Algorithms for Bioinformatics (Autumn 2011)

Exercise 5 (Thu 13.10, 10-12, BK107, Veli Mäkinen)

1. **Shortest common superstring and ATSP.**

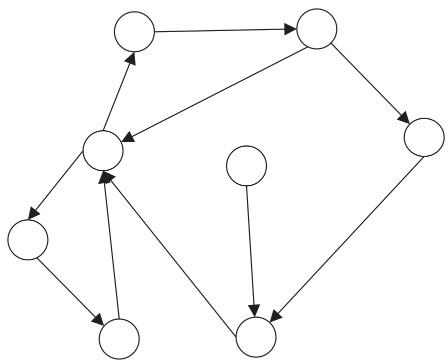
   Solve the shortest common superstring problem on set $S = \{\texttt{CTTA}, \texttt{TGAT}, \texttt{TACT}, \texttt{GATG}\}$ by reducing the problem to asymmetric traveling salesman problem through the prefix graph and dummy vertex as described at the lecture.

2. **Shortest common superstring and minimum weight cycle cover.**

   Simulate the 4-approximation algorithm for shortest common superstring problem on the same set S as above. Visualize also the minimum weight perfect matching corresponding to the minimum weight cycle cover. What is the real approximation factor achieved on this instance?

3. **Graph editing.**

   Eulerian path in a graph is a path that visits all *edges* exactly ones. Insert and delete minimum number of edges to/from the graph below so that it has an Eulerian path.



4. **Sequencing by hybridization.**

   A measurement from a hybridization experiment estimates that the 3-mer spectrum of s would be $\mathit{Spectrum}(s, 3) = \{\texttt{GAG}, \texttt{GAT}, \texttt{TAG}, \texttt{ATA}, \texttt{ATA}, \texttt{AGA}, \texttt{TAC}\}$. Construct s by the Eulerian path approach described at the course, taking into account that there might be one k-mer missing from the measured spectrum.

5. **Ultrametric condition.**

   Consider the *three-point condition*: A symmetric distance matrix $D = \{d_{ij} \mid 1 \le i \le n, 1 \le j \le n\}$ corresponds to an *ultrametric tree* if and only if $d_{ij} \le \max(d_{ik}, d_{kj}\}$ for all i, j, k. An ultrametric tree for D is an *edge-weighted tree* (positive weight associated to each edge) such that the sum of weights in the path from leaf i to node v and from leaf j to v are both $\frac{1}{2}d_{ij}$, where v is the lowest common ancestor of i and j. (Notice this is an alternative but semantically identical definition to what was used in the lectures).

a) Prove that the three-point condition can identically be stated as follows: two of the three values d_{ij}, d_{ik}, and d_{kj} are equal and one is smaller than the others.

b) Prove that the condition holds.

Research problem: Approximation algorithm for the shortest approximate superstring problem.

Recall the shortest approximate superstring problem: Find the shortest string that contains an occurrence of each given string in **S** within Hamming distance k (see more formal definition in exercise 3). Modify the 4-approximation algorithm to give an approximate solution for the shortest approximate superstring problem. Does the 4-approximation guarantee stay valid? Is the algorithm still polynomial time?

Hint. One way to proceed is to add vertices to the prefix graph that correspond to the Hamming-neighborhood *of each $s \in \boldsymbol{S}$ (all string that are within Hamming distance k from s). The challenge is to build a gadget of dummy nodes/edges and adjust edge weights so that minimum weight cycle cover works as you wish and can still be reduced to a polynomially solvable graph problem (like minimum weight perfect matching).*