Evolutionary Stability Optimizer User Guide

Welcome to the Evolutionary Stability Optimizer - **ESO**! We hope that using our product, you will be able to easily design synthetic sequences with higher stability and expression level.

Our basic service will allow you to easily find and rank **simple sequence repeats (SSR)** and **repeat mediated deletions (RMD)** sites for multiple sequences at once. These are mutational hotspots, which reduce genetic stability significantly [1-2].;

In mammalian or insectoid cells, methylation sites become significant factors in the host's stability, as they are epigenetic inheritance hotspots. For these cells, we offer the ability to find **methylation sites** as well, using the motifs offered by Wang et. al. [3].

For more specialized needs, users may provide their own motif file in MEME minimal file format, a very common format in the field. Using this method, you will be able to customize your engineering requirements, and define your own sites to avoid.

Finally, we believe that working with a list of problematic sites is inconvenient. We offer a further service of **optimizing the input sequence**. It is optimized for the avoidance of these sites, preservation of GC content specified by the user, and matching codon usage with the host organism. This optimization is done while maintaining codon translation within the ORF regions and avoiding changes in locations defined by the user.

In essence, you will provide input genetic sequences and receive an equivalent sequence, optimized for stability while maintaining expression level. At the end of the optimization process, the software may produce matching optimized sequences in Fasta or Genbank format, a zip report, including the annotated final sequence with the marked changes and CSV files that lists the successful constraints and objectives.

For this product's theoretical background, please refer to our paper, by Menuhin-Gruman et. al. [4]

For questions and suggestions, contact the developers at tamirtul@gmail.com and mention "ESO" in the title.

In the following sections, we will provide instructions on the use of these components.

Installation instructions – Beta Version

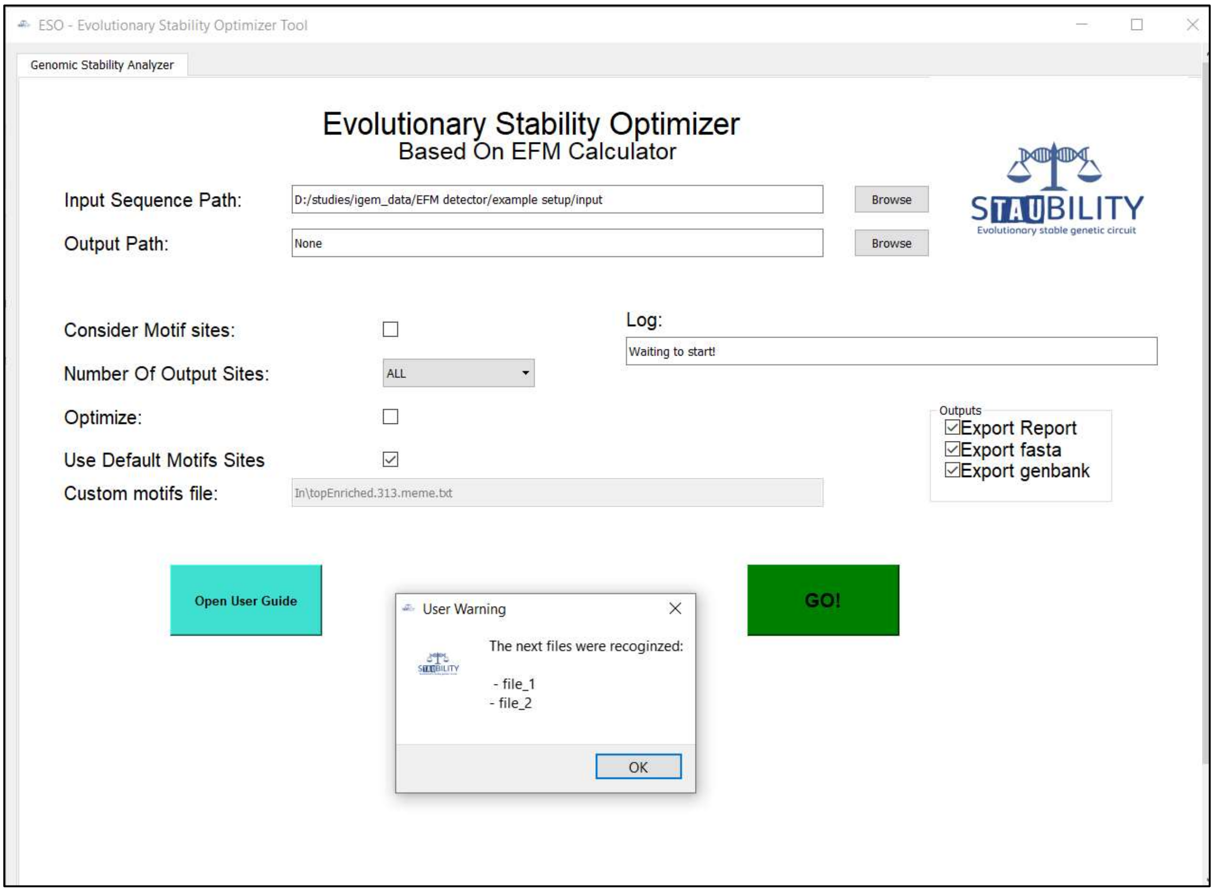
Our software is available in the "ESO" folder. The folder includes the actual software in the "GUI.exe" file and another subfolder called "In".

