Algorithms Seminar Session 4

Scribe: Sichen Zhong

10/02/14

We consider some other variations of the prefix matching problem:

1)Max Prefix *or* Suffix Matching (but not both) with Reversals
2)Max Prefix *and* Suffix Matching with Reversals
3)Max Substring Matching with Reversals
4)Max Substring Matching without Reversals

We are interested in the complexity and existence of approximation algorithms for the above problems. We give the Hardness proof of the Substring Matching without Reversals below.

Claim: Substring Matching without Reversals is NP-Hard.

Proof: We reduce Hamiltonian Cycle to Max Substring Matching with Reversals. Let $G(V, E)$ be a simply connected graph. Label the vertex set $V = \{v_1, v_2, ..., v_n\}$. Label the edge set $E = \{e_1, e_2, e_3, ..., e_m\}$. Each e_i will represent a letter in our substring. We construct the substrings using the following procedure:

-Look at vertex i. List all the e_j that have i as one of their endpoints in any arbitrary order.

-Repeat for all $i = 1, 2, ..., n$.

Since no two pairs of strings constructed in the manner above has more than one letter in common, it follows that if we can find the Hamilton Cycle of $G(V, E)$ in polynomial time, then we can also find the maximum substring matching in polynomial time. Hence, Substring Matching without Reversals is NP-Hard.

Further Open Problems to consider:

Q1: Is there a linear time approximation for the prefix matching problem?

Q2: For the Max Prefix or Suffix problem with no reversals, does Greedy work?

We give a table below on the complexity and approximability of different matching problems.

Table 1: Other types of Matching Problems

Problem	Complexity	c-approximation
Prefix Matching	NP-Hard	c=2
Max Prefix or Suffix Matching	?	c=$\frac{9}{7}$
Max Prefix and Suffix Matching	?	c=2
Max Substring Matching with Reversals	NP-Hard	c=2
Max Substring Matching without Reversals	NP-Hard	c=2