**An Approximation Algorithm for Motif Finding in DNA Sequences**

Hasnaa Al-Shaikhli  
Dr. Elise DeDoncker  
Department of Computer Science

### Abstract

- Motif finding is a significant problem in biology and computer science fields.
- Search for similar (not exact) motifs in multiple DNA sequences is non-trivial, and is a time-consuming problem.
- **IDEA:** design an algorithm that reduces the search space to decrease the run-time based on a d-neighbors set analysis.
- The d-neighbors set is a list of all instances for a subsequence x of length l with maximum allowed mutations ≤ d.
- Analyzing these sets to make use of the distance frequencies between neighbors without generating them would help to build the proposed algorithm and make it faster.

### Hypothesis

The proposed algorithm reduces average search time by narrowing the search space depending on prior computational knowledge.

### What is DNA?

- DNA stands for **Deoxyribonucleic Acid**.
- Recipe book that holds all instructions for making all proteins.
- DNA is a two-stranded long molecule that contains unique genetic code.
- DNA has a unique double helix shape, like a twisted ladder.
- DNA contains four basic building blocks or bases: adenine (A), cytosine (C), guanine (G) and thymine (T).
- The bases always pair together in the same way, A↔T, C↔G.

### What are DNA Motifs?

- DNA motifs are recurring short segments with expected length of 8-30 nucleotides. Motifs may occur with mutations.
- They often indicate binding sites for proteins such as transcription factors (TF).
- Motifs or Transcription Factor Binding Sites (TFBS) located upstream of genes.

**Figure 1:** Human Cell and DNA

### Motif Finding Problem

The planted (l, d) Motif Finding Problem can be described as follows:

**Input:** Given
1. Set of t sequences \( s_1, s_2, \ldots, s_t \) of length n over alphabet \( R = \{A, C, G, T\} \).
2. Two integers \( l \) and \( d \) where \( 0 \leq d < l \leq n \).

**Output:**
1. Consensus motif (original motif without mutations).
2. All subsequences (motifs) of length \( l \) at hamming distance \( d \) from the consensus motif.

### Motif Finding Algorithms

**Exact Algorithm**
Consumes a long time. Always obtains correct solution.

- **Examples:**
  - WINNOWER (2000)
  - SP-STAR (2000)
  - MITRA (2002)
  - PMS series (2005)
  - RISOTTO (2005)

**Approximate Algorithm**
- Faster.
- Does not necessarily return the correct solution.

- **Examples:**
  - MEME (1994)
  - CONSUNS (1999)
  - PROJECTION (2001)
  - MULTIPROFILE (2002)
  - Pattern Branching (2003)

### Computational Results

- Total number of d-neighbors \( N \) for a given subsequence \( x \) of length \( l \) with allowed mutation \( d \) is \( |N(x,l)|=\sum_{k=0}^{d} \binom{l}{k} \).
- If \( A, B \) are two strings of length \( l \) with \( d(A, B) \leq d \) and \( d_{A,B} \leq 2d \) and expected Hamming distance \( E_{2d} = 2d + \frac{4}{3}d^2 \).

**Figure 2:** DNA sequences with motifs upstream of genes

**Figure 3:** D-Neighbors Distance Frequencies Distribution for “AAAA”

**Figure 4:** Neighbors Distance Frequencies Distribution for “AAAA”

### Proposed Algorithm

**Input**
- \( M \): consensus motif
- \( n \): length of each DNA sequence
- \( d \): maximum allowed mutations
- DNA[n][t]: t=x DNA sequences

**Output**
- Consensus motif

**Algorithm**
For each subsequence of length \( l \) in the first sequence DNA(0) from starting positions 0 to \( n+1 \)
- If S is found in all other sequences within specificant distance range then add it to the Nominated Motifs
- Else Ignore S

### Conclusions and Future Work

- Implement the computational results in the designed algorithm to make it faster.
- Compare it with other approximate algorithms to test its performance.
- Test it on real DNA datasets such as TRANSFAC dataset.