Journal of Automata, Languages and Combinatorics ${\bf 9}~(2004)$ 1, 111–119 © Otto-von-Guericke-Universität Magdeburg

A DNA-BASED COMPUTATIONAL MODEL USING A SPECIFIC TYPE OF RESTRICTION ENZYMES

YASUBUMI SAKAKIBARA

Department of Biosciences and Informatics, Keio University, CREST, JST 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, 223-8522, Japan e-mail: yasu@bio.keio.ac.jp

and

HIROSHI IMAI

Department of Information Sciences, Tokyo Denki University Hiki-gun, Saitama 350-0394, Japan

ABSTRACT

The restriction enzymes are an important device which provides cutting operations of DNA molecules to construct a DNA-based computational model such as splicing systems [3]. In this paper, we employ a specific type of restriction enzymes which cut on both sides of their recognition sequences [6], and propose a new DNA-based computational model to generate formal languages. The new computational model realizes a context-dependent replacement system on strings and is shown to achieve universal computability using standard DNA-based operations such as annealing, cut, primer extension, and ligation.

Keywords: DNA computing, restriction enzyme, splicing system, parsing, formal language

1. Introduction

A restriction enzyme binds to DNA at a specific recognition site and then cuts DNA mostly within this recognition site. The cut can be blunt, or staggered leaving sticky ends [6]. Combined with ligases which link two fragments of DNA molecules, the restriction enzymes lead to a formal computational model of the recombinant behaviour of DNA molecules, called *splicing systems* [3]. A disadvantage of the splicing systems using restriction enzymes which cut within the recognition sites is that the system requires a number of restriction enzymes which have different recognition sites in order to implement different computational rules.

On the other hand, some restriction enzymes cut DNA molecules outside of their recognition site and these types of enzymes are more useful and interesting from the viewpoint of constructing DNA computers. Shapiro et al. [1] have successfully implemented the finite-state machine by the sophisticated use of a restriction enzyme