

Accepted Manuscript

A Fast Algorithm for the All-Pairs Suffix-Prefix Problem

Jihyuk Lim, Kunsoo Park

PII: S0304-3975(17)30551-0
DOI: <http://dx.doi.org/10.1016/j.tcs.2017.07.013>
Reference: TCS 11253

To appear in: *Theoretical Computer Science*

Received date: 23 February 2017
Revised date: 7 July 2017
Accepted date: 16 July 2017

Please cite this article in press as: J. Lim, K. Park, A Fast Algorithm for the All-Pairs Suffix-Prefix Problem, *Theoret. Comput. Sci.* (2017), <http://dx.doi.org/10.1016/j.tcs.2017.07.013>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



A Fast Algorithm for the All-Pairs Suffix-Prefix Problem[☆]

Jihyuk Lim^a, Kunsoo Park^{a,*}

^a*Department of Computer Science and Engineering, Seoul National University, Seoul 08826, Korea*

Abstract

The all-pairs suffix-prefix problem occurs as a subproblem of DNA sequence assembly where it is the most time-consuming part of the whole assembly. Although there are algorithms for the all-pairs suffix-prefix problem which are optimal in the asymptotic time complexity, they are slower than SOF and Readjoiner which are state-of-the-art algorithms used in practice. In this paper we present an algorithm for the all-pairs suffix-prefix problem that uses a simple data structure for storing input strings and advanced algorithmic techniques for matching, which together lead to fast running time in practice. Our algorithm is 14 times faster than SOF and 18 times faster than Readjoiner on average in real datasets and random datasets.

Keywords: all-pairs suffix-prefix problem, algorithm engineering, DNA sequence assembly

1. Introduction

The all-pairs suffix-prefix (APSP) problem is defined as follows: Given a collection \mathcal{S} of k strings S_1, S_2, \dots, S_k , find the longest suffix of S_i which is

[☆]A preliminary version of this paper was presented at the 16th International Symposium on Experimental Algorithms (SEA 2017). This research was supported by Collaborative Genome Program for Fostering New Post-Genome industry through the National Research Foundation of Korea(NRF) funded by the Ministry of Science ICT and Future Planning (No. NRF-2014M3C9A3063541).

*Corresponding author

Email addresses: jhlim@theory.snu.ac.kr (Jihyuk Lim),
kpark@theory.snu.ac.kr (Kunsoo Park)

Download English Version:

<https://daneshyari.com/en/article/6875891>

Download Persian Version:

<https://daneshyari.com/article/6875891>

[Daneshyari.com](https://daneshyari.com)