Efficient Algebraic Interval Queries on Biomedical Sequence Annotations

Yuan Luo MS, Peter Szolovits, PhD

Massachusetts Institute of Technology, Cambridge, MA

Abstract

We present an algorithmic framework based on augmented interval tree for solving algebraic interval queries on biomedical sequence annotations with optimal time complexity.

High throughput technologies yield vast volume of data that often comes at high velocity in different biomedical subdomains including genomic sequencing analysis and clinical time series analysis. In addition, many clinical natural language processing (NLP) systems employ stand-off annotations aligned by text coordinates that are used to index large numbers of electronic medical records (EMRs). Although individual problems differ in the nature of their data, these problems share common structure in that all can be abstracted to the interval storage and query problem. Thus, the ability to efficiently store, update and query the intervals is under increasingly pressing demand.

Previous works on applying interval indexing in the biomedical field limit their attention to the overlapping queries. On the other hand, to better characterize the relations between different intervals, Allen [1] proposed the now widely accepted theory of interval algebra, which includes 13 interval relations listed below, where i, j are intervals. Although originally proposed for calculating temporal logic, Allen's interval algebra generalizes naturally to other sequence annotations. In this work, we propose an efficient interval tree based algorithmic framework for querying intervals using each of the 13 interval relation given a query interval.

$i = j$	i is **equal** to j
$i < j\ (>)$[1]	i is completely to the **left** of j
$i\ m\ j\ \ (mi)$	i's ending point **meets** j's beginning point
$i\ d\ j\ (di)$	i is **during** j
$i\ s\ j\ (si)$	i **starts** same as j and finishes before j
$i\ f\ j\ (fi)$	i **finishes** same as j and starts after j
$i\ o\ j\ (oi)$	i **overlaps** j and starts before j

We call the problem of retrieving all intervals satisfying certain relation with a given interval (point) as a stabbing interval (point) query. Finding the interval with max weight containing the query point is called a stabbing-max point query. The state-of-the-art interval tree implementation in the biomedical domain [2] relies on the basic interval tree. Its stabbing interval query finds intervals that share common region with a given interval, which translates to "$o \vee oi \vee s \vee si \vee d \vee di \vee f \vee fi \vee =$" in Allen's algebra. In practice, stabbing interval queries on fine grained relations are often desirable. For example, most existing genetic databases assign their own customized identifiers to genetic sequences at various levels including locus, transcript, and probe. To search for mutations within the breast cancer early onset gene *brac2*, one needs to rely on genomic intervals and perform a "d" query given the interval of *brac2*. However, we prove that the query time used by the basic interval tree on relations such as "d", "o" and "oi" has a worst case complexity of $O(log^2\, n)$, far from being optimal. We address this problem by proposing an augmented interval tree with optimal stabbing max-point query time, insertion time, and updating time at the same time. The key idea is to embed a secondary tournament tree for each node in the basic interval tree to keep the intervals associated with that node in sorted fashion so that retrieving the max is efficient without sacrificing insertion and updating time complexity. We then reformulate queries on difficult relations in Allen's interval algebra as stabbing max-point queries, as shown in Table 1.

Table 1. Reformulations for stabbing interval queries on difficult relations in Allen's interval algebra. The interval s is the reference interval. $w(.)$ indicates interval weight.

Allen's relation	Reformulation
$s = [x, y]\ o\ s' = [x', y']$	$y \in s'$ and $w(s') = x' > x$
$s = [x, y]\ oi\ s' = [x', y']$	$x \in s'$ and $w(s') = y' > x$
$s = [x, y]\ d\ s' = [x', y']$	$y \in s'$ and $w(s') = -x' > -x$

Complexity analysis shows that our interval tree framework attains the optimal stabbing interval query time complexity $O(log\, n + k)$ on all relations in Allen's algebra, as well as the optimal time complexity $O(log\, n)$ for insertion and updating on the tree, where n is the number of tree nodes and k is the size of the output.

References

[1] Allen, J.F. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*. 26, (1983), 832–843.

[2] Mohammad, F. et al. 2012. AbsIDconvert: An absolute approach for converting genetic identifiers at different granularities. *BMC bioinformatics*. 13, (2012), 229.

[1] Relation in parentheses denotes the inverse relation.