Computational and Structural Biotechnology Journal 19 (2021) 6064–6079

COMPUTATIONAL AND STRUCTURAL BIOTECHNOLOGY JOURNAL

journal homepage: www.elsevier.com/locate/csbj

Determinants of efficient modulation of ribosomal traffic jams

Sophie Vinokour [a], Tamir Tuller [a,b,*]

[a] Department of Biomedical Engineering, Engineering Faculty, Tel Aviv University, Israel
[b] The Sagol School of Neuroscience, Tel Aviv University, Tel-Aviv 69978, Israel

ARTICLE INFO

Article history:
Received 19 July 2021
Received in revised form 17 October 2021
Accepted 20 October 2021
Available online 8 November 2021

Keywords:
Ribosomal traffic jams
Synthetic biology
Codon usage
mRNA translation
Genome evolution

ABSTRACT

mRNA translation is the process which consumes most of the cellular energy. Thus, this process is under strong evolutionary selection for its optimization and rational optimization or reduction of the translation efficiency can impact the cell growth rate. Algorithms for modulating cell growth rate can have various applications in biotechnology, medicine, and agriculture. In this study, we demonstrate that the analysis of these algorithms can also be used for understanding translation.

We specifically describe and analyze various generic algorithms, based on comprehensive computational models and whole cell simulations of translation, for introducing silent mutations that can either reduce or increase ribosomal traffic jams along the mRNA. As a result, more or less resources are available, for the cell, promoting improved or reduced cells growth-rate, respectively. We then explore the cost of these algorithms' performance, in terms of their computational time, the number of mutations they introduce, the modified genomic region, the effect on local translation rates, and the properties of the modified genes.

Among others, we show that mRNA levels of a gene are much stronger predictors for the effect of its engineering on the ribosomal pool than the ribosomal density of the gene. We also demonstrate that the mutations at the ends of the coding regions have a stronger effect on the ribosomal pool. Furthermore, we report two optimization algorithms that exhibit a tread-off between the number of mutations they introduce and their executing time.

The reported results here are fundamental both for understanding the biophysics and evolution of translation, as well as for developing efficient approaches for its engineering.

1. Introduction

mRNA translation accounts for up to 75% of the cell's total energy [1–5] and occurs in all living organisms [6]. Thus, deciphering, modelling, and engineering this process have important implications to every biomedical discipline, such as molecular evolution and comparative genomics [7], medicine and human health [8], biotechnology [9], agriculture [9], and more. Therefore, translation is under extensive evolutionary selection for minimizing the resources it consumes [4–5,10].

In the case of biotechnological procedures such as heterologous protein production, mRNA translation efficiency can have a very significant impact on the host [3]. Specifically, it can affect the cellular growth rate and thus its ability to generate heterologous proteins.

The number of ribosomes in a cell is limited (for example, in a yeast cell there are about 240,000 ribosomes[11]); this can cause a competition between the mRNA molecules, with possibly many ribosomes scanning the same transcript concurrently. For example, if more ribosomes bind to a certain mRNA molecule then the pool of free ribosomes in the cell is depleted, and this may lead to lower initiation rates in the other mRNAs [7,12–14].

It was shown in previous works [15] that one can modulate the free ribosomal pool (FRP) by reducing traffic jams via the introduction of silent engineered mutations within the first 50 codons of the endogenous genes, while restricting the change in translation efficiency of the mutated genes. It was shown that the FRP modulation led to a significant increase in both the growing rate and cell titer.

Here, we aimed at further understanding the performances and characteristics of algorithms for ribosomal pool modulation via the introduction of silent mutations.

Specifically, we study the possibility to minimize the number of mutations and the changes in the level of local translation

* Corresponding author at: Department of Biomedical Engineering, Engineering Faculty, Tel Aviv University, Israel.
E-mail address: tamirtul@tauex.tau.ac.il (T. Tuller).

https://doi.org/10.1016/j.csbj.2021.10.030

Nomenclature

Abbreviation/symbol	*Description*
ASEP	Asymmetric exclusion processes
CAI	Codon adaptation index
FRP	Free ribosome pool
GG	Global Greedy Version
LG	Local Greedy version
mRNA	Messenger RNA
MRS	Modified Region Size
ORF	Open reading frame
Pval	p-Value
R	Translation rate
RC	Read count
RD	Ribosome density
Ribo-Seq	Ribosome sequencing
RFM	Ribosome Flow Model
RFMIO	Ribosome flow models with input and outputs
RNA	Ribonucleic acid
tAI	tRNA adaptation index
TASEP	Totally asymmetric simple exclusion process
tRNA	Transfer RNA
TRT	Translation rate thresholds
UTR	Untranslated region
WCS	Whole cell simulation

efficiency as they may have unforeseeable impact to the intracellular balance and the organism fitness. In addition, we study the confinement of the mutations to a restricted region in order to reduce the challenge related to genome editing and to decrease the computational time of the algorithm. Furthermore, we tried to better understand how relevant variables such the expression levels of the gene and the efficiency of the modified codons tend to affect the algorithms. To this end, we study the performances of two relevant algorithms in terms of all the aspects mentioned above and their computational time.

Finally, we study the usage of our algorithms for decreasing and increasing the FRP by increasing and decreasing ribosomal traffic jams, respectively [16].

The approaches reported here can be used both for the reduction and for the enlargement of the cells growth rate, and thus can be employed in various biotechnical targets such as heterologous gene expression and attenuated vaccine development. Furthermore, we believe that the reported results can help us to better understand the biophysics and evolution of translation in endogenous genes. While there are quite a few previous studies the present translation models, to the best of our knowledge, this is the first study that demonstrated how optimization algorithms that are based on such models can be helpful for understanding translation evolution and biophysics.

2. Results

We considered and compared the performances of two strategies for modulating the ribosomal density. Both are "hill climbing" algorithms, but with different methods for obtaining the local optima (see Fig. 1):

1. "Global greedy" Version (GG) – where at each iteration a silent mutation that yields the largest increase/decrease of the ribosome density in the target mRNA is chosen out of all possible silent mutations.
2. "Local Greedy" Version (LG) – where at each iteration a silent mutation is introduced only if the randomly chosen silent codon increases/decrease (as required) ribosome density. This approach partially mimics evolution where an advantageous mutation is fixed.

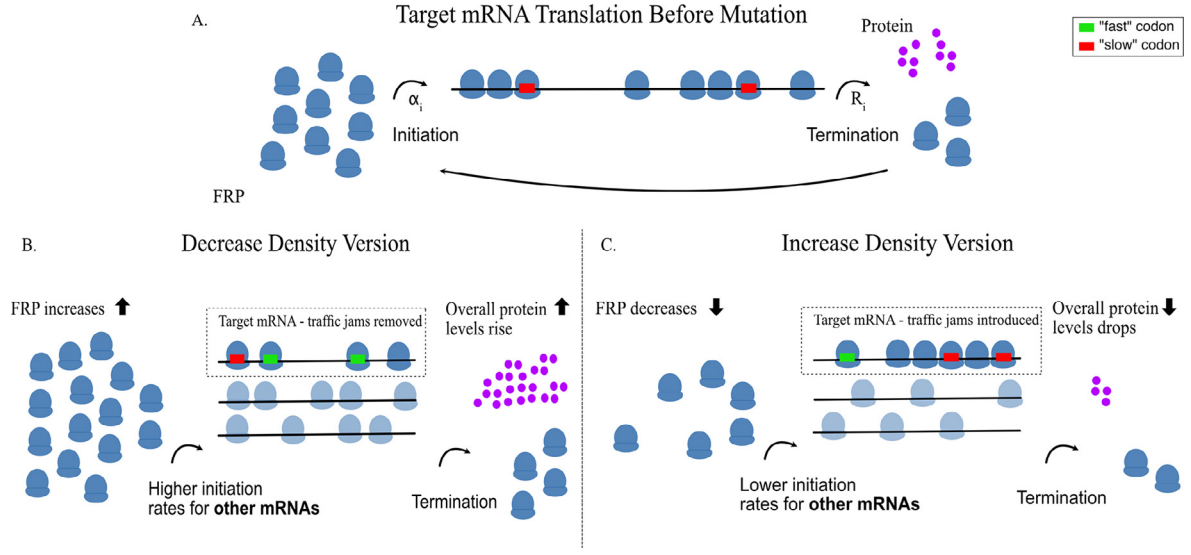


Fig. 1. A. An illustration of the target mRNA before the algorithm introduces silent mutations. α_i – is the mRNA's initiation rate of gene i, that is dependent on both the local mRNA-specific features and the size of the FRP (the bigger the pool, the higher is the ribosome diffusion rate). $\mathbf{R_i}$ – The protein translation rate of mRNA i, which is the same rate at which the elongating ribosomes are released back to the FRP. Red and green blocks represent "slow" and "fast" synonyms codon, respectively (see Methods for calculation of codon translation rates), where presence of "slow" codons create ribosomal traffic jams, "waiting" for the "slow" codon to be translated. B. The translation simulation after algorithm optimization when decreasing the mRNA ribosomal density: ribosomal traffic jams are removed by the introduction of "fast" synonymous codon (represented by green blocks), as a result the number of ribosomes on the target mRNA is reduced, the FRP increases, more ribosomes defuse to other mRNA in the cell, and the overall protein production level rises. C. The complementary case of increasing ribosome density, where mutation to "slow" synonymous codon introduced traffic jams on the target mRNA, consequently reducing the FRP and the overall protein levels as a result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In the case of many local maxima the LG may miss the right solution while the GG can find it.

In addition, we studied various constraints related to these algorithms (see Fig. 2).

First, minimizing the number of mutations in gene editing is a desirable challenge: the larger the number of mutations and the more spread they are along the gene, the more challenging and expensive it is to perform the relevant genomic editing.

Second, a large change of the relative translation rate of genes can bring an unforeseeable impact to the intracellular balance and the organism fitness.

Thus, we estimate the effect of minimizing the number of mutations in terms of running time and the number iteration steps on the algorithm performance, as well as other various constrains (e.g., translation rate threshold (TRT) and the modified region size (MRS), see Methods for extended details).

3. Performances of the algorithms *vs.* the number of steps, running time, and the number of mutations introduced

3.1. Performances vs. the number of steps and mutations

At the first step, we aimed at understanding the marginal effect of the number of steps and the number of mutations of our algorithms on the performances. Thus, we computed the mean effect of a step on the FRP over the analyzed genes.

In each graph in Fig. 3 the FRP is calculated by averaging the FRP over all the analyzed genes at each mutation/iteration.

