Multiple Sequence Alignments with Regular Expression Constraints on a Cloud Service System

Yu-Shiang Lin, National Tsing Hua University, Hsinchu, Taiwan
Chun-Yuan Lin, Chang Gung University, Tao-Yuan, Taiwan
Hsiao-Chieh Chi, National Tsing Hua University, Hsinchu, Taiwan
Yeh-Ching Chung, National Tsing Hua University, Hsinchu, Taiwan

ABSTRACT

Multiple sequence alignments with constraints are of priority concern in computational biology. Constrained sequence alignment incorporates the domain knowledge of biologists into sequence alignments such that the user-specified residues/segments are aligned together according to the alignment results. A series of constrained multiple sequence alignment tools have been developed in relevant literatures in the recent decade. GPU-REMuSiC is the most advanced method with the regular expression constraints, in which graphics processing units (GPUs) with CUDA are used. GPU-REMuSiC can achieve a speedup ratio of 29x for overall computation time based on the experimental results. However, the execution environment of GPU-REMuSiC must be constructed; it is a threshold for biologists to set up it. Therefore, this work presents an intuitive friendly user interface of GPU-REMuSiC for the potential cloud server with GPUs, called Cloud GPU-REMuSiC. Implementing the user interface via a network allows us to transmit the input data to a remote server without a complex cumbersome setting in a local host. Finally, the alignment results can be obtained from a remote cloud server with GPUs. Cloud GPU-REMuSiC is highly promising as an online application that is accessible without time or location constraints.

Keywords: Cloud Computing, CUDA, Dynamic Programming, Graphics Processing Units (GPUs), Multiple Sequence Alignments, Parallel Processing, Regular Expression, User Interface

INTRODUCTION

Sequence alignment is of priority concern in computational biology. Many methods have been developed to solve sequence alignment-related problems for biological research. Two sequences with a high sequence similarity generally refer to the functional or structural similarity among them. For a target sequence, biologists can elucidate the molecular structure, evolutionary relationships, and functional features according to the sequence similarities.

DOI: 10.4018/jghpc.2013070105
among well-studied sequences. While attempting to perform sequence alignments, Needleman and Wunsch (Needleman, & Wunsch, 1970) developed the well-known dynamic programming algorithm to solve the global pair-wise alignment problem. Smith and Waterman (Smith & Waterman, 1981) devised a similar algorithm to solve the local pair-wise alignment problem by extending the Needleman and Wunsch algorithm. Thereafter an increasing number of sequence alignment problems and corresponding algorithms have been defined and proposed. Besides the pair-wise alignment problem, multiple sequence alignment problems were also thoroughly elucidated, along with several methods developed to obtain the optimal solution for multiple sequence alignments, e.g., Carrillo and Lipman (1988) algorithm. However, a previous work demonstrated that using the dynamic programming algorithm to solve multiple sequence alignment problems is an NP-hard problem. Of the many approximation and heuristic algorithms for multiple sequence alignments developed in the past, i.e. ClustalW (Thompson, Higgins, & Gibson, 1994), a progressive multiple sequence alignment method. Most of these methods have a time complexity \(O(n^2)\) and a space complexity \(O(n)\), where \(n\) refers to a maximum length of sequences.

In practice, biologists often possess their domain knowledge, structure or conserved patterns of the sequences for analysis. Therefore, a previous work incorporated the constrained sequence alignment problem to include the biologist’s knowledge into sequence alignments in order to increase the validity of the alignment results. Myers et al. (1996) developed position constraints for multiple sequence alignments. For instance, a position constraint can be defined as a situation in which the symbol in a position \(i\) of sequence \(A\) must be aligned before the symbol in a position \(j\) of sequence \(B\) in the multiple sequence alignment results. Tang et al. (2003) devised single-residue (character) constraints for multiple sequence alignments, denoted as CMSA. Via this constraint, the user-specified residue must be aligned together in the multiple sequence alignment results. The time and space complexities of solving CMSA problem are \(O(\alpha K n^4)\) and \(O(n^4)\), respectively, where \(\alpha\) denotes the number of residues constrained and \(K\) represents the number of sequences. Thereafter, an increasing number of constrained multiple sequence alignment problems and algorithms (Tsai, 2003; Chin, Santis, Ferrara, Ho & Kim, 2004; Chin, Ho, Lam & Wong, 2005; He, Arslan, & Ling, 2006) have been proposed in the recent decade. Tsai et al. (2004) developed the MuSiC method for CMSA problem with segment constraints (pattern). The time and space complexities of MuSiC are \(O(rK n^2)\) and \(O(rn^2)\), respectively, where \(r\) represents the number of segments constrained. Based on MuSiC, Lu and Huang (2005) devised the MuSiC-ME method, in which a memory-efficient algorithm is designed to improve the pair-wise alignment with constraints by adopting the divide-and-conquer approach. The time and space complexities of MuSiC-ME are \(O(rK n^3)\) and \(O(un)\), respectively, where \(u\) denotes the total length of the constrained segment. Chung et al. (2007) proposed the RE-MuSiC method, in which regular expressions are used as the constraint formulation to estimate important residues and locate conserved structural elements. By using regular expression constraints, RE-MuSiC helps biologists to incorporate the domain knowledge in the alignment procedure. According to a previous work (Chung, Lu & Tang, 2007), the time and space complexities of regular expression for constrained sequence alignment are \(O(V^n n^2)\) and \(O(V^2 n)\), respectively, where \(V\) refers to the set of states in an automaton equivalent to an input regular expression.

Despite the development of numerous methods to solve the CMSA problem, the computational time increases with an increasing sequence length \((n)\) or number of alignment sequences \((K)\). This problem limits the applicability of those methods when the sequence length or the number of alignment sequences is large. Hence, several parallel algorithms (He & Arslan, 2005, 2006, 2006, 2006; Tsai, Lin, Chung & Tang, 2006) have been developed.