Please make sure that you keep the GUI file and the "ln" subfolder in the same folder! Otherwise, the software will not work. Do not change the folder names.

To launch the software:

1. After downloading the "ESO" folder with all its content, run the "GUI.exe" file **from the same folder.**
2. Your operating system may ask for permissions to run the program, grant them.
3. A CMD window will open. This is a black window, where notes will appear while the processes continue, allowing you to track the progress of the mutational-sites detection and optimization step for each input file.
4. The main software window will launch a moment after the CMD.

Main Window:



ESO - Evolutionary Stability Optimizer Tool

Genomic Stability Analyzer

Evolutionary Stability Optimizer
Based On EFM Calculator

Input Sequence Path: D:/studies/igem_data/EFM detector/example setup/input [Browse]

Output Path: None [Browse]

Consider Motif sites: □

Number Of Output Sites: ALL

Optimize: □

Use Default Motifs Sites ☑

Custom motifs file: In\topEnriched.313.meme.txt

Log: Waiting to start!

Outputs
☑Export Report
☑Export fasta
☑Export genbank

User Warning

The next files were recognized:

- file_1
- file_2

Input Sequence Path:

In this field, you define which genetic sequences you wish to analyze. Use the "browse" button to select a directory in which your input files reside. The selected directory and all its subdirectories will be searched for **Fasta and Genbank** files, or their gzipped (e.g. “fasta.gz”) versions. The analysis will be performed on all sequences contained within.

When the input folder is selected, a list of recognized filenames will pop up.

Note: We currently do not support paths in languages other than English! This includes both the directory of the files, and the filenames themselves.

Output Path:

In this field, you define in which directory to output the software’s results. Each input file’s relative path is defined as a directory within the output path, and the results for the file are exported to this directory. This ensures the relative ordering between inputs is maintained.

For example, if your input directory contains the following setup:

📂 input_folder_1

📄 sequences_1A.fasta (one sequence within)

📄 sequences_1B.fasta.gz (one sequence within) input_folder_2

sequences_2A.genbank (two sequences within)

Then your specified output directory may include:

output_folder

input_folder_1

sequences_1A_optimized.fasta (one sequence within)

sequences_1A_optimized.gb (one sequence within)

sequences_1B_optimized.fasta (one sequence within)

sequences_1B_optimized.gb (one sequence within)

sequences_1A

sequences_1A_optimized_0.gb (one sequence within)

Optimization_report_0.zip

methylation_sites.csv

recombination_sites.csv

slippage_sites.csv

sequences_1B

sequences_1B_optimized_0.gb (one sequence within)

Optimization_report_0.zip

methylation_sites.csv

recombination_sites.csv

slippage_sites.csv

input_folder_2

sequences_2A_optimized.fasta (two sequences within)

sequences_2A_optimized.gb (two sequences within)

sequences_2A

sequences_2A_optimized_0.gb (one sequence within)

sequences_2A_optimized_1.gb (one sequence within)

📄 Optimization_report_0.zip

📄 Optimization_report_1.zip

📄 methylation_sites.csv

📄 recombination_sites.csv

📄 slippage_sites.csv

Where 📂 and 📂 are new folders that were created to match the relative input order. Each subfolder 📂 contains the per-sequence output as will be described below.