Fig. 3 describes the average FRP change as a function of the number of silent mutations (Fig. 3A and B) and iterations (Fig. 3C and D) performed by the algorithms (GG and LG). As can be seen, the LG algorithm requires substantially more mutations to reach the same change in the FRP compared to the GG algorithm (Fig. 3A and B): 20 mutations in the GG algorithm reach the same number of ribosomes as 65/150 mutations of the LG version for increasing/decreasing cases, respectively. As for the number of iterations performed by the algorithms (Fig. 3C and D), we see an even larger gap. 20 iterations are required by the GG algorithm, as opposed to 160 iterations for increasing the density and 220 iterations for decreasing the density that are required by the LG algorithm. Thus, if we aim at minimizing the number of mutations an algorithm such as GG is superior. It is important to emphasize that minimizing the number of mutations is important due to the fact that more mutations increase the probability that we will introduce an unintended deleterious modification to the cell. In addition, introducing more mutations is usually a more challenging technical endeavor.

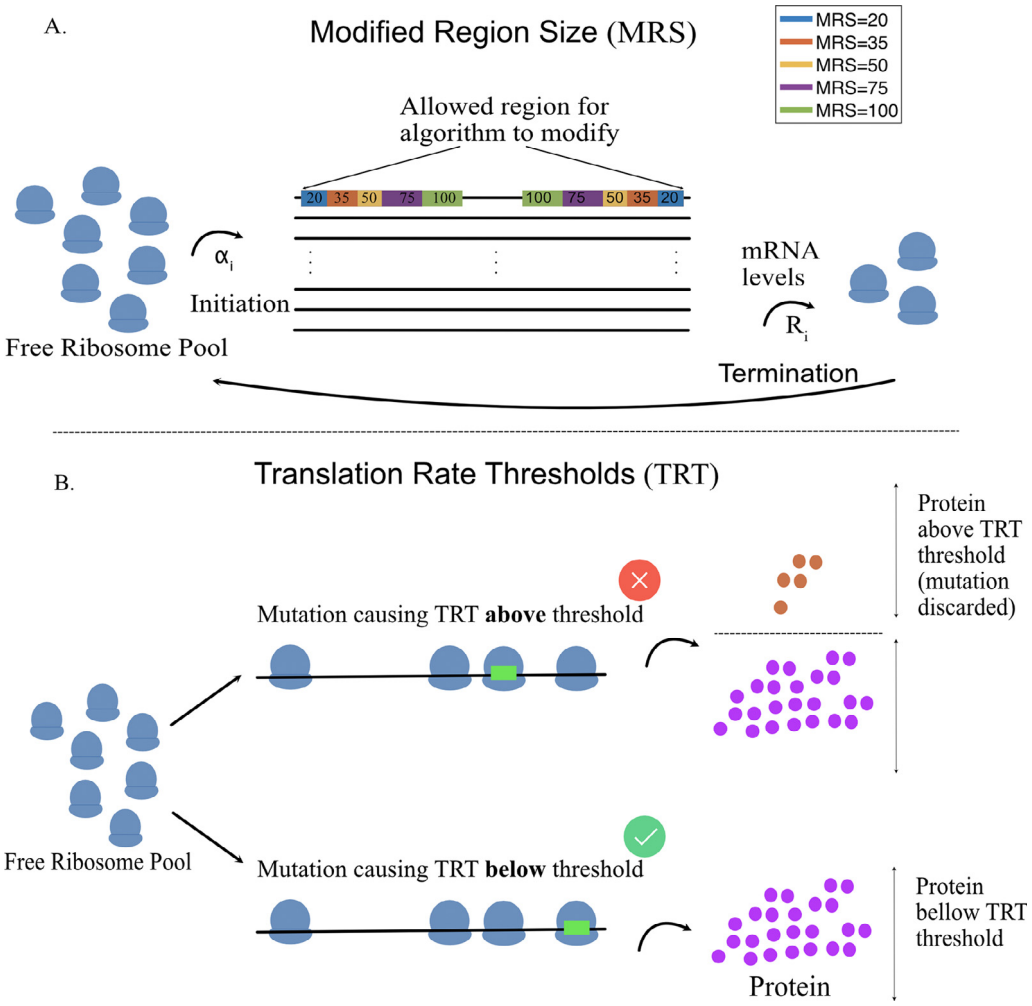


Fig. 2. A. An illustration of the modified region size, where we evaluated five region sizes (20, 35, 50, 75, 100 codons) from the beginning and to the end of the ORF, where mutations are allowed (omitting the very beginning and end). B. An illustration of the translation efficiency, where a silent mutation is made only if it brings a change to the FRP while maintaining the target mRNA's protein translation level under a certain threshold rate, thus the top mutation is discarded and the lower mutation is chosen. We evaluated five translation rate thresholds, for both increasing and decreasing the FRP (1, 5, 10, 25, 50%).

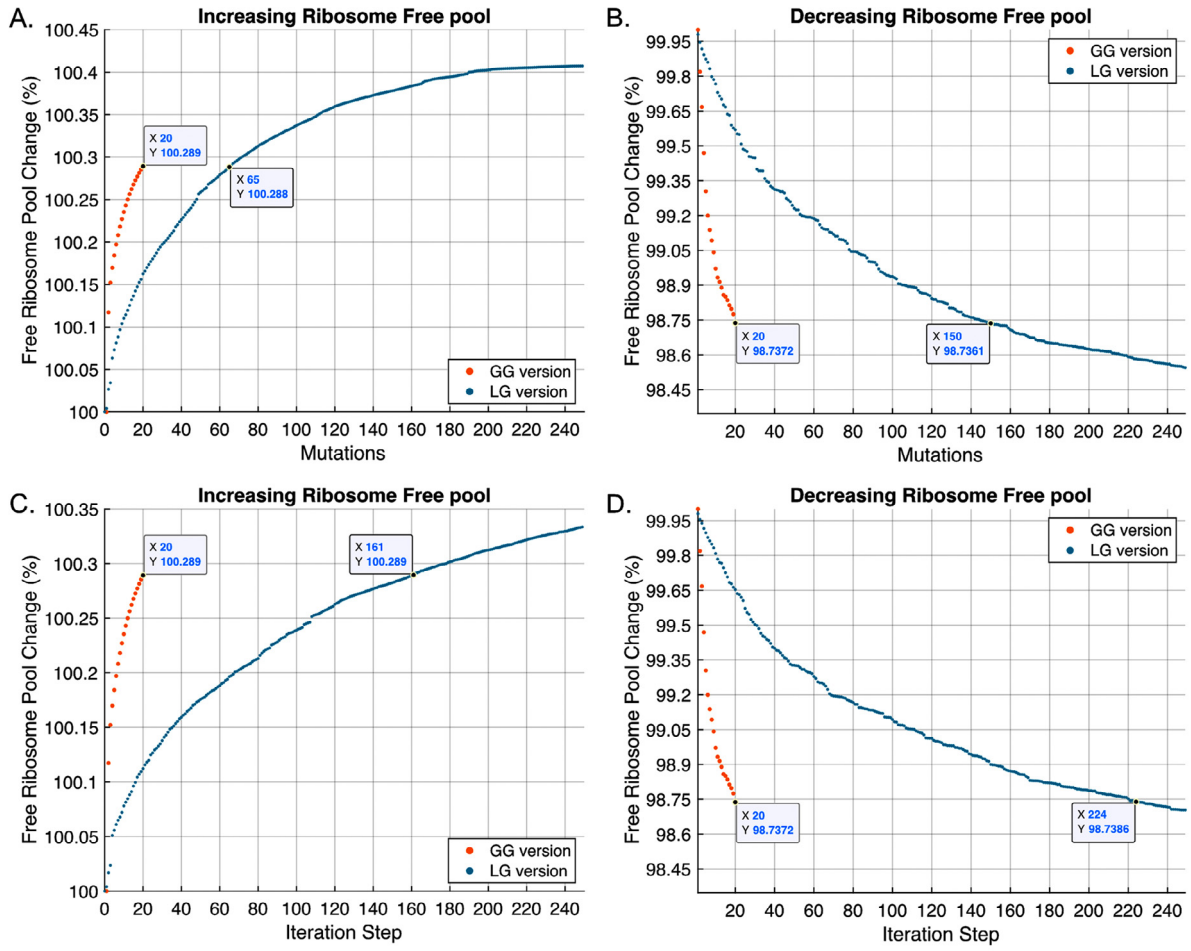


Fig. 3. Average FRP changes as oppose to the number of mutations introduced by the algorithm (A., B.) and the number of iterations of the algorithm (C., D.) for both GG and LG versions. In the GG algorithm, each iteration results in a mutation, unlike in the LG case where only iterations that increase/decrease the density on the target mRNA result in a mutation. The analysis here corresponds to the 250 most highly expressed genes, with a TRT of 10%.

3.2. Performances vs. running time

Computational simulation of a whole cell translation model result in large running time. Thus, reducing the simulation running time is a desirable goal. In Fig. 4, we estimate the cost of reducing the running time in terms of algorithm performance: comparing the average performance of the GG and LG algorithms, as a function of the average running time. The average performance is

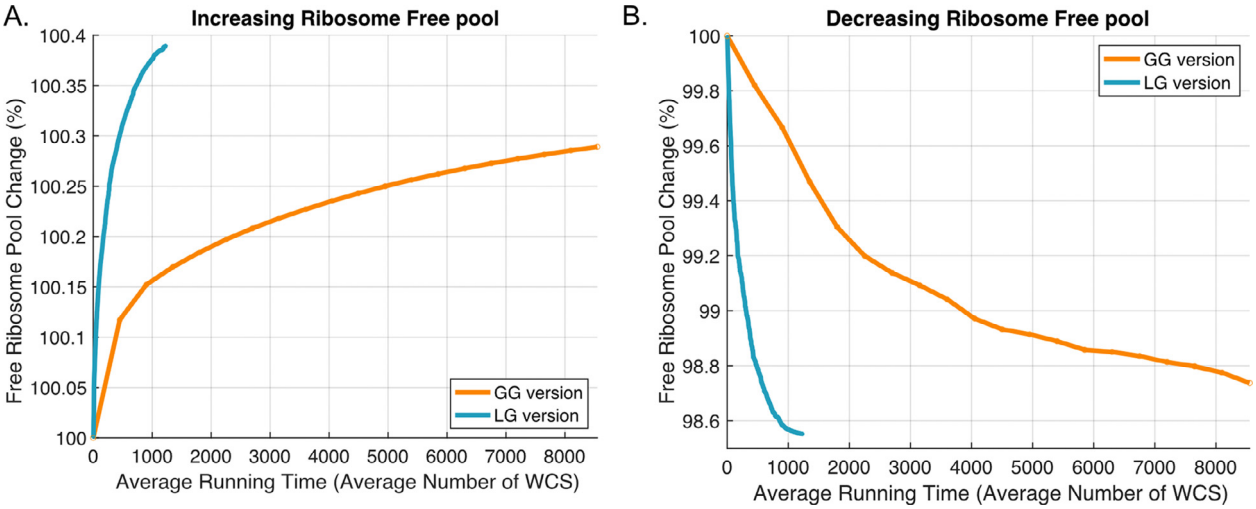


Fig. 4. The average running time, measured in an average number of WCS, to reach a certain change in the FRP (in %), for both increasing (A.) and decreasing (B.) versions. For example, to reach an average enlargement of 100.1% of the size of the FRP the LG version will run an average of 4000 WCS and generate four mutations, compared to the LG, that will take only 50 WCS but will generate 10 mutations on average. The result presented here are for 250 most highly expressed genes, with a TRT of 10%.

