TITLE1
TITLE2
TITLE3

Tong Dong Qiu

CE-MS-2017

Delft University of Technology

Abstract

DALIGNER or Daligner

TU Delft Delft University of Technology

CE Lab Computer Engineering Laboratory

TITLE

THESIS

submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

by

AUTHOR
born in PLACE, COUNTRY

Computer Engineering
Department of Electrical Engineering
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

TITLE

by AUTHOR

Abstract

Laboratory	:	Computer Engineering
Codenumber	:	CE-MS-2017-number
Committee Members	:	
Advisor:		, CE, TU Delft
Chairperson:		, CE, TU Delft
Member:		, CE, TU Delft
Member:		, CE, TU Delft

Dedicated to my family and friends

Contents

List of Figures **vii**

List of Tables **ix**

List of Acronyms **x**

Acknowledgements **xi**

1 Introduction **1**

2 Background **3**

3 Concept **5**
3.1 Pacbio reads 5
3.2 Daligner 5

4 Specification **7**

5 Experiments **9**

6 Conclusion **11**

Bibliography **13**

List of Definitions **15**

A **17**

List of Figures

List of Tables

Acknowledgements

AUTHOR
Delft, The Netherlands
September 15, 2017

1 Introduction

2 Background

3 Concept

MAIN IS TRUE

3.1 Pacbio reads

Daligner finds alignments between long, noisy reads. Pacific Biosciences has commercially launched its first sequencer in 2011. It is able to output reads with an average of 1000 bases, which is significantly longer than **NGS!** (**NGS!**) reads [1]. In 2014, a new polymerase-chemistry combination was released, called P6-C4. This version can output average read lengths of 10000-15000 bases, and its longest reads can exceed 40000 bases [**?**]. While the drawback is that these reads have an error rate of 12-15%, this can be compensated by the distribution of these errors [**?**]. First, the set of reads is a nearly Poisson sampling of the sampled genome. This implies that there exists a coverage c for every target coverage k, such that every region of the genome is covered k times [**?**]. Secondly, the work of Churchill and Waterman [**?**] implies that the accuracy of the consensus sequence of k sequences is $O(\epsilon^k)$, which goes to 0 as k increases. This means that if the reads are long enough to handle repetitive regions, in principle a near perfect de novo assembly of the genome is possible, given enough coverage.

Important points for de novo DNA sequencing are: what level of coverage is needed for high quality assembly? And how to build an assembler that is able to deal with high error rates and long reads? Most previous assemblers work with **NGS!** reads, which are much shorter and have much lower error rates. Some algorithms used in these assemblers, such as **DBG!** (**DBG!**) [**?**] would grow too large for high error rates and long reads. Since Daligner was build, new methods of using **DBG!** with long reads have been developed, but they rely on a short read based **DBG!** to correct errors in long reads [**?**][**?**].

3.2 Daligner

The first step in an **OLC!** (**OLC!**) assembler is usually finding overlaps between reads [**?**]. BLASR [**?**] was the only long read aligner at the time, and inpsired Daligner. It uses the same filtering concept, but with a cache-coherent threaded radix sort to find seeds, instead of a BWT index [**?**]. The most time-consuming step is extending the seed hit to find an alignment. To do this, Daligner uses a novel method which adaptively computes furthest reaching waves of the older O(nd) algorithm [**?**], combined with heuristic trimming and a datastructure that describes a sparse path from the seed hit to the furthest reaching point.

Daligner performs all-to-all comparison on two input databases A, with M long reads $A_1, A_2, ...A_M$ and B, with N long reads $B_1, B_2, ...B_N$ over alphabet $\Sigma = 4$ It reports

alignments $P = (a, i, g)x(b, j, h)$ such that $len(P) = ((g - i) + (h - j))/2 \geq \tau$ and the optimal alignment between $A_a[i + 1, g]$ and $B_b[j + 1, h]$ has no more than $2\epsilon \cdot len(P)$ differences, where a difference can be either an insertion, a deletion or a substitution. Both τ and ϵ are user settable parameters, where τ is the minimum alignment length and ϵ the average error rate. The correlation, or percent identity of the alignment is defined as $1 - 2\epsilon$.

An edit graph for read $A = a_1a_2...a_m$ and $B = b_1b_2...b_n$ is a graph with $(m+1)(n+1)$ vertices $(i, j) \in [0, M] \times [0, N]$. It also has three types of edges:

- deletion edges $(i - 1, j) \rightarrow (i, j)$ with label $\begin{bmatrix} a_i \\ - \end{bmatrix}$ if $i > 0$.
- insertion edges $(i, j - 1) \rightarrow (i, j)$ with label $\begin{bmatrix} - \\ b_j \end{bmatrix}$ if $j > 0$.
- diagonal edges $(i - 1, j - 1) \rightarrow (i, j)$ with label $\begin{bmatrix} a_i \\ b_j \end{bmatrix}$ if $i, j > 0$.

An alignment between $A[i + 1, g]$ and $B[j + 1, h]$ is described as a sequence of labels from vertex (i, j) to (g, h). A diagonal edge can be either be a match edge, when $a_i = b_j$, or a substitution edge. If a match edge has weight 0, and the other edges have weight 1, the weight of the total path is the number of differences in the alignment it represents. To find suitable alignments, we have to find a read subset pairs P such that $len(P) \geq \tau$ and the weight of the lowest scoring path between (i, j) and (g, h) in the edit graph of A_a and B_b is not more than $2\epsilon \cdot len(P)$.

The O(ND) algorithm tries to find progressive waves of furthest reaching (f.r.) points until the endpoint is reached. The goal is to find longest possible paths starting at a starting point $\rho = (i, j)$ with 0 differences, then 1 difference, then 2 and so on. After d differences, the possible paths can end in diagonals $\kappa \pm d$, where $\kappa = i - j$ is the diagonal of the starting point. The furthest reaching point on diagonal k that can be reached from ρ with d differences is called $F_\rho(d, k)$. A collection of these points for a particular value of d is called the d-wave emanating from ρ, and defined as $W_\rho(d) = F_\rho(d, \kappa - d), ..., F_\rho(d, \kappa + d)$. $F_\rho(d, k)$ will be refered to as $F(d, k)$, where ρ is implicitely understood from the context.

In the O(ND) paper it is proven that:

$$F(d, k) = Slide(k, maxF(d - 1, k - 1) + (1, 0), F(d - 1, k) + (1, 1), F(d - 1, k + 1) + (0, 1) \tag{3.1}$$

4 Specification

5 Experiments

6 Conclusion

Bibliography

[1] K. Davies, "Get smrt: Pacific biosciences unveils software suite with commercial launch," April 2011. [Online]. Available: http://www.bio-itworld.com/news/04/29/2011/Pacific-Biosciences-software-commercial-launch.html

List of definitions

.. ...

A