular DNA) that is 2500 bases long, whose restriction map you wish to construct. You treat the plasmid DNA with a set of restriction endonucleases and measure the size of the resulting fragments by gel electrophoresis to obtain the following results.

   EcoRI - 2500
   HindIII - 2500
   PstI - 2500
   MboI - 1300, 800, 400
   MboI + EcoRI - 1300, 600, 400, 200
   MboI + HindIII - 1300, 800, 300, 100
   MboI + PstI - 1000, 800, 400, 300
   EcoRI + HindIII - 2000, 500
   EcoRI + PstI - 1600, 900
   HindIII + PstI - 2100, 400

   Construct a restriction map based on this information. To break the circularity, place base pair 1 at the HindIII cleavage site.

Problem Set 4

1. In this problem, you must figure out how many clones you require for a large-scale sequencing project of a bacterium whose genome is 2 million bases long. Assume that you break the genome into fragments of average length 2000 bases each, and that you can detect clone overlaps of 10% or more. As you re-assemble the clones into contigs, some gaps ("oceans") are inevitable, and suppose you are willing to tolerate 10 gaps.

   (a) How many clones do you expect you will need? What is their coverage?

   (b) What is the probability that you will have a gap of at least 20,000 bases at the end of one of your contigs?

2. Consider a mapping problem with non-unique probes that occur at a Poisson rate of λ along the chromosome and unit clones distributed uniformly over the entire chromosome that cover it completely. Suppose we obtain the ordering of the clones using Hamming distance information of hybridization with probes (say by solving the Hamming TSP problem that arises from this instance). The goal in this problem is to show that this method is statistically consistent - in other words, as the number of probes used in the hybridization experiment increases (i.e., as λ increases), the ordering output by any method based on the Hamming distances approaches the true ordering of the clones with probability one.

   To solve this problem, first define a "true distance" d between two unit clones, say, as the sum of the differences between their respective endpoints. Then it suffices to show that as the number of probes increases, the relative ordering between pairs of clones according to the Hamming distance approaches the ordering according to the true distance defined above. In particular, for pairs i, j and k, l, if the estimated Hamming distances due to the probe hybridizations are denoted by h, then show that $h_{ij} < h_{kl} \iff d_{ij} \leq d_{kl}$ with probability one as the number of probes increases.

3. Show if the score function for protein threading ignores interactions between amino acids, while still allowing variable-length loop regions, the threading problem can be solved in polynomial time.

Problem Set 5

1. Consider the window based approach to motif recognition given in class. We are given an amino acid subsequence $a_1, a_2, \ldots, a_n$, scores $s_1, s_2, \ldots, s_{n-w+1}$, where s_i is the "score" of a w-long window starting at amino acid a_i. (E.g., for window length 5, s_1 is the score of the window containing amino acids a_1, a_2, a_3, a_4, a_5.) Show that finding the maximum window scores for all amino acid residues (i.e., for each amino acid residue, finding the maximum score of any window containing it) can be computed in $O(n)$ time, independent of the window size w.

2. (a) Give an example of an HP protein sequence of length n for which half of the residues are hydrophobic but for which there are no possible H-H contacts on a square lattice. (b) Give another simple lattice for which the same sequence can get $O(n)$ H-H contacts.

3. Show that the number of possible structures (self-avoiding walks) of length n^2 on a $n \times n$ square lattice is exponential in n.

Problem Set 6

General notes: Let T be a rooted tree with no degree 1 nodes and with leaf labels drawn from a set S. We can represent T either in the traditional way (with pointers from parents to children), or as follows. Label each internal node with the set of labels on the leaves below it. Then we can define T by the set of labels on the internal nodes of T. For example, the tree $\{\{a,b\},\{a,b,c\},\{d,e\},\{a,b,c,d,e\}\}$ describes a tree where the root has two subtrees, one containing leaves d, e, and their common parent. The other subtree below the root has two internal nodes, and so forth.

1. Consider the Species tree $\{\{a,b\},\{a,b,c\},\{d,e\},\{f,g\},\{d,e,f,g\},\{a,b,c,d,e,f,g\}\}$ and the Gene tree $\{\{a,d\},\{a,c,d\},\{f,g\},\{e,f,g\},\{b,e,f,g\},\{a,b,c,d,e,f,g\}\}$. What is the smallest number of duplication (paralogous) events which can explain this arrangement?

2. We showed that ultrametrics have the *subdominant property*, that is, for every matrix M, there is an ultrametric $U_M \leq M$, such that if U' is an ultrametric and $U' \leq M$, then $U' \leq U_M$. We can similarly define the *superdominance property* by replacing $\leq$ with $\geq$ in the above definition. For each of the following, prove or give a counter-example.

   (a) Ultrametrics have the superdominant property.

   (b) Additive metrics have the subdominant property.

   (c) Additive metrics have the superdominant property.

   (d) Metrics have the subdominant property.

   (e) Metrics have the superdominant property.

3. In class, we showed that by rooting a tree T at a leaf a we can define a centroid C^a such that $T + C^a$ is ultrametric. Suppose we wanted to root T at the midpoint between two leaves a and b. How would you define C^{ab} so that $T + C^{ab}$ is ultrametric?