Please note that a single input file can contain **several sequences**, as in "sequences_2A.fasta" in the previous example. In this case, the software will refer to the sequences with an **index**, starting from 0. For instance, 'sequences_2A.fasta' contained two sequences, so the first will be assigned with index '0' and the second will be assigned with '1'.

We will introduce the content of each output file after further explanation of the parameters in the GUI.

Note: We currently do not support paths in languages other than English! This includes both the directory of the files, and the filenames themselves.

Consider Motif Sites:

With this tick box, you can select whether to find and avoid motif sites. These are defined by PSSM matrices appearing within a file in **Meme minimal** format. These can be either the default PSSM matrices provided by our software, which enable identification of methylation sites, or other matrices provided by the user for custom engineering needs.

Number of output sites:

In this field, you define how many sites of each type to avoid, where the most unstable hotspots are avoided first. This parameter defines a **trade-off between stability and expression** – the more hotspots are considered, the fewer degrees of freedom are reserved for optimizing expression.

Optimize:

With this tick box, you can select whether to perform optimization on the input sequence and return a new sequence, ready for use. **This is highly recommended, as improving the sequence manually normally takes much effort.**

This optimization achieves several objectives:

1. Maintaining codon translation within the ORF regions.
2. Avoiding changes to the sequence in regions defined by the user.
3. GC content – maintaining the local frequency of GC nucleotides within a specified range. This is important for various biological purposes, including maintaining a high expression level and genomic stability.

4. Avoidance of the hotspots found – improves genomic stability. For computational considerations, we only offer to avoid the first ten sites from each type – ten RMD ('recombination') sites, ten SSR ('slippage') sites, and ten methylation sites when relevant.
5. Codon usage – replacing the codons used to generate amino acids in order to match the relative codon frequency within the host organism. The underlying assumption is that the genome of the host went through selective pressure for stability and expression in some form. Thus, by matching the sequence to the host, it will likely have higher levels of stability and expression as well.

When the optimize option is ticked, a menu of optimization parameters will open, which will now be introduced.

Note: at this stage, our software is limited to optimizing ten sequences. If there are more than ten sequences within the input directory, the optimization module will not be available.

Use default motifs sites:

This option is ticked by default. If the user ticks the Consider motif sites option, then the default PSSM matrices provided by the software enable identification of methylation sites.

These sites are major hotspots within mammalian and insectoid cells [3-6]. For these cells, searching for methylation sites is vital.

However, methylation sites are irrelevant for other cell types, adding unnecessary engineering constraints. Thus, do not tick this box for other types of cells.

