MARSeG - ReadMe

Theodor Sperlea

March 9, 2016

1 Introduction

Recent advantages in cloning methods, such as the MoClo system [Weber et al., 2011] allow for rapid cloning of large, periodic constructs from a small number of initial oligonucleotides (REF Paper). In some cases, this puts a certain pressure on the oligonucleotide's design: They need a high level of degeneracy in order to make the final construct as non-periodic as possible as this might lead to homologuous recombination and thus to loss of information. But also, the construct might need to be free of certain motifs, for example restrictions sites that are needed for the assembly. MARSeG is implemented in Java and R and generates DNA fragments that achieve both goals.

2 Installation and Quick Start

No installation is needed. MARSeG is run using the command line using the command

```
java -jar [location of MARSeG]/MARSeG.jar
```

and without arguments. User input is asked for after starting the program. The program needs Java and R on the computer and runs only on Windows operating systems, as the communication between Java and R is written in an Windows-specific manner. There are no other dependencies.

3 How does MARSeG work?

Starting from the (degenerate) main sequence, the suffix and the prefix that are provided by the user, MARSeG creates n motif-free or motif-avoiding (still degenerate) output sequences. The motifs are also provided by the user in a motif list (fig. 1).
For each of the n output sequences and for every motif, the program loops through all positions of the sequence and checks whether this motif could occur on this position. If that is the case, the program will reduce the amount of possibilities at one of the positions of the sequence in such a way that the motif will not occur there.
Internally, nucleotides, nucleobases or positions are represented using a 4-bit coding format, in which each of the bits represents the possibility of each of the "real" nucleotides adenine, cytosine, guanine and thymine appearing at this position (tab. 1). This representation of nucleotides as sets of possible defined nucleotides enables easy calculation of new nucleotides, as e.g. 16 NAND 1 = 15 (i.e. N NAND T = V or 1111 NAND 0001 = 1110) are then feasible.
As the decision which of the bases in the sequence is largely random (except for bases that cannot be changed with the given nucleotide from the motif), the generated sequences are all different and nee further evaluation.

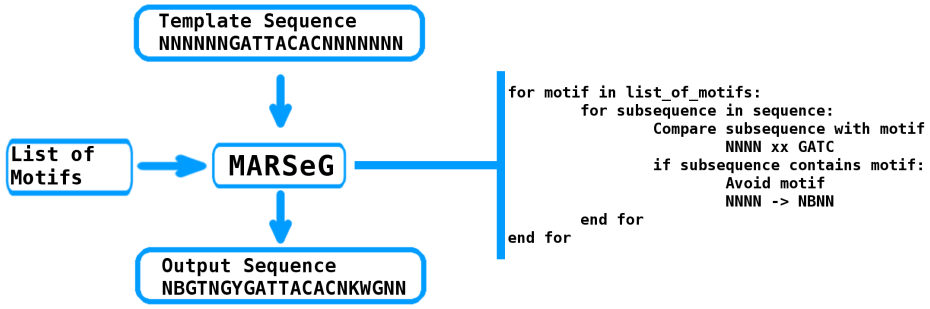

Figure 1: **The workflow of MARSeG.** The pseudocode represents only the core part of the program.

After the motif-avoiding, randomized sequences are generated, they are evaluated by calculating the mean homology and longest common substring (LCS) values of the defined sequences one can create from the degenerated

sequence. The generation of the defined sequences is a random process, again, and the program creates 100 sequences per MARSeG sequence. These 100 sequences are then compared pairwise for homology and LCS and these values are stored.
The output of the program consists of a full list of all generated sequences and two .png files which show the sequences with the lowest homology and LCS values, respectively.

Table 1: 4-bit nucleotide code

single-letter	**adenine**	**cytosine**	**guanine**	**thymine**	**4-bit code**	**value**
A	X	-	-	-	1000	8
C	-	X	-	-	0100	4
G	-	-	X	-	0010	2
T	-	-	-	X	0001	1
R	X	-	X	-	1010	10
Y	-	X	-	X	0101	5
M	X	X	-	-	1100	12
K	-	-	X	X	0011	3
W	X	-	-	X	1001	9
S	-	X	X	-	0110	6
B	-	X	X	X	0111	7
D	X	-	X	X	1011	11
H	X	X	-	X	1101	13
V	X	X	X	-	1110	14
N	X	X	X	X	1111	15

4 Input/Output

4.1 Input

After the program is started, it asks for information with the following seven questions:

"Location of motif list"

Asks for the location of a file which contains the motifs that should be avoided when generating sequences. The file should be a plain text (.txt) file and contain the DNA sequence of the motifs separated by a single white space, e.g.:

```
GTGCAC TCTAGA CTCGAG GAAGAC GTCTTC GGTCTC GAGACC GATATC GAATTC
```

The user should be cautious with the selection of the motif and the number of motifs that are chosen to avoid in the generated sequences as the amount of motifs increases the homology of the sequence (fig. 2). Furthermore, a higher amount of motifs increases the probability that the program arrives at a point where it cannot remove a given motif from the degenerated template sequence, which will cause the program to fail.