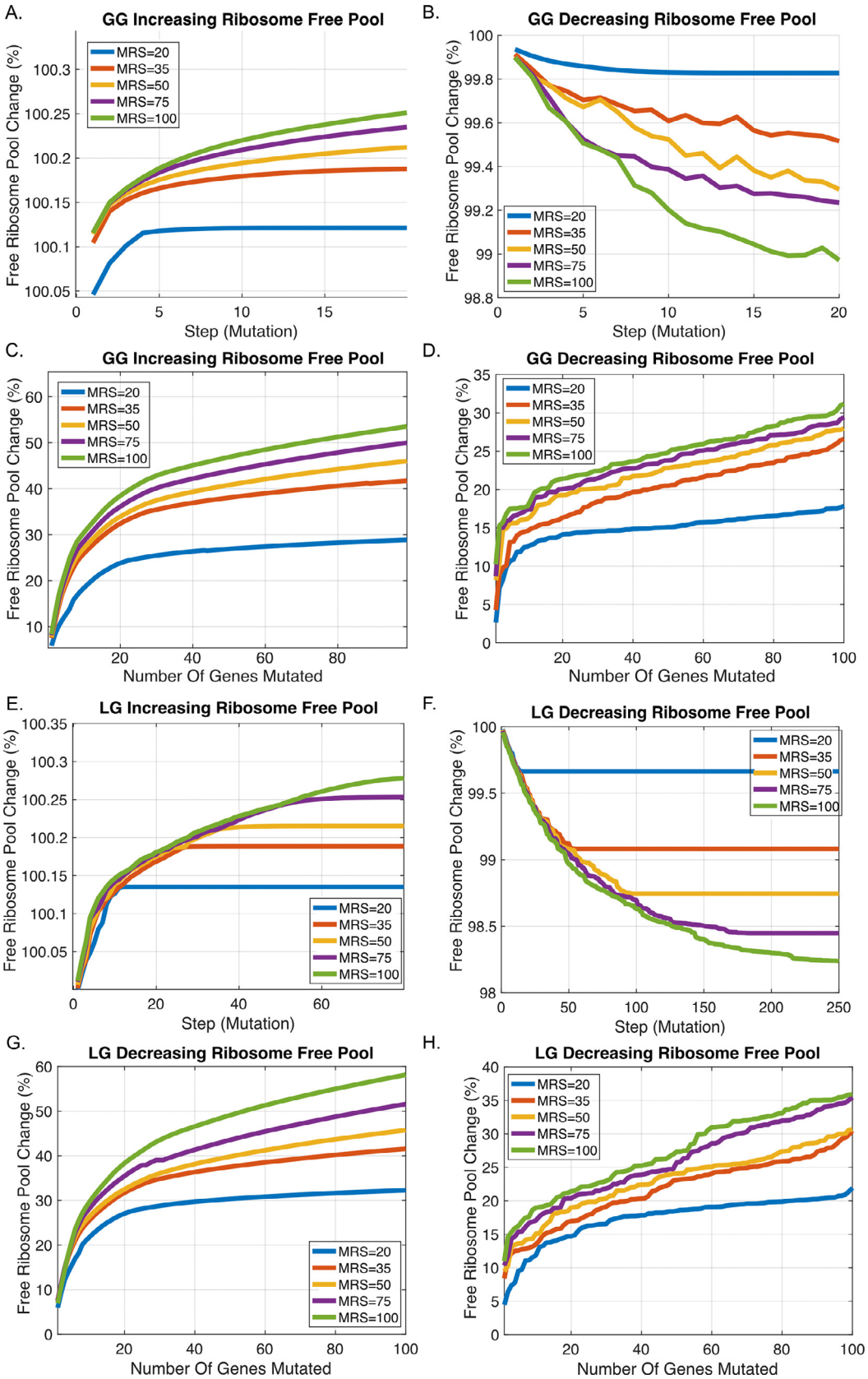


Fig. 5. The impact of MRS on algorithms performance. The mean effect on the FRP size per gene when additional mutations are introduced (A., B. for increasing and decreasing FRP versions for GG and E., F. for LG, respectively), and the effect when additional genes are mutated (C., D. For both versions for GG and G., H. for LG). Each color is the MRS in the beginning and end of the ORF that allows mutations. Note that while adding mutations to increase/decreases the FRP, the algorithm can also saturate and converge to a local optimal point; in such case additional mutations don't further affect the FRP this is represented by a constant line in the figure (in E. and F.). In this analysis the TRT was set to be 10% for both increasing and decreasing the FRP.

calculated by averaging the FRP over all the analyzed genes at each mutation. The average running time is estimated based on the number of times each algorithm performs a whole cell simulation (WCS) (see details in the Methods section).

Fig. 4 describes the average running time measured in WCS. As can be seen, it takes exponentially more time to improve the FRP with GG in comparison to LG. For example, to get 0.25% improvement in the FRP we will need around 5000 WCS in the case of GG and 120 in the case of LG. Thus, the results reported here and in the previous sub-section suggest a trade-off between the number of mutations and the running time for both GG and LG algorithms.

4. The effect of the algorithm constraints, modified region size, and change in translation rate threshold on its performance

In an effort to minimize the impact of the algorithms' modification on various additional intracellular processes, while modulating the FRP, we impose several constrains in our algorithms.

In this section we aimed at learning the impact of these constrains on the algorithm performance. We believe that the analyses reported here will allow us understanding the optimal constraints (e.g., size of the modifies region and the allowed translation rate change) that still allow a substantial change in FRP.

4.1. Mutations in a modified region size vs. performance

In an effort to confine the mutations to a specific region of the mRNA molecule for both computational and gene editing reasons we impose an MRS constraint in the beginning and the end of the target mRNA. Both algorithms, that ran with an MRS τ, allow mutations to be made only in the first and last τ codons, not including the very first 15 and last 10 codons due to initiation and termination signals [17]. To assess the impact MRS has on the performance of the algorithm, we are evaluating five sizes of MRS (20, 35, 50, 75, 100 codons). The MRSs are tested by both algorithms, both for decreasing and increasing the FRP, and the results are reported in Fig. 5.

The figure includes both the mean effect on the FRP size per gene when additional mutations are introduced (Fig. 5A, B, E and F) and the effect when additional genes are mutated (Fig. 5C, D, G and H).

As can be seen (see Fig. 5C and D) in both cases large fraction of the effect (50% when increasing the FRP and 27% when decreasing the FRP), relatively to the initial non-mutated FRP and after editing 100 mutated genes, appears when working with MRS which is <75. Increasing the MRS to 100 gives a 55%/32% rise/drop in the FRP relatively the initial non-mutated FRP for the increasing/decreasing version respectively.

These results suggest that practically we can modify only the first and last 50 codons to find a good computationally efficient and biotechnologically cheap solution.

It is important to emphasize that Fig. 5 (A and B (GG), E and F (LG)) report the average impact obtained on the FRP when of mutating only **one** mRNA; this means that mutating even one mRNA can increase the FRP by up to 0.25% or decrease it by 1% in average, for both algorithms. Thus, as can be seen in Table 1, mutating a small set of specific genes (e.g., two most relevant genes) can have a large FRP change by more than 10% (see also Fig. 6).

Table 1 summarizes the effect of mutating the 10 top genes, each with 20 mutations optimized by GG with different MRS.

Table 1
The table summarizes the increase/decrease of FRP one modifying only the top 10 genes with the largest effect, each gene with 20 mutations made by the GG algorithm.

MRS (codons)	GG FRP Increase version (%)	GG FRP Decrease version (%)
20	18.46%	14.15%
35	25.95%	15.59%
50	27.15%	16.09%
75	28.65%	17.09%
100	30.50%	18.69%

4.2. Translation rate threshold (TRT) vs. performance

In order to modify traffic jams in the cell without affecting other phenotypes, the algorithms are constrained so the change in translation rate of the mutated mRNA is limited. To maintain the local translation rate, we choose at each iteration the codon that gives the highest change in the ribosome density, *while maintaining protein translation level under a certain threshold rate.* In order to evaluate the effect that the translation rate threshold (TRT) has on the algorithm performance, we evaluate several TRTs: 1, 5, 10, 50 percentage for both algorithm versions.

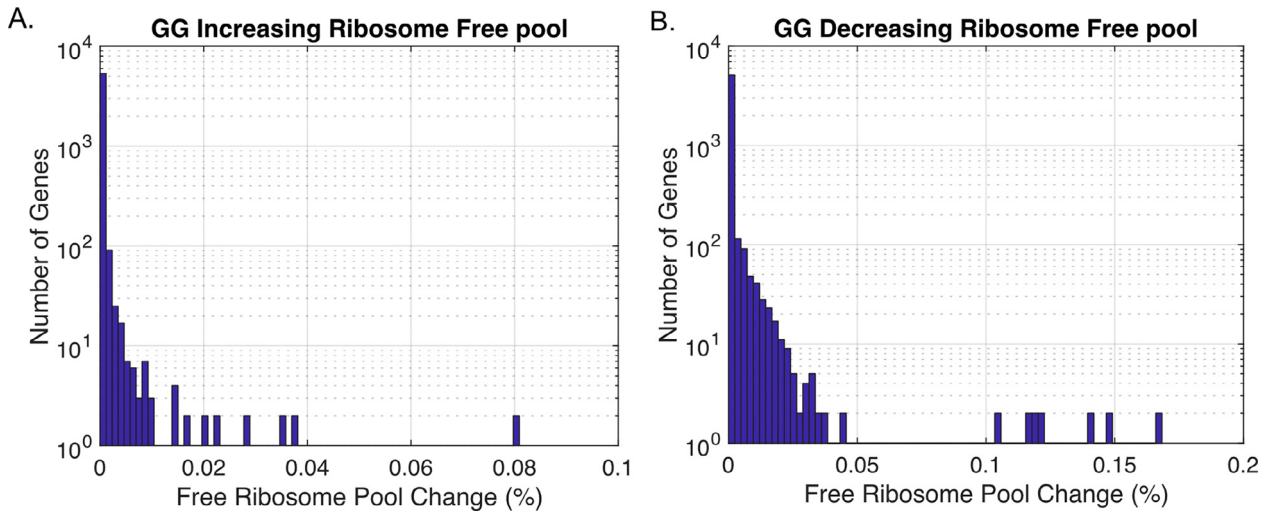


Fig. 6. Distribution of mRNAs as a function of the effect on FRP as a result of the GG version algorithm for both increase (A.) and decrease (B.) pool version. The analysis was performed based on MRS of 50 codons and TRT of 10%.