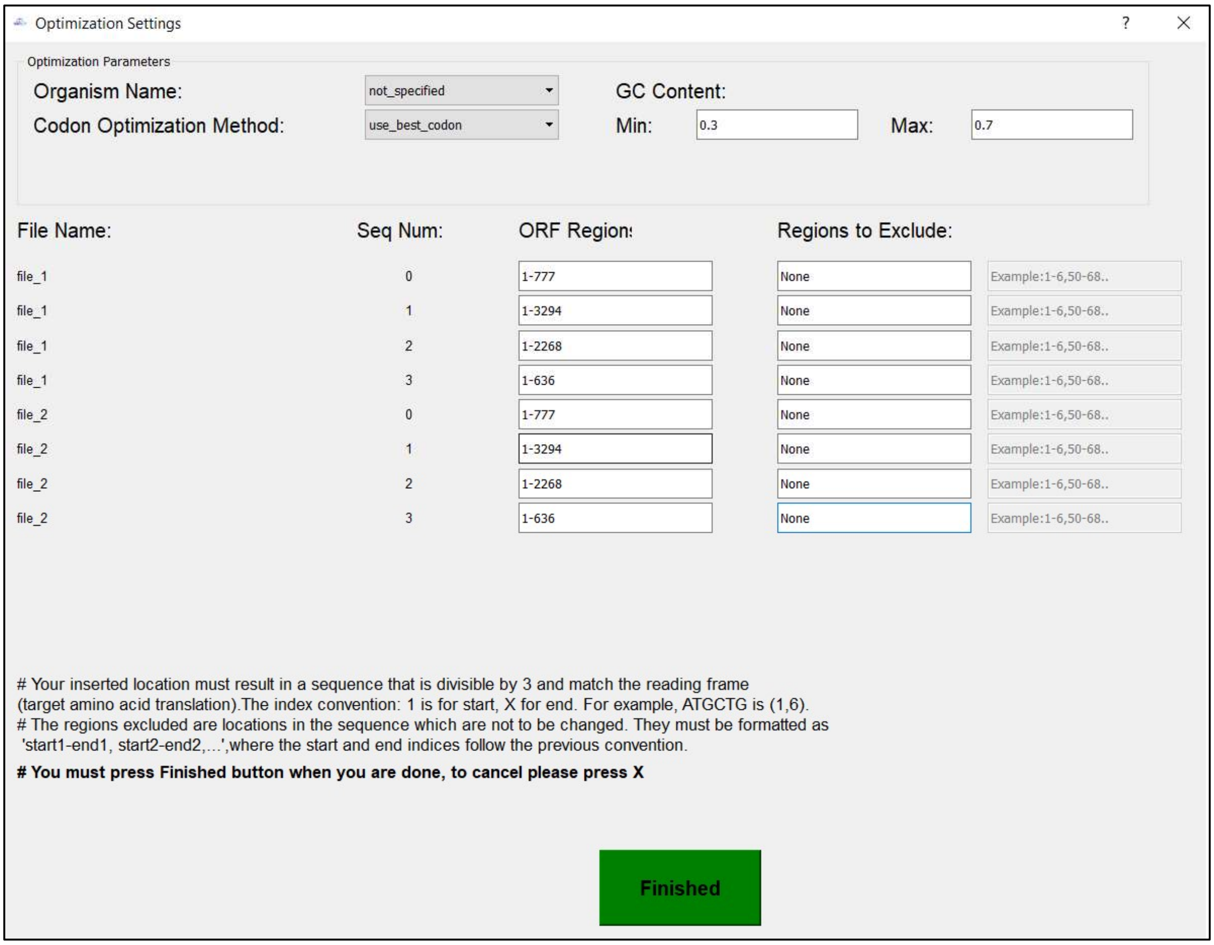
If the user has custom PSSM sites, then they may untick the box and input the path for these sites.

outputs:

Export full report – whether to output a full optimization report or just identified hotspots,

Export genbank/fasta – whether to output an optimized final sequence in genbank/fasta format.

Optimization Window:



Optimization Parameters

Organism Name: not_specified

Codon Optimization Method: use_best_codon

GC Content:

Min: 0.3 Max: 0.7

File Name:	Seq Num:	ORF Region:	Regions to Exclude:	
file_1	0	1-777	None	Example:1-6,50-68..
file_1	1	1-3294	None	Example:1-6,50-68..
file_1	2	1-2268	None	Example:1-6,50-68..
file_1	3	1-636	None	Example:1-6,50-68..
file_2	0	1-777	None	Example:1-6,50-68..
file_2	1	1-3294	None	Example:1-6,50-68..
file_2	2	1-2268	None	Example:1-6,50-68..
file_2	3	1-636	None	Example:1-6,50-68..

Your inserted location must result in a sequence that is divisible by 3 and match the reading frame (target amino acid translation).The index convention: 1 is for start, X for end. For example, ATGCTG is (1,6).
The regions excluded are locations in the sequence which are not to be changed. They must be formatted as 'start1-end1, start2-end2,...',where the start and end indices follow the previous convention.

# You must press Finished button when you are done, to cancel please press X

Organism name:

In this drop-down menu, it is possible to select a host organism. The codon usage optimization of the sequence would attempt to match this organism. The default value is 'not specified', meaning the host organism is not available and codon usage will not be optimized.

If your host organism is not within the proposed list, it is currently not supported.

Codon optimization method:

In this drop-down menu, different optimization methods can be selected for codon usage. **The default value is *use best codon***, and as it is the most popular method in literature, it will often serve your needs well.

These are the supported optimization methods:

1. Use best codon - each codon will be replaced by the most frequent synonymous codon in the host organism. This is equivalent to Codon Adaptation Index (CAI) optimization that is often described in the literature.
2. Match codon usage - the final sequence's codon usage will match as much as possible the codon usage profile of the target species. This method is used throughout the literature, for instance - Hale and Thomson 1998 [7].
3. “Harmonize RCA”:

A mapping is found between synonymous codons in the target gene and the host genome in the optimization process. This mapping is optimized to preserve the codon usage frequency between them.

In other words, each codon will be replaced by a synonymous codon whose usage in the target organism matches the use of the original codon in its host organism.