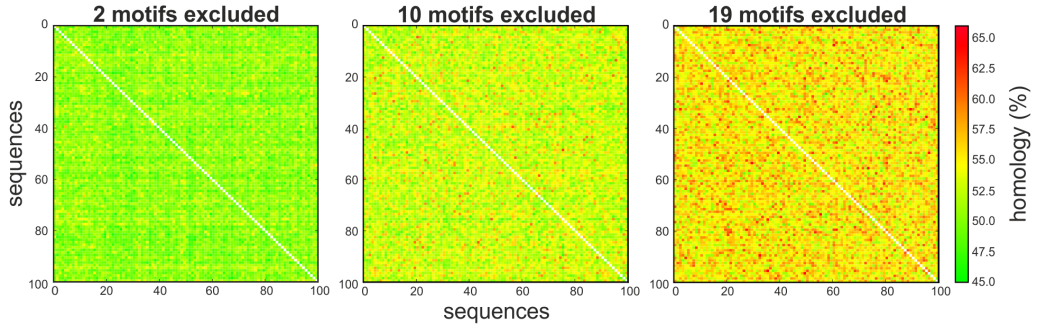

Figure 2: **An increasing number of excluded motifs increases the pairwise homology of the generated sequences.** From three sequences that were generated using MARSeG and motif lists of different lengths without prefix or suffix sequences and a main sequence of N(200), 100 defined sequences were generated, each. These sequence sets were used to calculate pairwise homology values using a Needleman-Wunsch algorithm.

"Where to save to?"

Asks for a folder in which the output files are stored.

"Pattern for main sequence"

Asks for a template for the generated, motif-less sequences. This sequence needs to have degenerated nucleotides and mustn't contain occurrences of the motifs provided in the motif list. However, as long as they do not constitute a direct occurrence of one of the motifs, defined bases are allowed. Sequences that contain repetitions of single characters can be written using a short notation as follows: `N(150)` will be handled as a sequence of 150 N's, `AGGCGV(5)S(3)N(6)` is `AGGCGVVVVVSSSNNNNNN`.

"Prefix sequence" & "Suffix sequence"

Asks for two sequences that will be added to the main sequence that can contain motifs that are on the motif list. Motif occurrences will only be removed from these sequences if a part of the motif is on the main sequence.

"Number of sequences to generate"

Asks for the number of sequences that the program will generate. As the algorithm works in a stochastic manner, the more sequences that are generated, the higher the probability that a sequence is generated with preferred characteristics. However, the runtime of the program is dependent on the amount of sequences - 100 to 150 sequences should suffice for the most applications, especially as the user will use only a couple of sequences.

"Location of Rscript.exe"

As a part of the program is written in R, the program needs to know the location of R on the computer.

4.2 Output

The is the main output file is MARSeG_output.txt, which contains all degenerated sequences generated by MARSeG. It is formated in a fasta format and contains further information in the description line of every sequence (fig. 3). The "possibilities per base" value is calculated by multiplying the individual base possibilities and dividing by the length of the sequence, the GC content is calculated using the sequences that were defined from this degenerate template and so was the fraction of sequences that contain one

```
>  Sequence pair number:  1 Possibilities per base:  1.43 GC content:  42.31% Fraction of
sequences with motifs:  0.0 %
TTTTAGGAAGGTCTCGGGAGNWNCVNTCCCTATCAGTGATAGAGADNTNCNNNNNAATTGTGAGCGGATAACAATTNNNNATNNNNATNNN
TCCCTATCAGTGATAGAGADCHNNNNNGNAATTGTGAGCGGATAACAATTNNNNNTANNTDNNRNTCGTGGGTAVGANTCAAHGSTVNAAT
TAGTNNNTRNNNNRTCCCTATCAGTGATAGAGANTNCNNNNNDAATTGTGAGCGGATAACAATTNNNNCRHNMNTWNNNTCCCTATCAGTG
ATAGAGANRNTNCNNNNAATTGTGAGCGGATAACAATTNNNSCNNCGCTGGAGACCAAGGATTAA
```

Figure 3: **One sequence as example of formatting in MARSeG_output.txt.**

of the motifs in the motif list.
Additionally, there are two images created by the program and saved as .png files into the folder specified when asked for an output folder. These two images suggest two sequences, that have either the lowest homology or LCS value.

If you use this program, please cite ...

References

[Weber et al., 2011] Weber, E., Engler, C., Gruetzner, R., Werner, S., and Marillonnet, S. (2011). A modular cloning system for standardized assembly of multigene constructs. *PLoS ONE*, 6(2):e16765.