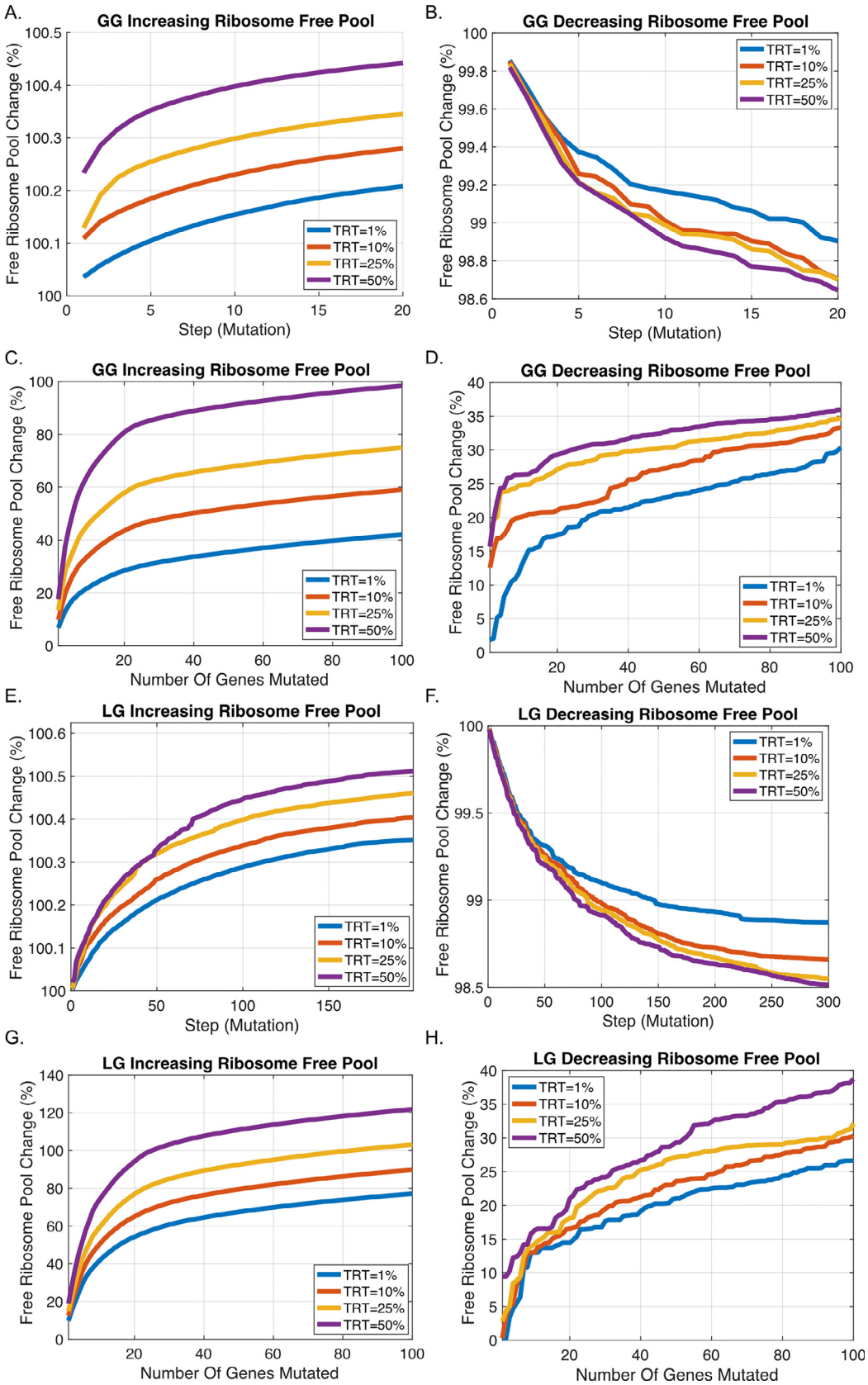


Fig. 7. Impact of TRT level on algorithms performance. The mean effect on the FRP size per gene when additional mutations are introduced (A., B. for increasing and decreasing FRP versions for GG and E., F. for LG) and the effect when additional genes are mutated (C., D. For both versions for GG and G., H. for LG). Each color corresponds to a TRT from the set of 1,10, 25, and 50.

The results here suggest that restricting the translation rate can have a dramatic effect on the performances of the algorithm. Specifically, the increase in the free pool (in comparison to its size when there are no editing) is 60% higher when allowing 50% in the TRT change and mutating 100 genes, in comparison to same number of mutated genes and allowing 1% change in the TRT (Fig. 7C). Thus, a mutation that is introduced during the evolutionary process, or is based on a design which have dramatic effect on the traffic jam will probably have also dramatic effect on the translation rate.

5. The effect of genes expression level on the algorithm's performances

Highly expressed genes tend to consume more ribosome [18]. This may suggest that mutations in such genes will have higher effect on the global ribosomal allocation. On the other hand, these genes tend to undergo stronger evolutionary selection, and this may decrease our ability to optimize them. Thus, in this subsection we aimed at quantifying the typical effect of the expression levels on the algorithms' performances.

Fig. 8 suggests that mRNA expression levels of a gene have dramatic effect on the performances of the algorithm in both cases of an increase or a decrease in the FRP. Specifically, when using the top third of the mRNA in terms of their expression the GG algorithm yields after 20 steps a 210/400 times higher effect on the pool for the increase/decrease version of the algorithm respectively in comparison to the lowest third of the genes in terms of their expression.

The average density is calculated by averaging the mean ribosomal density of each mRNA over all the analyzed genes at each mutation/iteration. We see that although increasing the FRP the RD is reduced by up to 20% more on the lowest third of the genes compared to the top two thirds (Fig. 9), the impact on the FRP is stronger when mutating mRNAs with higher mRNA expression levels (Fig. 8). This result demonstrates that optimizing mRNAs with higher mRNA expression levels has a greater impact even though such genes tend to already have more efficient elongation profile. This can also explain the natural evolutionary tendency to optimize highly expressed genes over lowly expressed (see, for example, [19]).

6. The chosen optimized codon

The GG algorithm optimizes ribosome allocation based on both the **codon translation rate** and **codon location** along the mRNA. In this section we aimed at better understanding the type of codons that are chosen by the algorithms and their location within the ORF.

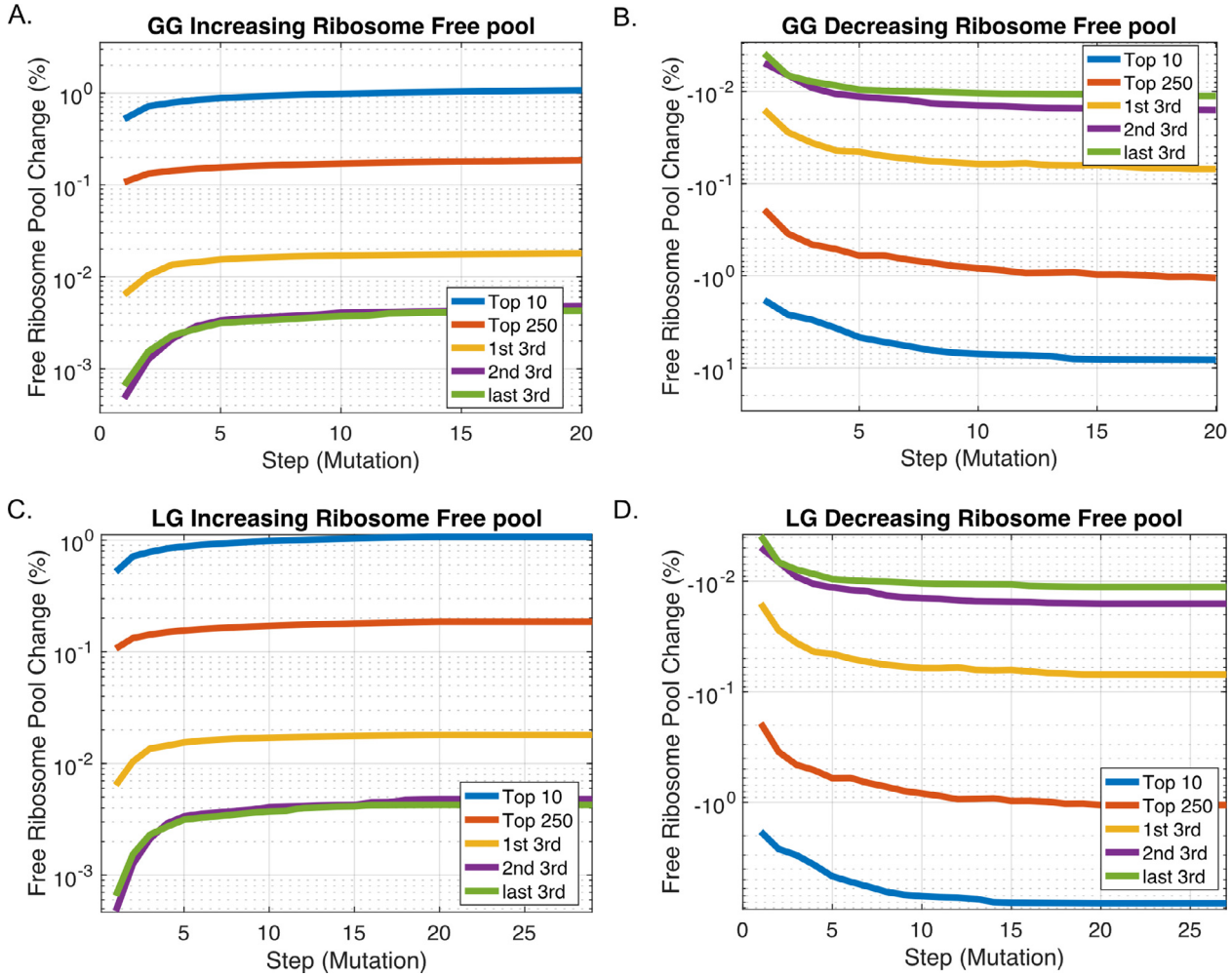


Fig. 8. Impact of mRNA expression levels on algorithms performance. The target mRNAs were divided by their mRNA expression levels: from top 10, top 250 and top third mRNA to last third mRNA levels, the figure shows the relative average FRP change of each group. The graphs include the relative (in %) average FRP at each iteration step for both versions (GG version – A., B., LG version – C., D.). In all the cases, both the GG and the LG algorithms were performed with an MRS of 50 codons and TRT of 10%.