For example, AAT and AAG are synonymous codons. If in the target gene the frequency of AAG is 0.7, and in the host genome, the frequency of AAT is approximately 0.7, the AAG will be mapped to AAT. When optimizing the sequence, the frequency of AAG will be optimized to match the frequency of AAT in the host organism [8].

GC content:

You can select the minimal and maximal allowed values for GC content. This value determines the frequency of G and C within a genetic frequency, normalized between zero and one for no GC and all GC, respectively.

It is assumed that for lower values of GC content, a sequence is more stable, since it has been proven that genes with high GC content had a substantially elevated rate of mutations, both single-base substitutions and deletions [5]. On the other hand, allowing only low GC content values would limit the optimization's degrees of freedom, and permit lower maximal expression.

The default values that allow reasonable optimization are 0.3-0.7.

ORF coding regions:

For each file and subsequence within the file, you will be requested to provide the open reading frames(ORF) regions to be codon optimized.

The default value for ORF regions is 1-(last index in sequence), which means that the default ORF is the entire sequence. In order to define other ORF regions, they must be provided in the following format: "start1-end1, start2-end2, …". **The start and end indexes must be integers between 0 and the last index in the sequence.**

All ORF regions must have lengths divisible by 3. Otherwise, they cannot be divided into codons.

For example, an ORF written as "1-6, 8-13" would be the first six nucleotides of the sequence, and another ORF region of 6 nucleotides with a 1 nucleotide gap between them.

The optimization preserves the coding sequence (CDS, amino-acid translation of the reading frame). Thus, it is very important to carefully fill the indices, otherwise the wrong ORF will be defined, and a completely different protein will be produced.

Regions to exclude:

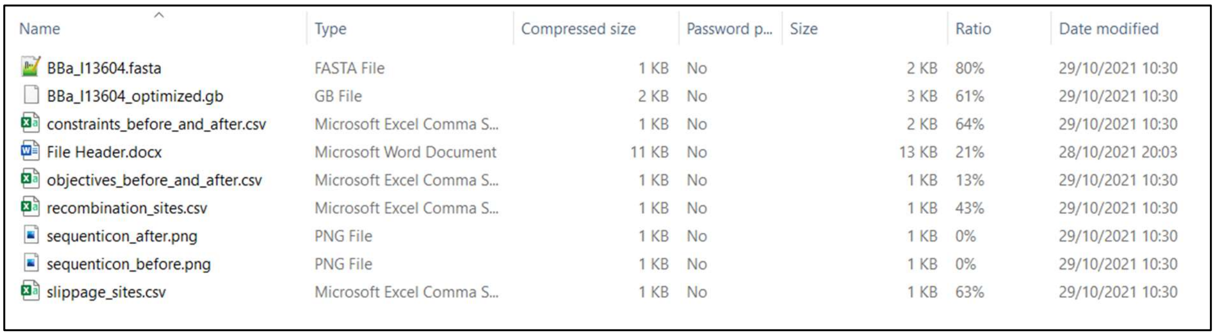
For each sequence, any number of exclusion regions may be defined. These are regions within the sequence which may not be changed – which may be important for various engineering needs.

The default value for exclusion regions is None, which means any nucleotide may be changed. In order to define these regions, they must be defined in a similar manner to the ORF regions – **though they may be of arbitrary length**.

Finish:

When you have inserted all the parameters, click on the "finish" button. You will be back in the main window. At any time, you can return to the optimization window by using the button that now appears in the main window next to the Optimize thick box. The parameters you entered are kept each time you use the optimization window.

The content of the output folder



Name	Type	Compressed size	Password p...	Size	Ratio	Date modified
BBa_I13604.fasta	FASTA File	1 KB	No	2 KB	80%	29/10/2021 10:30
BBa_I13604_optimized.gb	GB File	2 KB	No	3 KB	61%	29/10/2021 10:30
constraints_before_and_after.csv	Microsoft Excel Comma S...	1 KB	No	2 KB	64%	29/10/2021 10:30
File Header.docx	Microsoft Word Document	11 KB	No	13 KB	21%	28/10/2021 20:03
objectives_before_and_after.csv	Microsoft Excel Comma S...	1 KB	No	1 KB	13%	29/10/2021 10:30
recombination_sites.csv	Microsoft Excel Comma S...	1 KB	No	1 KB	43%	29/10/2021 10:30
sequenticon_after.png	PNG File	1 KB	No	1 KB	0%	29/10/2021 10:30
sequenticon_before.png	PNG File	1 KB	No	1 KB	0%	29/10/2021 10:30
slippage_sites.csv	Microsoft Excel Comma S...	1 KB	No	1 KB	63%	29/10/2021 10:30

As stated above, inside the output directory you will find a sub-directory and an optimized output in either Fasta or Genbank format for each input file, as selected by the user. The output will match the file hierarchy of the input directory. In file's subdirectory, you will find:

1. The optimized subsequences, each in in genbank format.
2. A list of the mutational hotspots by category (CSV format), including polymerase slippage, recombination, and if selected - methylation or custom motif sites.

