Algorithms for Bioinformatics (Autumn 2011)

Study group (Wed 12.10, 10-12, B222, Veli Mäkinen)

Choose one of the problems. At study group, we'll form groups on popular items.

1. **Interval graphs.**

   Find out what is *lexicographic breadth-first search (Lex-BFS)* and how it is computed. Simulate the Lex-BFS algorithm with some interval graph.

2. **Analysis of shortest superstring approximation.**

   Study proofs of Lemmas 7.2 and 7.3 at page 63, Vazirani, *Approximation algorithms* (the part of proof for shortest superstring 4-approximation algorithm skipped at lecture).

3. **Eulerian cycle and path.**

   Find out how the linear time algorithms for finding Eulerian cycle and Eulerian path work. *Note:* Proving correctness of the algorithm is a constructive proof for one direction of Euler theorem. Other direction is straightforward.

4. **Preprocessing for gene rearrangement study.**

   Consider you have the genome sequences of two species A and B and you would like to study their rearrangement distance. Each gene in A may have several putative homologs with different local alignment score in B, and vice versa. How would you find a one-to-one mapping between all genes in A to genes in B so that the sum of the corresponding local alignment scores is maximized? Here we may assume that A has at most as many genes as B (otherwise their role can be switched). *Hint. Reduce to a graph problem and add some dummy nodes/edges.*

5. **Computation around ℓ-mers.**

   Show that computing the frequency/count of each ℓ-mer in a string $s \in \Sigma^*$ can be done in $O(|\Sigma|^\ell + |s|)$ time by filling a table of size $|\Sigma|^\ell$ and scanning s once from left to right. Here $|\Sigma|$ is the alphabet size (e.g. 4 for DNA).

   On large ℓ, $O(|\Sigma|^\ell)$ may be the dominating term. Is there any way of doing the same computation in $O(|s|)$ time?

6. **Protein sequencing.**

   Let T be the *theoretical mass spectrum* of a peptide $P = p_1 p_2 \cdots p_n$ (short sequence of amico acids), consisting of the masses of its prefixes and suffixes computed from its molecular formula. *Tandem mass spectrometry (MS/MS)* can estimate the same mass spectrum, but unfortunately, checking whether the measured spectrum M is identical to T is not enough to identify the peptide: While breaking the peptide at each possible bond, some small parts from both fragments of the molecule may be lost. Also the experimental spectrum may be completely missing some masses and may be containing some masses from background chemical noise.

For simplicity, let us assume that only the masses of prefixes are included in M and T, and that the masses can be ordered from smallest to largest corresponding to the length of the prefix. How would you define a good distance measure between M and T. How can it be computed? Can you extend your distance measure definition to include simultaneously prefix and suffix masses? Can you extend the algorithm for this case?

7. **Microarrays and probe selection.**

   Microarrays were designed for sequencing by hybridization problem, yet, they are now commonly used for measuring *gene expression*: A *probe* (DNA fragment of length ℓ) is designed for each gene such that the complementary DNA of its mRNA product hybridizes to the probe. Thousands of copies of the same probe are are attached to the spot reserved for that gene in the microarray. Expression level of a gene can be estimated by observing how much cDNA product is hybridized to that spot. *Probe selection problem* is to design a probe for each gene such that only the mRNA product of that gene hybridizes to the probe. Cross-hybridizations can occur when there is similar sequence in some other gene's DNA. How would you solve the probe selection problem given the genome sequence and the genes of interest?

8. **Correctness of UPGMA.**

   Prove that UPGMA algorithm constructs an ultrametric tree if the distances are ultrametric.

9. **Correctness of Neighbor Joining.**

   Study the proof at pages 190-191 in Durbin, Eddy, Krogh, and Mitchison. *Biological Sequence Analysis: Probabilistic models of proteins and nucleid acids.* Cambridge University Press 1998. Notice that their notation D_{ij} equals our $d_{ij} - u(C_i) - u(C_j)$.