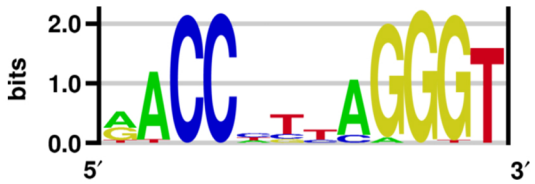
Motif Finding in DNA Sequences

Using Multiobjective Artificial Bee Colony and Asynchronous Island Model

Róbert Cuprik
email: robertcuprik@hotmail.com
supervised by Professor Pavol Návrat

Why search for motifs?

Motifs are short sequences in DNA also called binding sites. These parts precede other sequences of nucleotides in DNA that are used by individual to produce proteins.
Search for these motifs is difficult as the only information we use is that they tend to repeat between genes. This is further complicated by the fact that motifs suffer from mutations such as nucleotide substitution.



Logo of motif found in yeast.

Motif finding as a problem of optimization

Motif finding problem can be defined as multiple sequence local alignment problem, where task is to detect overrepresented motifs in multiple sequences. Such problem can be solved as a problem of optimization with following objectives and constraints.

Maximize given objective functions:
- Length of the found motif
- Support - based on number of sequences in which was motif found
- Similarity / Entropy - metrics that show how similar is motif in between sequences

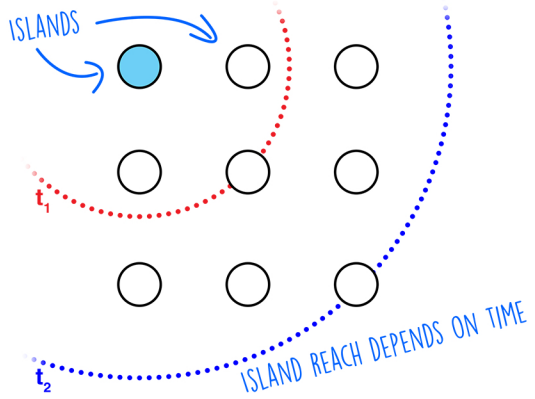
Satisfy given constraints:
- Motif length (7 - 64 nucleotides)
- Minimum Support value
- Minimum motif complexity

How to find motifs?

To find solutions for motif finding problem we use bio-inspired algorithms. Multiple artificial bee colonies work in parallel using asynchronous island model. Each colony produces new bees that represent found motifs. Bees improve in respect to objective functions between iterations and so improves the quality of found motifs.

Asynchronous Island Model

Each island represents single thread that runs sequential multi objective artificial bee colony (MOABC). Islands exchange bees asynchronously. After each iteration of MOABC island has a chance to share or receive bees from other colonies.
Each island is given position in 2D space. This position and elapsed time determine island's neighbourhood from which island can receive bees. Island's reach grows with elapsed time. We use shared memory and immutable data structures to support this migration process.



Motif filtering

Multiple islands produce many bees and subsequently many motifs. To suppress false positives we defined bucket score for additional motif evaluation. Given motif length we compute n-grams from input sequences. N-grams are scored by hamming distance to motif. Scored n-grams are then filtered by maximum hamming distance that is derived from motif length. N-grams are then split into buckets by their score. Bucket B_i contains n-grams with hamming distance i. Motif score is computed as:

$$\sum_{i=0}^{|B|-1} \frac{|B_i|}{i+1}$$

Where $|B|$ is number of buckets and $|B_i|$ is number of motifs in given bucket. Best bees from each island are evaluated by bucket score and then sorted. Only the first few are considered as final solutions.

Results

We tested our solution on dataset made by Tompa et al. In following table we show our results *(DP)* in comparison to other tools. *sTP/sFP* are true/false positive motifs found while *sSN* and *sPPV* are sensitivity and precision.

	MEME	GLAM	Weeder	ANN-Spec	QuickScore	DP
sTP	58	24	84	81	17	251
sFP	358	473	207	874	897	2708
sSn	0,111	0,046	0,161	0,155	0,033	0,166
sPPV	0,139	0,048	0,289	0,085	0,019	0,084