   Each file includes a 'sequence_number' column corresponding to the sequence index within the input file.
3. An optimization report (zip format) for each subsequence, when 'optimize' and 'export full report' are specified.

Mutational report – csv files

The hotspots found belong to several varieties: Simple Sequence Repeats, Repeat Mediated Deletions, and methylation (or alternative PSSM's). Each of these types are summarized in a different csv file.

Simple Sequence Repeats (SSR)

These sites are composed of repeating base units. In the translation process, this could cause a polymerase slippage mistake, which would add or remove a base unit. These hotspots are ranked according to their instability in a file named slippage_sites.csv, which includes the following columns:

Start/end: the start/end index of the site within the respective sequence. The convention used is that the first index in a sequence is 0, the start index is included in the site, and the end is excluded.

Length_base_unit: the length of the repeating base unit.

Sequence: the nucleotide sequence of the SSR site.

Num_base_units: how many repeating units are within the site.

log10_prob_slippage_ecoli: the instability score of a site. It is linear with the empirical mutational probability in E. Coli. While the mutational probabilities for other organisms differ, they will be monotone with this score, and thus this score is a meaningful instability measure.

Sequence_number: the sequence index within the file. 0 corresponds to the first sequence, 1 to the second sequence, and so on.

Repeat Mediated Deletions (RMD)

RMD sites are long, identical sequences appearing in different locations within the genome. This could lead to a recombination error, where the intermittent genetic code is deleted.

These hotspots are ranked according to their instability in a file named recombination_sites.csv, which includes the following columns:

Start/end_1/2: the start/end index of the first/second site with the identical sequence. The convention used is that the first index in a sequence is 0, the start index is included in the site, and the end is excluded.

Sequence: the repeating nucleotide sequence of the RMD sites.

Location_delta: the distance between the end of the first site to the start of the second site. The closer they are, the more likely the error.

Site_length: the repeating sequence's length. The longer it is, the more likely the error.

log10_prob_recombination_ecoli: the instability score of a site. It is linear with the empirical mutational probability in E. Coli. While the mutational probabilities for other organisms differ, they will be monotone with this score, and thus this score is a meaningful instability measure.

Sequence_number: the sequence index within the file. 0 corresponds to the first sequence, 1 to the second sequence, and so on.

Methylation Sites

Methylation sites are significant epigenetic inheritance hotspots in mammalian and insectoid cells. They increase DNA folding and reduce expression levels. For these types of cells, it is imperative to avoid methylation sites as well.

In order to detect these sites, we used this comprehensive database containing 313 reported **methylation motifs** by Wang. et al 2019 [4]. We compare each subsequence within your gene with each methylation site in a probabilistic manner and ranking the subsequences by their likelihood of being a methylation site.

The user may select to provide their own PSSM matrices, which would allow designing a sequence with custom engineering requirements.

These hotspots are ranked according to their likelihood to be motif sites in a file named methylation_sites.csv, which includes the following columns:

Actual site: sequence matched.

Actual_site_rev_conj: the reverse conjugate of the site matched. Relevant, as a methylation site can be found on the reverse conjugate as well.

Matching_motif: methylation site detected.

Start/end_index: the start/end index of the methylation site. The convention used is that the first index in a sequence is 0, the start index is included in the site, and the end is excluded.

log10_site_match: the probabilistic likelihood of the subsequence matching the methylation site, measured in log scale. A score of 0 means a perfect match, and the lower the score, the lesser the match, and the less likely the site is to perform methylation.

Num_nucleotides: how many nucleotides are matched by the methylation site.

0-13: the probability of each nucleotide for each position along the methylation site.

Sequence_number: the sequence index within the file. 0 corresponds to the first sequence, 1 to the second sequence, and so on.