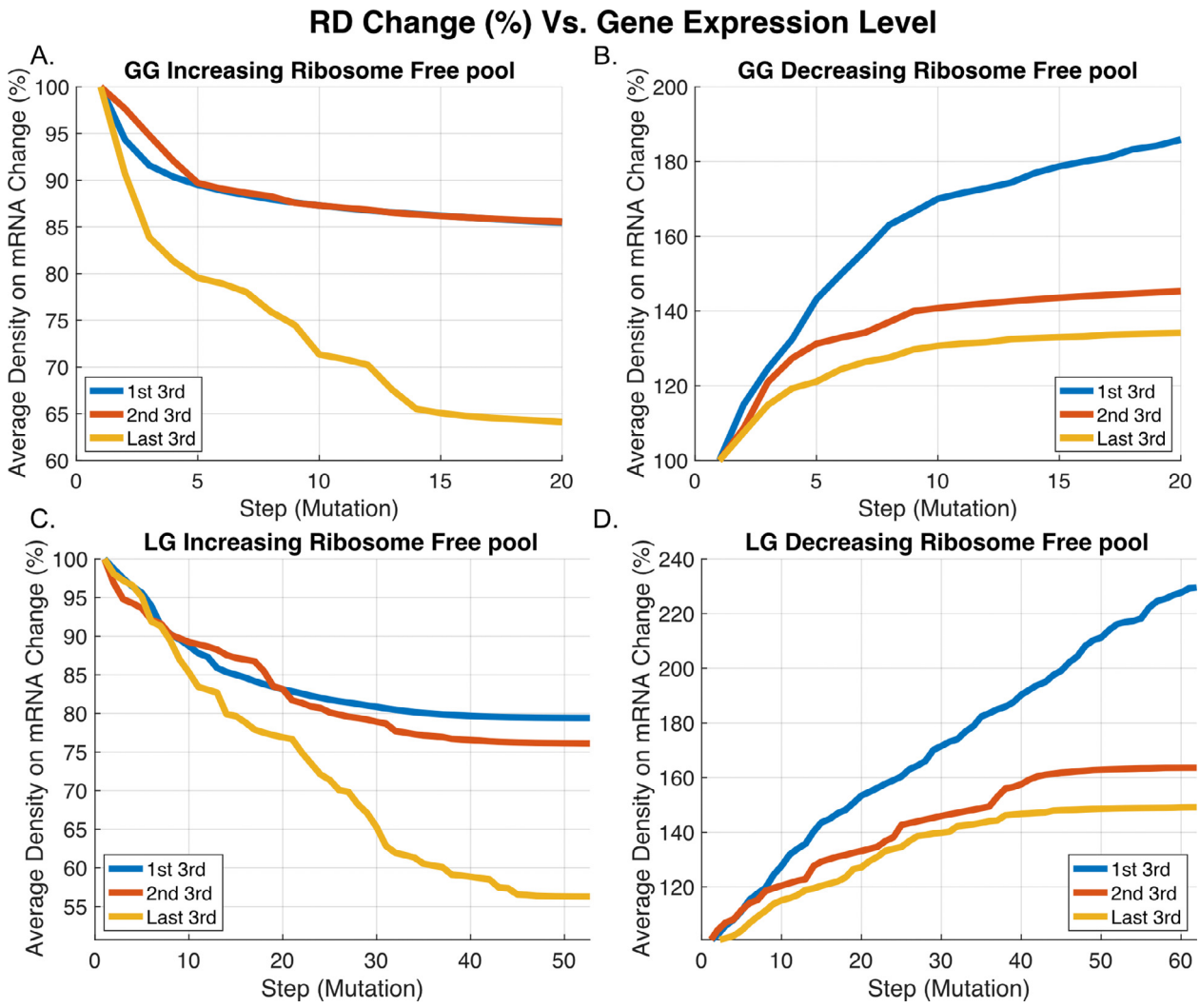


Fig. 9. The average ribosome density (RD) on the target mRNA at each iteration step for both increasing and decreasing FRP versions (GG version – A., B., LG version – C., D.). The target mRNAs were divided by their mRNA expression levels: from top third mRNA levels (blue) to last third mRNA levels (yellow). GG algorithm was performed with an MRS of 50 codons and TRT of 10% on whole set of mRNAs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Special attention is given to the first few codons that are chosen by the algorithm, which are expected to yield the largest impact. We believe that these results may also enrich our understanding of translation evolution.

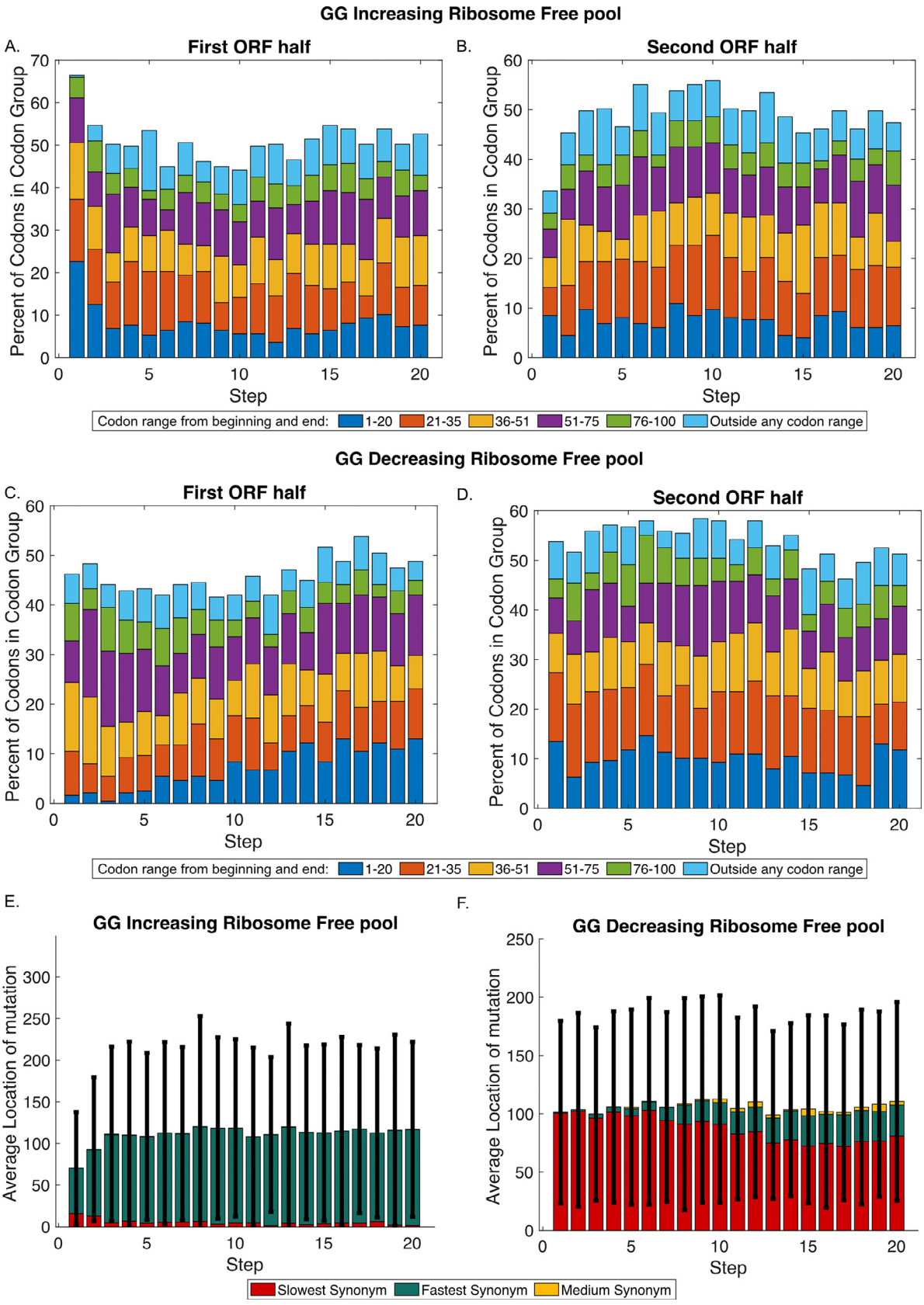
Fig. 10(A and D) demonstrates that when the GG algorithm runs without any MRS (i.e. at each iteration any codon within the ORF can be chosen to be optimized) the vast majority of codons chosen (above 80% percent) are located withing the maximum MRS of 100 codon (both for increasing FRP (Fig. 10A and B) and decreasing FRP (Fig. 10C and D)), and ~50% of codons mutated by the GG algorithm are located with MRS equals to 50 (i.e. the very first and last 50 codons). We note that the average length of the highly expressed genes ORFs is 272 codons, while the average % of mutations that appear in the 50 codons (27%) and the first 100 codons (42%) which are larger than expected (Fig. 10A and C).

When decreasing the FRP, by increasing ribosome density on a target mRNA with the introduction of traffic jams, there is a preference of placing very slow codons at the end of the ORF, however Fig. 10 (bottom right) suggests that as the number of mutations grow, there is also a preference for "fast" codons. As the portion of "fast" codons grows, they tend to be placed at the beginning of the ORF. For increasing the FRP, Fig. 10E suggests that the very first mutations made by the algorithm are placed at the beginning of the ORF, with abundance of "placing" very "slow" codons, suggesting a creation of a "ramp" in the beginning of the ORF in agreement with the ramp theory [7,17].

7. The distinct effect of gene determinants on the algorithm's performances

In order to map the type of mRNAs that would be the most effective to optimize we correlate gene features with the FRP impact of the gene based on our GG algorithm. We consider the following features: 1) the codon adaptation index (CAI) which is a measure of codon usage bias [20] (see details in the Methods section); 2) mRNA level, 3) RD) Ribo-seq read count normalized by the length and mRNA levels of the gene; see details in the Methods section), 4) mRNA length, 5) Ribo-Seq RC (the ribosome sequencing read count, the number of ribosome-reads mapped to coding sequence after normalization by the mRNA length; Methods section)). Fig. 11 depicts both the correlation with each feature and the partial correlation when controlling for all other features.

As can be seen, the mRNA levels have a strong correlation with the impact on the FRP, and as also can be seen in Fig. 8, RD,


GG Increasing Ribosome Free pool
A.
First ORF half
B.
Second ORF half
Percent of Codons in Codon Group
Step
Codon range from beginning and end:
1-20
21-35
36-51
51-75
76-100
Outside any codon range
GG Decreasing Ribosome Free pool
C.
First ORF half
D.
Second ORF half
E.
GG Increasing Ribosome Free pool
Average Location of mutation
F.
GG Decreasing Ribosome Free pool
Slowest Synonym
Fastest Synonym
Medium Synonym

Ribo-seq RC, and CAI also have significant district correlation with the impact on the FRP, albeit a much lower partial correlation. Specifically, the partial correlation of the mRNA length is not significant, suggesting that any trend shown by the Spearman correlation is heavily dependent on the other features (mRNA expression, RD features, CAI, Ribo-Seq RC) that this mRNA length correlates to.

8. Discussion

We analyze algorithms to increase or decrease the cell growth rate that strives to modulate the change in protein levels, with an introduction of silent mutations in a confined region of the mRNA molecule.

We study the effect of various features and algorithmic parameters on the performances of these algorithms. Among others we studied variables such as restricted region size, translation rate threshold range, expression levels, codons used, and running time.

We derive the following conclusions from our analysis. First, we demonstrate that there is a tradeoff between the two analyzed algorithms, GG and LG , in terms of the number of mutations and the algorithm running time: the LG algorithm tend to improve the FRP with significantly lower running time (which is an important resource when running whole cell simulations) but its solutions tend to include significantly more modifications (which many be more expensive to introduce with genome editing tools and may have an unintended effect).

