Complexity of a Collision-Aware String Partition Problem and Its Relation to Oligo Design for Gene Synthesis

Anne Condon[1], Ján Maňuch[2], and Chris Thachuk[1]

[1] University of British Columbia, Vancouver BC V6T 1Z4, Canada
{condon,cthachuk}@cs.ubc.ca
[2] Simon Fraser University, Burnaby BC V5A 1S7, Canada
jmanuch@sfu.ca

Abstract. Artificial synthesis of long genes and entire genomes is achieved by self-assembly of DNA oligo fragments - fragments which are short enough to be generated using a DNA synthesizer. Given a description of the duplex to be synthesized, a computational challenge is to select the short oligos so that, once synthesized, they will self-assemble without error. In this paper, we show that a natural abstraction of this problem, the *collision-aware string partition problem*, is NP-complete.

1 Introduction

There is extensive literature concerned with the study of string properties and algorithms for various string problems [8],[10],[7],[4]. Many well known string problems such as *longest common subsequence*, *hitting string*, and *bounded Post correspondence* have been shown to be NP-complete [2]. In this paper, we prove the hardness of another basic string problem, that of partitioning a string into unique substrings of bounded length, and demonstrate its relation to a problem in contemporary synthetic biology, that of synthesizing long strands of DNA.

A DNA *strand*, or *oligo*, is a string over the alphabet { `A,C,G,T` }. The *complement* O' of an oligo O, is determined from O by replacing each `G` with a `C` and vice versa, each `T` with an `A` and vice versa, and reversing the resulting string. Thus, the complement of the string `CGCATAC` is `GTATGCG`. Simplistically, a DNA *duplex* consists of a sense strand (top strand) S and an anti-sense strand (bottom strand) S', where S' is the complement of S.

Technology for synthesis of long DNA strands is enabling new advances in genomics and synthetic biology, such as production of novel or disease-resistant proteins [1]. Recently, this technology was used to artificially construct a complete bacterial genome larger than 500K bases [3]. Since DNA synthesis machines can be used to reliably produce only short DNA oligos, long DNA duplexes are typically synthesized via assembly of many short DNA oligo fragments [12].

To assemble correctly, the short DNA oligos must (a) be substrands of the sense and antisense strands of the given duplex, of length bounded by a given bound. They should (b) *cover* the duplex: if the oligos are ordered by distance

X. Hu and J. Wang (Eds.): COCOON 2008, LNCS 5092, pp. 265–275, 2008.

from one end of the duplex, they should alternate between sense and antisense strands, with some overlap between successive oligos, thus enabling assembly via hybridization of complementary parts. Additionally, (c) no oligo should self-hybridize, and (d) no pair of oligos should *collide*, that is, hybridize to each other. If either (c) or (d) happens, proper assembly is foiled. Standard polynomial-time thermodynamically-drived nucleic acid secondary structure prediction algorithms can be used to test if an oligo or pair of oligos fail conditions (c) or (d). Since there is some flexibility in the length of the oligos, there are exponentially many ways to select the oligos. The *collision-aware oligo design for gene synthesis problem (CA-ODGS)* is: given a DNA duplex and length bound k for condition (a), determine whether there is a set of oligos that satisfies all conditions (a) through (d).

CA-ODGS has been widely studied in the literature[15],[11],[5],[1],[6]: for more details see the work of Villalobos *et al.* [15] and the references therein. The variant of the problem which removes the collision condition (d) can be solved in linear time [13]. However, no polynomial time algorithm is known for the general case when all four conditions must be satisfied. The importance of addressing the collision condition will increase as progress towards multiplexed gene synthesis and entire genome synthesis continues [14].

We conjecture that the CA-ODGS is NP-hard. To provide some evidence for this, we show NP-completeness of a simplified version of the problem, which abstracts away thermodynamic details, while retaining the key challenge of collision-aware partitioning. Informally, the variation asks whether a single string (as opposed to a duplex) can be partitioned into short substrings of bounded length, no two of which are identical. For example, consider partitioning the string *theimportantthingisnevertostopquestioning* into substrings having a maximum length of 3. One possible solution is shown in Figure 1 (top). Notice, however, that some partitions may produce substrings which are identical as is the case for the other partition shown in Figure 1 (bottom) where both the substrings *th* and *ing* appear twice. We note, however, that there are instances of the problem for which it is trivial to determine that no solution is possible, dependent on the string length, n, alphabet size, σ, and maximum substring length, k. For example, the string in Figure 1 cannot be partitioned into unique substrings when $k = 1$. In general, there is no solution when $n > \Sigma_{i=1}^{k} \sigma^i$. Likewise, the problem is solvable in constant time if both k and σ are constant: as above, if $n > \Sigma_{i=1}^{k} \sigma^i$ there is no solution, otherwise, n must also be constant and all possible partitions can be checked in constant time.

The paper is organized as follows. In Section 2, we formally introduce the collision-aware string partition (CA-SP) problem and motivate its similarity to the CA-ODGS problem, as well as point out differences between them. In Section 3, a polynomial time reduction from 3SAT(3) to CA-SP with an unbounded alphabet is given which implies the NP-completeness of the CA-SP problem. In Section 4, we show that the problem remains NP-complete for alphabet of size 4. In Section 5, we add complement-awareness to the string partitioning problem and show that the problem is still NP-complete.