Optimization report – zip file

Each zip file contains the following files:

1. *Objective_before_and_after.csv* –
   Reports the objective score in CSV format. For example, an objective function could be to optimize the codon usage based on S. Cerevisiae as the host organism.

When the organism's name is "not_specified", this file will be empty, since unlike the constraints, the only objective in our software is the codon optimization.

2. *Contraints_before_and_after.csv* –

   contains the objective and the constraints that were successfully maintained in CSV format. For example, the first constraint is to keep the GC content in the range of 30%-70% in local windows of 70 base-pairs. The "start" and "end" columns specify the indices within the sequence that should maintain this constraint. The "before" and "after" contains the constraint status before and after optimization – the GC content was 30%-70% before optimization (PASS) and was kept that way.

   The second constraint is "CDS", which is short for Coding Sequence, meaning the amino-acid translation is preserved in the ORF.

   From the third constraint, we see the mutational hotspots that were detected and avoided (such as "no AA"), or regions which are not to be changed.

3. *Filename_optimized_i.gb* –

   the optimized sequence in GenBank format, not annotated.

4. *Final_sequence_with_edits.gb* –

   the optimized sequence in GenBank format, annotated with the edits during optimization. Thus, you can conveniently track the changes during the process.

5. *Sequenticon_before.png* -

   A sequenticon is an icon that is unique to the final output sequence. This is an important feature, especially when dealing with large sets of input sequences (which are often renamed or updated), because it enables the user to differentiate between sequences that otherwise might get confused with one another. Sequenticons provide a simple visual way to know that two sequences are different (different identicons) or very probably the same (same identicon).

   The "Sequenticon_before.png" refers to the sequence before optimization.

6. *Sequenticon_after.png* -

   A sequenticon of the final sequence after optimization.

In some computers, the software will also produce a PDF report. This property requires the installation of the WeasyPrint package. An example of such a report appears below. It contains the information presented in the other output files, displayed in a single PDF file. If you are interested in obtaining a PDF report automatically as well, refer to this link.

Note, within the supplementary file (separately available from our website), you can see a test case, including inputs and outputs. Feel free to explore it!

References

[1] Jack, B. R., Leonard, S. P., Mishler, D. M., Renda, B. A., Leon, D., Suárez, G. A., & Barrick, J. E. (2015). Predicting the Genetic Stability of Engineered DNA Sequences with the EFM Calculator. ACS synthetic biology, 4(8), 939–943. https://doi.org/10.1021/acssynbio.5b00068

[2] Renda, B. A., Hammerling, M. J., & Barrick, J. E. (2014). Engineering reduced evolutionary potential for synthetic biology. Molecular bioSystems, 10(7), 1668–1678. https://doi.org/10.1039/c3mb70606k

[3] Greenberg, M.V.C., Bourc'his, D. The diverse roles of DNA methylation in mammalian development and disease. *Nat Rev Mol Cell Biol* **20,** 590–607 (2019). https://doi.org/10.1038/s41580-019-0159-6

[4] Curradi, M., Izzo, A., Badaracco, G., & Landsberger, N. (2002). Molecular mechanisms of gene silencing mediated by DNA methylation. *Molecular and cellular biology*, *22*(9), 3157–3173. https://doi.org/10.1128/mcb.22.9.3157-3173.2002

[5] Newell-Price, J., Clark, A. J., & King, P. (2000). DNA methylation and silencing of gene expression. *Trends in endocrinology and metabolism: TEM*, *11*(4), 142–148. https://doi.org/10.1016/s1043-2760(00)00248-4

[6] Baylin, S. DNA methylation and gene silencing in cancer. *Nat Rev Clin Oncol* **2,** S4–S11 (2005). https://doi.org/10.1038/ncponc0354

[7] Hale, R. S., & Thompson, G. (1998). Codon Optimization of the Gene Encoding a Domain from Human Type 1 Neurofibromin Protein Results in a Threefold Improvement in Expression Level inEscherichia coli. Protein Expression and Purification.188-185 ,(2)12 ,

[8] Claassens, N. J., Siliakus, M. F., Spaans, S. K., Creutzburg, S. C., Nijsse, B., Schaap, P. J., ... & Van Der Oost, J. (2017). Improving heterologous membrane protein production in Escherichia coli by combining transcriptional tuning and codon usage algorithms. PloS one, 12(9), e.0184355