Second, we show that most of the contribution to the FRP decrease or increase can be attributed to the modifications of the first and last 50 codons. Specifically, when we modify only the first and last 50 codons, we can get a solution which is very close to the solution obtained with no MRS. This result supports the ramp theory which suggests that the codons at the beginning of the coding regions are under stronger selection for ribosomal traffic jam control [7,17,21]. It also specifically suggests that the effect of a typical silent mutation during evolution on traffic jams will be much higher when the mutation occurs at the ends of the coding sequence.

Third, our analysis suggests that there are small number of genes where mutations in them can have a dramatic effect on the FRP. They can specifically increase or decrease it by up to 100% and 35%, respectively. These genes are thus expected to be under stronger evolutionary selection pressure.

Forth, our analyses suggest that there is relatively strong coupling between the number of ribosomes and traffic jams on a mRNA and its translation rate. It is challenging to find solutions with strong effect on FRP that have weak effect on translation rate. This may suggest that during evolution genes that undergo evolution to decrease their ribosomal consumption (and thus increase growth rate) may undergo as a side effect a selection for improving their translation rate and thus their protein levels. Thus, this suggests a novel non-direct mechanism of gene expression evolution.

Furthermore, our analyses suggest that mRNA levels of a gene are by far the most relevant predictor regarding the expected effect of its design by our algorithms on FRP. In addition, the RD in the mRNAs and the CAI of the coding sequence also have a minor distinctive effect. Thus, these results demonstrate that while highly expressed genes are already optimized for traffic jam minimization, [7,21–23] additional mutations in them are expected to still have large impact on the organismal fitness. Thus, in this context our results suggest that even the most optimized genes should be under further evolutionary selection for additional translational optimization. Our results may also suggest that highly expressed genes are not in their most optimal state.

Finally, our analysis demonstrates that decreasing traffic jams and increasing the FRP in many cases (up to 22% of the cases) the best solution includes the introduction of the slowest codon; similarly, for increasing traffic jams and decreasing the FRP in many cases (up to 24% of the cases) the best solution includes the introduction of the fastest codon. This also demonstrates the gap between the actual complex solutions for optimal translation elongation profile and (usually too) simplistic indices (see, for example[24]) that assume that all the positions in the coding sequence have the same effect on translation efficiency.

Our study demonstrates for the first time how biological systems can be studied based on the analysis of algorithms intended to optimize them. We demonstrated here that the analyses of the performances of these algorithms not only teach us about how to develop novel tools in synthetic biology; they also provide important insight related to the evolution of genomes and the intracellular biophysical state.

Since our approach can be implemented for any organism, it would be interesting in future research to analyze additional organisms and evaluate if the variability and robustness found here impacts the algorithm based on *S. cerevisiae*. It will be also interesting to perform similar research related to other cellular aspects in addition to translation; this can include other gene expression steps, and in addition other intracellular processes that can undergo optimization.

The greedy algorithms that is described in this study share some properties with the way evolution converge to "solutions": e.g. the search is "greedy" (at each generation the best solution is "kept"), the search is relatively local (i.e. the modification are "small"), and translation efficiency is related to the fitness objective; thus, some of the reported conclusions related to the algorithms themselves may be related directly to the evolutionary processes. However, it is important to also emphasize the fact that there are many differences between our algorithms and evolution: e.g. there are type of mutations (such as indels, and larger aberrations) that are not represented in our algorithms, the objective function is not solely related to translation and can be very complex and non-constant, and the process is stochastic.

The reported analysis is based on the fundamental biophysical aspects of translation which are in general common to all organisms. Thus, we expect that the generally the conclusions reported in our study will be valid for large amount of organisms and cell types; however, there are clearly organismal characteristics that are expected to affect the strength of some of the conclusions. For example, in organisms with low levels of selection for codon

Fig. 10. Mutation locations along the ORF. Result of the GG version algorithm based on 500 highly expressed mRNAs **with no MRS** and a TRT of 10%. Both for the increasing (A., B.) and decreasing (C., D.) FRP versions. A.-D. Each bar is an algorithm step from first to 20th, and each color represents the percent of codons that belong to a **codon range of** the ORF, for both first half (A., C.) and second half (B., D.) of the ORF. For example, when increasing FRP (A.) at the first step ~22% of codons mutated are "chosen" by the algorithm to be located at codon range 21–35 at the first half of the ORF (i.e., first 21–35 codons of the ORF) and (B.) about 8.5% are located at codons 21–35 (top orange) at the end of the ORF (i.e., last codons 21–35 of the ORF). Only around 10–15% of codons "chosen" by algorithm are located outside any MRS (i.e., 85–90% of mutated codons are located inside the first and last 100 codons). E.-F. Each bar is the average codon location that was "chosen" by the algorithm to mutate, at each algorithm step, with the standard deviation of the location represented by the black bar. Red is the portion of times the algorithm "chose" to place slowest synonymous codon, green is for fastest codon, and yellow indicates that medium rate codons are placed. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

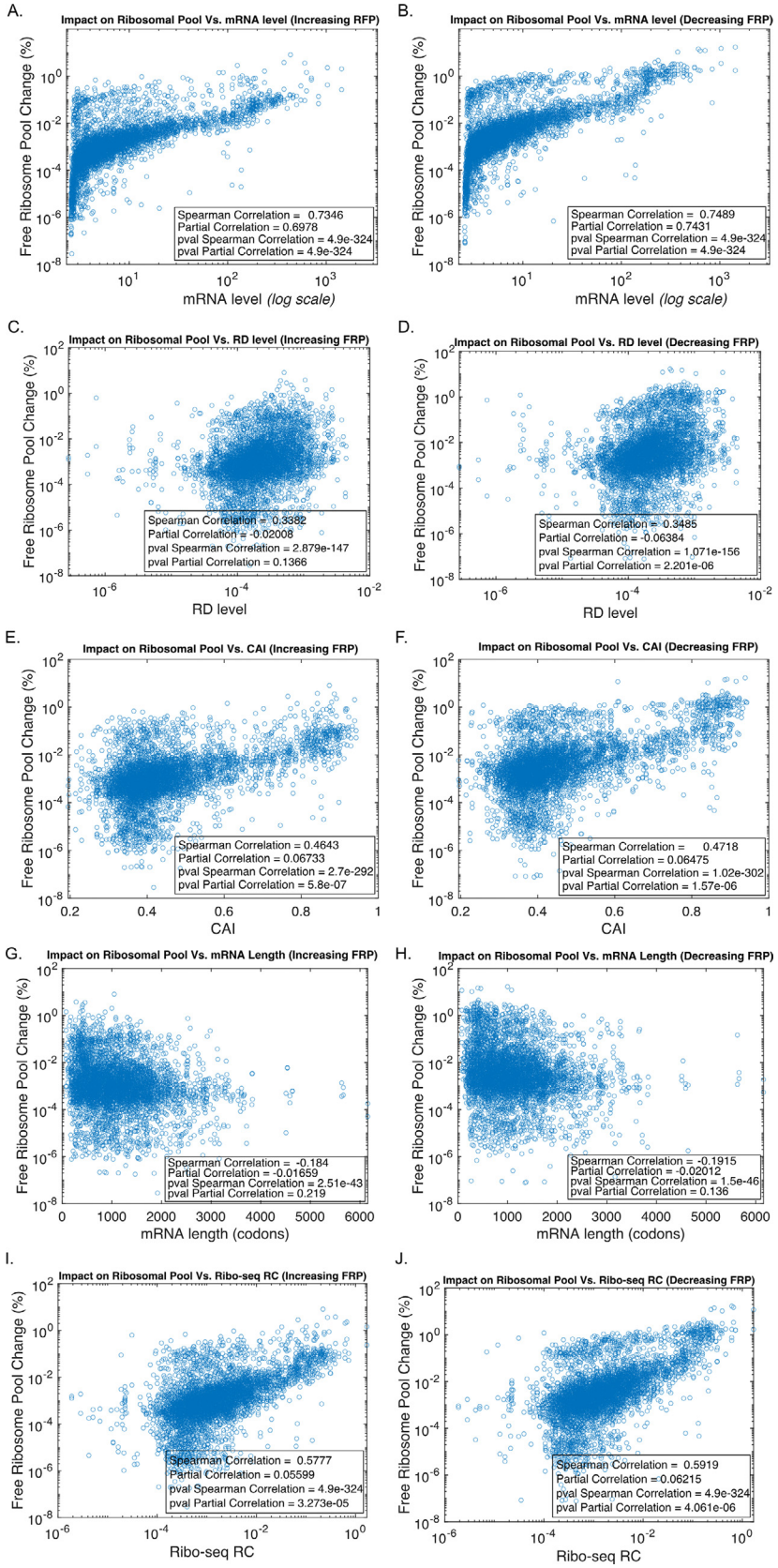


Fig. 11. Correlation between the effect of the gene on the FRP as a function of mRNA features: **mRNA levels** (A., B.), **RD level** (C., D.), **CAI** (E., F.), **mRNA length** (G., H.) and **Ribo-Seq** (I., J.). All the analyses here were performed with the increase pool version of the GG version algorithm with an MRS of 50 codons and TRT of 10%.

usage bias (e.g. mammals) we expect it to have lower correlation with the performances of the models. Similarly, in organisms or conditions where the number of ribosomes is very low (in comparison to the number and the lengths of the mRNAs) we expect that the initiation rate will be relatively very low and thus the amount of traffic jams and the ability to generate them will be much lower. In such cases effect of our algorithms on the FRP will be lower. Another relevant factor is the relative abundances of tRNAs in the cell: if the levels of tRNA are similar (or non rate-limiting) it will be harder to engineer traffic jams as the codon decoding rates are expected to be more uniform; in such cases our algorithms will also be less efficient.

Recent studies have demonstrated that codon bias can affect not only translation elongation but also other aspects such as mRNA stability and transcription efficiency. In this study we modeled and focused only on the effect of codons on translation. The advantage of our analysis is in the fact that it can evaluate and study this specific aspect (it cannot be done experimentally). The disadvantage is in the fact that (as in the case of any other model) in practice the effect of introducing mutations can be different than predicted here; however, it is important to mention that a previous study has demonstrated a correlation between the model predictions and experimental results [15].

It is also important to mention that since it is known that mRNA molecules with more ribosomes are also expected to be more stable [25–27] we expect such mRNAs to have more effect on the performances of our algorithms; thus, we expect, for example, a larger correlation between the RD and the algorithm performances.

Finally, as in the case of other computational studies in the field, it will be interesting to repeat the analysis via experiments. However, this study is based on the statistical analysis of millions of mutations in thousands of genes that allowed as to infer the various statistical relations which are reported in the paper. Thus, relevant experimental follow up should also be based on at least thousands of cases; with the current technologies for genome editing this will be a very ambitious project. This emphasize the advantage of analyzing whole cell simulations for better understanding the effect of mutations on complex intracellular processes and the why they can be used in synthetic biology.

