Chapter 8
Reconfigurable Hardware for DNA Matching*

Abstract. DNA sequence matching is used in the identification of a relationship between a fragment of DNA and its owner by mean of a database of DNA registers. A DNA fragment could be a hair sample left at a crime scene by a suspect or provided by a person for a paternity exam. The process of aligning and matching DNA sequences is a computationally demanding process. In this chapter, we propose a novel parallel hardware architecture for DNA matching based on the steps of the BLAST algorithm. The design is scalable so that its structure can be adjusted depending on size of the subject and query DNA sequences. Moreover, the number of units used to perform in parallel can also be scaled depending some characteristics of the algorithm. The design was synthesized and programmed into FPGA. The trade-off between cost and performance were analyzed to evaluate different design configuration.

8.1 Introduction

Bioinformatics is a field of biological science which deals with the study of methods for storing, retrieving and analyzing biological data such as DNA. It also involves finding the genes in the DNA sequences of various organisms, developing methods to predict the structure and/or function of newly discovered proteins and structural RNA sequences, clustering protein sequences into related families. Specifically, it includes solving the problem of aligning similar proteins in general and DNA in particular 10.

One of the main challenges in bioinformatics consists of aligning DNA. DNA stripes are long sequences of DNA bases, which are represented as A (Adenine), C (Cytosine), G (Guanine) and T (Thymine). In this sense, algorithms are specifically developed to reduce time spent in DNA alignment and matching, evaluating similarity degree between the subject and the query sequence. These algorithms are usually based on dynamic programming, which work well providing a fair tradeoff

* This chapter was developed in collaboration with Edgar José Garcia Neto Segundo.
between time and cost for short sequences. However, commonly these algorithm take exponentially more time as DNA sequences get longer.

The major advantage of the methods based on dynamic programming are the commitment to discover the best match. However, that commitment requires huge computational resources [7, 4]. DNA matching algorithms based on heuristics [8] emerged as an alternative to dynamic programming in order to reduce the required high computational cost. Instead of aiming at the best alignment(s), heuristics-based methods attempt to find a set of acceptable or pseudo-optimal matches. Ignoring unlikely alignments, these techniques have improved the performance of DNA matching [3, 5, 10]. Among heuristics-based methods, BLAST [1, 2] and FASTA [9, 7] stand out. Both of these algorithms have well defined procedures for the three main stages of aligning algorithms, which are seeding, extending and evaluating. BLAST is the fastest algorithm known so far [1, 2, 6]. In this chapter, we focus of this algorithm and propose a massively parallel architecture suited as an ASIC for DNA matching using BLAST. The main objective of this work is the acceleration of the aligning and matching procedures.

This chapter is organized as follows: First, in Section 8.2, we sketch briefly the steps used in the BLAST algorithm; Thereafter, in Section 9.2, we detail the proposed parallel architecture, pointing out specifically its scalability characteristics; Subsequently, in Section 9.8, we describe the setup used to implement the proposed architecture on FPGAs and evaluate the performance of the design; Finally, in Section 8.5, we draw some concluding remarks and point out directions for future work.

8.2 BLAST Algorithm

The BLAST (Basic Local Alignment Search Tool) [1] algorithm is a heuristic search-based method that seeks words in the subject sequence \( s \) of length \( w \) that score at least \( T \), called the alignment threshold, when aligned with the query sequence \( t \). The scoring process is performed according to predefined criteria that are usually prescribed by geneticists. This task is called seeding, where BLAST attempts to find regions of similarity to begin its matching procedure. This step has a very powerful heuristic advantage, because it only keeps pairs whose matching score is larger than the pre-defined threshold \( T \). Of course, there is some risk of leaving out some worthy alignments. Nonetheless, using this strategy, the search space decreases drastically, and hence accelerating the convergence of the matching process.

After identifying all possible alignments locations or seeds, the algorithm proceeds with the extension stage. It consists of extending the found alignments to the right and left within both the subject and query sequences, in an attempt to find a locally optimal alignment. Some versions of BLAST introduce the use of a wildcard symbol (\( . \)), called the gap, which can be used to replace any mismatch [7, 10]. Here, we do not allow gaps. Finally, BLAST try to improve score of high scoring pairs, HSP, through a second extension process and the dismissal of a pair is done when the corresponding score does not reach a new pre-defined threshold. HSPs that meet this criterion will be reported by BLAST as final results, provided that they do