9. Methods

9.1. Ribosomal profiling data

S. cerevisiae ribosomal profiling raw reads (2 replicates) and mRNA levels (2 replicates) were taken from [28].

9.2. Genome assembly

S. cerevisiae genomic data (R64-1-1) was downloaded from BioMart. 5′UTR and 3′UTR annotations (spliced) were obtained from [29]. Genome contains 6664 genes, the number of *S. cerevisiae* ribosomes used in the simulation is 200,000 [11], with 60,000 mRNAs [30], scaled according to the mRNA levels from [28]. According to [30] the number of free ribosomes in the pool is about 15%, i.e. 30,000.

9.3. Whole-cell computational model overview

We used the RFMNP (RFM (Ribosome Flow Model) network with a pool) [12] to model translation, which is a general dynamical model for large-scale simultaneous mRNA translation and competition for ribosomes based on combining several ribosome flow models with input and outputs (RFMIOs) [31], interconnected via a pool of free ribosomes. Each gene is represented by a single copy of a RFMIO. The ribosome flow model (RFM) [31] is a deterministic mathematical model for mRNA translation that can be derived by a mean-field approximation of an important model from statistical physics called the totally asymmetric simple exclusion process (TASEP[32–34]).

The dynamics of the system is expressed by a set of ordinary differential equations that describes the time evolution of the ribosomal densities in the different RFMIOs and the free pool. In the RFM, mRNA molecules are divided into n consecutive sites consisted of codons. The state variable $p_i(t)$ describes the normalized ribosomal occupancy level at site i at time t, where $p_i(t) = 1[p_i(t) = 0]$ indicates that site i is completely full [empty] at time t. The movement of the ribosome that occupies the i-th site to the consecutive site is with rate λ_i. Given the state $p_i(t)$, it follows that the rate of ribosome flow into/out of the system is given by: $\lambda[1 - p_1(t)]$ and $\lambda_n p_n(t)$ respectively. Hence, the rate of ribosome flow from site i to site $i+1$ is given by: $\lambda_i p_i(t)[1 - p_{i+1}(t)]$. Thus, we get the following set of n nonlinear first-order ordinary differential equations that describe the process of translation elongation:

$$\begin{cases} \frac{dp_1(t)}{dt} = \lambda[1 - p_1(t)] - \lambda_1(t)[1 - p_2(t)] \\ \frac{dp_i(t)}{dt} = \lambda_{i-1}p_{i-1}(t)[1 - p_i(t)] - \lambda_i p_i(t)[1 - p_{i+1}(t)]\, 1 < i < n \\ \frac{dp_n(t)}{dt} = \lambda_{n-1}p_{n-1}(t)[1 - p_n(t)] - \lambda_n p_n(t) \end{cases} \quad (1)$$

A network of m RFMIOs interconnected via a pool of free ribosomes models translation while competing for the available, limited ribosomal resource. Competition for the available ribosomal resource leads to indirect coupling between the different mRNAs. For example, if more ribosomes bind to a certain mRNA molecule, then the pool of free ribosomes in the cell is depleted, and this may lead to lower initiation rates in the other mRNA. This is modeled as: $G_j = \lambda_{0j} \tanh\left(\frac{Z}{c}\right)$, where λ_{0j} denotes the initiation rate of gene j, Z denotes the size of the FRP, and c is a parameter of the model fitting local and global aspects of the FRP. The use of *tanh* is appropriate for modelling a saturating function and is a standard function in ASEP (asymmetric exclusion processes) models with a pool, because it is 0 when Z is 0, and for all $Z > 0$ sufficiently small $G_j(Z)$ is linearly proportional to Z.

Briefly (for a detailed description see [31]), the steady state translation rate R is calculated as follows: in steady state the occupation probabilities are constant in time and equal to $\{\pi_1, \cdots, \pi_n\}$, thus

$$R = \lambda_n \pi_n \quad (2)$$

This rate is also equal to the steady state rate at which ribosomes leave the mRNA strand (after translating the entire sequence). At steady state, the left-hand side of equation (1) is a zero, yielding:

$$\begin{cases} \lambda[1 - \pi_1] = \lambda_1\pi_1[1 - \pi_2] = R \\ \lambda_{i-1}\pi_{i-1}[1 - \pi_i] = \lambda_i\pi_i[1 - \pi_{i+1}] = R 1 < i < n \\ \lambda_{n-1}\pi_{n-1}[1 - \pi_n] = \lambda_n\pi_n = R \end{cases} \quad (3)$$

Solving equation 3 for R can be done numerically. Due to the intense time and memory requirements of solving the entire ODE system, we imply a linear-algebraic iterative implementation of RFMNP: A function of the initiation and elongation rates are made to construct a symmetric, non-negative tridiagonal matrix. The square root of the maximal eigenvalue of set matrix is the protein translation rate [35], which is what we use as a proxy of translation rate threshold (TRT). Solving the equations with the linear-algebraic approach rather than the entire ODE system provides a substantial speedup. We performed an iterative implementation of RFMNP to ensure steady state:

Let H denote the number of ribosomes in the system, Z the number of free ribosomes, (H and Z are determined according to the literature, see above), M_j the number of mRNA copies of gene j, and x_i^j the number of ribosomes on site i in gene j. In steady state

$$H = Z + \left(\sum_j \sum_i x_i^j \cdot M_j\right)$$

The RFMNP iterations are performed as followed.

In the first iteration we set Z_0 to be equal to the literature free pool. The size of the free pool Z_0 is a factor of the initiation rate: for each gene j, the initiation rate $G_{0j} = \lambda_{0j} \tanh\left(\frac{Z_0}{c}\right)$, where λ_{0j} is the estimated local initiation rate, and $\tanh\left(\frac{Z_0}{c}\right)$ is the global initiation rate of gene j. The RFM model is calculated for every gene separately until convergence. The sum of all ribosomes occupying all of the genes, removed from H gives us: $Z_1 = H - \left(\sum_j \sum_i x_i^j(0) \cdot M_j\right)$.

At the k' th iteration: $Z_k = H - \left(\sum_j \sum_i x_i^j(k-1) \cdot M_j\right)$, with $k-1$ being the density of the previous iteration as input to the current. A binary search is performed on the range [Z_k, Z_{k-1}], updating $Z_k = Z_k - (Z_k - Z_{k-1}) * per$, where per is in [0.75, 0.95], depending on the convergence rate of the iterations: starting off with $per = 0.75$, in case of divergence, we reduce the step size while increasing per. We then update the initiation rates with the found Z_k: $G_{kj} = \lambda_{0j} \tanh\left(\frac{Z_k}{c}\right)$.

The iterations are terminated if $abs(Z_{k-1} - Z_k) < \varepsilon$, with ε being 10^{-2}.

9.4. Whole-cell model parameters

The RFMNP has three parameters which need to be estimated: initiation rates, codon elongation rates, and c. These parameters were estimated using ribosome profile data and an iterative algorithm. For a detailed description see [15].

Briefly, the initial initiation rates were estimated based on the measured ribosomal read count divided by the mRNA levels and then normalized so that the median codon initiation rate of all *S. cerevisiae* mRNAs becomes 0.8 per second [36].

Codon elongation rates were calculated based on the tRNA Adaptation Index (tAI[23]), where at the base of the index is the observation that elongation rate associated with a codon is proportional to the abundance of the tRNA species that recognize it, with regards to all anti-codons that can recognize the same codon, with different efficiency weights. Let n_i denote the number of tRNA isoacceptors recognizing codon i. Let $tCGNij$ denote the copy number of the j th tRNA that recognizes the i th codon and let S_{ij} be a parameter corresponding to the efficiency of the codon-anticodon coupling between codon i and tRNA j.

The absolute adaptiveness, W_i, for each codon i is defined by

$$W_i = \sum_{j=1}^{n_i} (1 - S_{ij}) tCGN_{ij}$$

From W_i we obtain p_i, which is the probability that a tRNA will be coupled to the codon, i.e.

$p_i = \frac{W_i}{\sum_{j=1}^{61} tCGN_j}$. The expected time on codon i is $t_i = \frac{1}{p_i}$.

As for the RFM, the input rate is per site. The RFM is divided into sites consisting of 10 consecutive codons (the approximate size of the *S. cerevisiae* ribosome[37]). If the remainder after the last site is five codons or less, it is incorporated in the previous site, thus avoiding extremely fast site due to short length sites. The rate of each site is the inverse of the summation of the times of each of the codons on set site, i.e., $\lambda_n = \frac{1}{\sum_{i=0}^{10} t_i}$.

10. The hill climbing algorithm

This is an iterative algorithm that starts with an arbitrary solution to a target function, then attempts to find a better solution by making an incremental change to the previous solution. If the change produces a better solution, another incremental change is made to the new solution, and so on until no further improvements can be made.

In our case the target function is the RFM, the solution we strive to optimize is the ribosome density on the target mRNA (either increase or decrease). An iteration step is defined as follows.

Choosing the current codon location, i.e., the location of the codon in the gene, is defined by the first nucleotide (nt) of a codon, for example, if the second codon is mutated, its location within a gene would be the fourth nt. Then we mutate a codon at its location, to each of its synonymous codons in the target mRNA. A WCS is then calculated (by RFM) with each of the synonymous codons. In case of an improved solution (increase/decrease in ribosome density on the target mRNA) we update the target sequence to the new codon. We run the algorithm in two similar version, LG version, and GG version, defined by the chosen codons to mutate.

The input to the target function (RFM) are the λ s of the target sequence, each synonymous codon will yield a different λ in its RFM site (see codon rate). The output of the function is the ribosome density profile.

10.1. Local greedy version (LG)

In this version a step is defined by randomly choosing a codon location, and for each of the synonymous codons we calculate the RFM function. If the synonymous codon produces a better result, we update the sequence to the synonym. We run enough iterations to choose each possible location, and the algorithm stops when no further increase/decrease in ribosome density can be found.

10.2. Global greedy version (GG)

This is similar to the LG version, but with a key difference: at each iteration at every codon location, all the synonymous codons are introduced to the RFM, and the codon that produces the largest change in the ribosome density is chosen. The algorithm stops when no change at any location yields an improvement.

10.3. Algorithm constrains

We strive to control the cell ribosome resources with a minimal impact on other cells translation processes. We divide and test these restrictions into a few categories:

One such aspect is the encoded amino acid sequence of the proteins, meaning all the mutations must be silent.

10.4. Translation rate threshold (TRT)

One reason to minimize the impact on whole cell is due to cases where extreme changes in the protein levels of some proteins can be destructive to the cell. We impose a protein translation change threshold, i.e., we choose at each iteration the codon that gives us the largest change in the ribosome density, while maintaining the target mRNA's protein translation level under a certain threshold rate τ. Increasing τ is expected to improve the performances of the algorithm in terms of the increase in the FRP. We evaluated a number of translation rate thresholds (*1,10, 25, 50%*) for both increasing and decreasing the FRP. Our aim of using a threshold τ is to change only the ribosomal traffic jams without affecting

other aspects that we do not know how to model and/or that are not included in our model.

10.5. Modified region size

Initiation & termination regions

Any mutations in translation initiation and termination signals regions can yield a large and unforeseeable impact to the translation process, which we want to avoid. It is known that the first and last codons of the ORF (15 and 10 codons long, respectively) hold important regulatory signals [17]. Thus, the algorithm will not allow any mutations in these regions.

10.6. Expected MRS

Considering both the computational task and future genome editing we want to confine the silent mutations the algorithm makes to a specific region in the mRNA. One of our goals is to find the location of this optimal region and its size.

Since the biggest impact of the RFM translation rate is in the beginning and end of the mRNA in endogenous genes [38–39], we will impose areas in the beginning and end of the mRNA that allow modifications only in these regions. We evaluated five region sizes (20, 35, 50, 75, 100 codons) from the beginning and end of the ORF where mutations are allowed (omitting the very beginning and end as set above).

This is done as long as it does not reduce or increase the gene's translation rate (TR) beyond a threshold τ.

11. Algorithm execution time

Computational simulation of the whole cell translation model with an order of magnitude of $2 * 10^5$ ribosomes and $6 * 10^4$ mRNA molecules is a challenging time-consuming effort. Therefore, reducing the number of WCS it takes to evaluate a mutation will considerably reduce the running time to optimize target genes. We estimate the cost of reducing running time and iteration steps has on the algorithm performance by comparing the average performance of the GG version and LG version algorithms, as a function of running time. The running time is calculated based on the number of times each algorithm performs a WCS, and the performance is measured in average FRP:

1. GG version: at every iteration one mutation is introduced, meaning to make one mutation the algorithm runs over the codons of the whole sequence (under given restrictions), and for every codon WCS is calculated for each of its synonymous codons:

$$\textit{number of WCS for 1 mutation} = \textit{number of codons in sequence} * \textit{number of synonyms of each codon}$$

The *number of codons in sequence* varies: if we run MRS of 50 we will consider 100 codons per iterations (an average of 300 WCS per iteration), however without an MRS we will consider all the codons in the mRNA and this can result order of magnitude 10^3 WCS for one mutation.

2. LG version: at each iteration only one codon is chosen randomly and WCS are ran for each of its synonymous codons, meaning: $average\ number\ of\ WCS\ for\ 1\ iteration = 3$. Since in the LG version not all iterations result in a mutation, it may take several iterations for one mutation to be made.

12. Gene determinants

The Codon Adaptation Index (CAI) [20] is a technique for analyzing codon usage bias. The CAI measures the deviation of a given protein coding gene sequence with respect to a reference set of genes. We used as a reference set the top 5% expressed mRNAs. The CAI is defined as the geometric mean of the weight associated to each codon over the length of the gene sequence (measured in codons):

$$CAI = \exp\left(\frac{1}{L}\sum_{l=1}^{L} ln(w_i(l))\right) \quad (1)$$

For each amino acid, the weight of each of its codons, in CAI, is computed as the ratio between the observed frequency of the codon (f_i) and the frequency of the synonymous codon (f_j), derived from the reference set, for that amino acid, i.e. $w_i = \frac{f_i}{\max(f_j)}, ij \in [synonymous\ codons\ for\ amino\ acid]$

The Ribo-Seq RC are the measured number of ribosome read counts devided by the mRNA length (Ribo-Seq measurements described above), while the RD is the *ribo*-seq read count normalized by the length and mRNA levels of the gene.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partially supported by a grant from the: THE ELA KODESZ INSTITUTE FOR MEDICAL PHYSICS AND ENGINEERING. The work is also supported by grants from the Israeli Ministry of Science, Technology and Space. We thank Dr. Yoram Zarai for helpful comments.

References

[1] Lane N, Martin W. The energetics of genome complexity. Nature 2010;467(7318):929–34.
[2] Buttgereit F, Brand MD. A hierarchy of ATP-consuming processes in mammalian cells. Biochem. J. 1995;312:163–7.
[3] Mahalik S, Sharma AK, Mukherjee KJ. Genome engineering for improved recombinant protein expression in escherichia coli. Microb. Cell Fact. 2014;13.
[4] Gorochowski TE, Avcilar-Kucukgoze I, Bovenberg RAL, Roubos JA, Ignatova Z. A Minimal Model of Ribosome Allocation Dynamics Captures Trade-offs in Expression between Endogenous and Synthetic Genes. ACS Synth. Biol. 2016;5:710–20.
[5] Gustafsson C, Minshull J, Govindarajan S, Ness J, Villalobos A, Welch M. Engineering genes for predictable protein expression. Protein Expr. Purif. 2012;83(1):37–46.
[6] Boyle, J. Molecular biology of the cell, 5th edition by B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Biochem. Mol. Biol. Educ.* **36**, 317–318 (2008).
[7] Tuller T, Carmi A, Vestsigian K, Navon S, Dorfan Y, Zaborske J, et al. An evolutionarily conserved mechanism for controlling the efficiency of protein translation. Cell 2010;141(2):344–54.
[8] Bahir I, Fromer M, Prat Y, Linial M. Viral adaptation to host: a proteome-based analysis of codon usage and amino acid preferences. Mol. Syst. Biol. 2009;5:311.
[9] Gustafsson, C., Govindarajan, S. & Minshull, J. *Codon bias and heterologous protein expression. Trends in Biotechnology* vol. 22 346–353 (Elsevier Current Trends, 2004).
[10] Shachrai I, Zaslaver A, Alon U, Dekel E. Cost of Unneeded Proteins in E. coli Is Reduced after Several Generations in Exponential Growth. Mol. Cell 2010;38:758–67.
[11] Warner JR. The economics of ribosome biosynthesis in yeast. Trends Biochem Sci 1999;24(11):437–40.
[12] Raveh A, Margaliot M, Sontag ED, Tuller T. A model for competition for ribosomes in the cell. J. R. Soc. Interface 2016;13(116):20151062. https://doi.org/10.1098/rsif.2015.1062.
[13] Kudla G, Murray AW, Tollervey D, Plotkin JB. Coding-sequence determinants of gene expression in Escherichia coli. Science 2009;324(5924):255–8.
[14] Gyorgy, A. & Del Vecchio, D. Limitations and trade-offs in gene expression due to competition for shared cellular resources. *Proc. IEEE Conf. Decis. Control* **2015-February**, 5431–5436 (2014).

[15] Zur H, Cohen-Kupiec R, Vinokour S, Tuller T. Algorithms for ribosome traffic engineering and their potential in improving host cells' titer and growth rate. Sci. Rep. 2020;10:21202.
[16] Sauna ZE, Kimchi-Sarfaty C. Understanding the contribution of synonymous mutations to human disease. Nat. Rev. Genet. 2011;12:683–91.
[17] Tuller T, Zur H. Multiple roles of the coding sequence 5' end in gene expression regulation. Nucleic Acids Res. 2015;43:13–28.
[18] Ingolia NT, Ghaemmaghami S, Newman JRS, Weissman JS. Genome-wide analysis in vivo of translation with nucleotide resolution using ribosome profiling. Science (80-.) 2009;324:218–23.
[19] Bennetzen, J. L. & Hall, B. D. Codon selection in yeast. *J. Biol. Chem.* (1982).
[20] Sharp PM, Li WH. The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. Nucleic Acids Res. 1987;15:1281–95.
[21] Tuller T, Waldman YY, Kupiec M, Ruppin E. Translation efficiency is determined by both codon bias and folding energy. Proc. Natl. Acad. Sci. U. S. A. 2010;107:3645–50.
[22] Kudla G, Murray AW, Tollervey D, Plotkin JB. Coding-sequence determinants of expression in escherichia coli. Science (80-.) 2009;324:255–8.
[23] dos Reis M, Savva R, Wernisch L. Solving the riddle of codon usage preferences: a test for translational selection. Nucleic Acids Res. 2004;32:5036–44.
[24] Bahiri-Elitzur S, Tuller T. Codon-based indices for modeling gene expression and transcript evolution. Comput Struct Biotechnol J 2021;19:2646–63.
[25] Bazzini AA et al. Codon identity regulates mRNA stability and translation efficiency during the maternal-to-zygotic transition. EMBO J. 2016;35:2087.
[26] Presnyak V, Alhusaini N, Chen Y-H, Martin S, Morris N, Kline N, et al. Codon Optimality Is a Major Determinant of mRNA Stability. Cell 2015;160 (6):1111–24.
[27] Edri S, Tuller T, Jang SK. Quantifying the Effect of Ribosomal Density on mRNA Stability. PLoS ONE 2014;9(7):e102308. https://doi.org/10.1371/journal.pone.0102308.
[28] McManus CJ, May GE, Spealman P, Shteyman A. Ribosome profiling reveals post-transcriptional buffering of divergent gene expression in yeast. Genome Res. 2014;24:422–30.
[29] Kinsella, R. J. *et al.* Ensembl BioMarts: A hub for data retrieval across taxonomic space. *Database* **2011**, (2011).
[30] Zenklusen D, Larson DR, Singer RH. Single-RNA counting reveals alternative modes of gene expression in yeast. Nat. Struct. Mol. Biol. 2008;15:1263–71.
[31] Reuveni S, Meilijson I, Kupiec M, Ruppin E, Tuller T. Genome-scale analysis of translation elongation with a ribosome flow model. PLoS Comput. Biol. 2011;7:1002127.
[32] Zia RKP, Dong JJ, Schmittmann B. Modeling Translation in Protein Synthesis with TASEP: A Tutorial and Recent Developments. J. Stat. Phys. 2011;144:405–28.
[33] Lb S, Rk Z, Kh L. Totally asymmetric exclusion process with extended objects: a model for protein synthesis. Phys. Rev. E. Stat. Nonlin. Soft Matter Phys. 2003;68:17.
[34] Schadschneider A, Chowdhury D, Nishinari K. Stochastic Transport in Complex Systems. *Stochastic Transport in Complex Systems* (Elsevier 2011. https://doi.org/10.1016/C2009-0-16900-3.
[35] Poker G, Zarai Y, Margaliot M, Tuller T. Maximizing protein translation rate in the non-homogeneous ribosome flow model: a convex optimization approach. J. R. Soc. Interface 2014;11(100):20140713. https://doi.org/10.1098/rsif.2014.0713.
[36] Chu D et al. Translation elongation can control translation initiation on eukaryotic mRNAs. EMBO J. 2014;33:21–34.
[37] Wolin SL, Walter P. Ribosome pausing and stacking during translation of a eukaryotic mRNA. EMBO J. 1988;7:3559–69.
[38] Levin D, Tuller T. Genome-Scale Analysis of Perturbations in Translation Elongation Based on a Computational Model. Sci. Rep. 2018;8(1). https://doi.org/10.1038/s41598-018-34496-3.
[39] Poker G, Margaliot M, Tuller T. Sensitivity of mRNA translation. *Sci. Rep.* 2015;5 (1). https://doi.org/10.1038/srep12795.