Bioinformatics, YYYY, 0–0
doi: 10.1093/bioinformatics/xxxxx
Advance Access Publication Date: DD Month YYYY
Original Paper

Subject Section

ChimeraUGEM: unsupervised gene expression modeling in any given organism

Alon Diament[1#], Iddo Weiner[1,2#], Noam Shahar[2#], Shira Landman[2], Yael Feldman[2], Shimshi Atar[1], Meital Avitan[1,2], Shira Schweitzer[2], Iftach Yacoby[2*], and Tamir Tuller[1,3*]

[1] Department of Biomedical Engineering, The Iby and Aladar Fleischman Faculty of Engineering, Tel Aviv University, Tel Aviv 6997801, Israel, [2] School of Plant Sciences and Food Security, The George S. Wise Faculty of Life Sciences, Tel Aviv University, Tel Aviv 6997801, Israel, [3] The Sagol School of Neuroscience, Tel Aviv University, Tel Aviv 6997801, Israel

*To whom correspondence should be addressed. #These authors contributed equally to this work.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Regulation of the amount of protein that is synthesized from genes has proved to be a serious challenge in terms of analysis and prediction, and in terms of engineering and optimization, due to the large diversity in expression machinery across species.

Results: To address this challenge, we developed a methodology and a software tool (ChimeraUGEM) for predicting gene expression as well as adapting the coding sequence of a target gene to any host organism. We demonstrate these methods by predicting protein levels in 7 organisms, in 7 human tissues, and by increasing *in vivo* the expression of a synthetic gene up to 26-fold in the single-cell green alga *C. reinhardtii*. The underlying model is designed to capture sequence patterns and regulatory signals with minimal prior knowledge on the host organism and can be applied to a multitude of species and applications.

Availability: Source code (MATLAB, C) and binaries are freely available for download for non-commercial use at http://www.cs.tau.ac.il/~tamirtul/ChimeraUGEM, and supported on macOS, Linux, and Windows.

Contact: tamirtul@.tauex.tau.ac.il (TT) Orcid: 0000-0003-4194-7068 and iftachy@tauex.tau.ac.il (IY) Orcid: 0000-0003-0177-0624

Supplementary information: Supplementary methods and figures are available at Bioinformatics online. Additional documentation is available online at http://www.cs.tau.ac.il/~tamirtul/ChimeraUGEM.

1 Introduction

Analysis and engineering of gene expression is at the core of the understating of various biomedical topics (Alberts *et al.*, 2005), and the development and synthesis of many biomedical and biotechnological products, such as chemicals, metabolites, drugs and vaccines (Wurm, 2004; Terpe, 2006; Ferrer-Miralles *et al.*, 2009; Demain and Vaishnav, 2009; Frenzel *et al.*, 2013). A number of tools for modeling expression based on codon usage measures, such as the Codon Adaptation Index (CAI) (Sharp and Li, 1987), have been developed in recent years (Peden, 2000; Wu *et al.*, 2005; Puigbò *et al.*, 2008; Gaspar *et al.*, 2012).

However, this approach neglects many additional coding sequence-related factors that may affect gene expression regulation, such as tRNA availability (Reis *et al.*, 2004; Welch *et al.*, 2009; Tuller *et al.*, 2011; Dana and Tuller, 2014b), mRNA structure (Kudla *et al.*, 2009; Tuller, Waldman, *et al.*, 2010; Goodman *et al.*, 2013), translation initiation (Kozak, 1999; Zur and Tuller, 2013; Chu *et al.*, 2014; Tuller and Zur, 2015), ribosomal traffic (Tuller, Carmi, *et al.*, 2010), co-translational folding (Kimchi-Sarfaty *et al.*, 2007; Kramer *et al.*, 2009; Zhang *et al.*, 2009), transcription factor binding sites (Stergachis *et al.*, 2013), transcription elongation speed (Churchman and Weissman, 2011; Xia, 1996; Cohen *et al.*, 2018), splicing signals (Barash *et al.*, 2010; Zafrir and Tuller, 2015; Weiner *et al.*, 2018), and ribosome stalling motifs (Stadler and Fire, 2011; Li *et al.*, 2012; Sabi and Tuller, 2015, 2017), to name

a few. Typically, the 'codes' of these different mechanisms are not reflected in a modular manner and codes of various different mechanisms may appear in the same region (see, for example, (Tuller and Zur, 2015)); in addition, these gene expression codes are organism-specific and novel mechanisms of genes expression regulation are frequently discovered even in well-studied model organisms (Yordanova *et al.*, 2018). Thus, a generic model-based approach that captures the biophysics of all these aspects does not exist and developing a unified model for all currently engineered organisms is currently far from being feasible.

Recently, an unsupervised, high-dimensional and model-free approach for analyzing the coding sequence has been proposed and named Chimera (Zur and Tuller, 2015). Chimera has been successfully applied to gene expression prediction based on the average repetitive substring statistic (ChimeraARS) (Zur and Tuller, 2015; Ben-Yehezkel *et al.*, 2015; Zafrir and Tuller, 2017), and to engineering (ChimeraMap) (Weiner *et al.*, 2018). Here, we significantly extend the two algorithms for analysis and design, provide an efficient and accessible software tool based on this approach – ChimeraUGEM (unsupervised gene expression modeling) – and experimentally demonstrate its applicability.

2 Methods

2.1 Chimera algorithms

2.1.1 ChimeraARS

The ChimeraARS (cARS) model measures the average repetitive substring (ARS) length in a given sequence. That is, for each position in the sequence it detects the longest substring that starts at that position and also appears in a set of reference sequences and returns the average substring length across all positions (**Algorithm S1**). The measure assumes that if longer repetitive substrings tend to appear in the sequence, this suggests that it is has evolved to become more optimized to the organism's gene expression machinery, and thus it is probably more highly expressed (Zur and Tuller, 2015). Hence, the Chimera approach is aimed at dealing with all the challenges mentioned above by analyzing the distribution of sub-sequences (and consequently the various codes) that appear in the host coding regions in an unsupervised manner and without prior biophysical models.

Specifically, the implementation here is based on suffix arrays, so that the longest substrings can be searched for all positions in the target in $O(|T|^2 \log|R|)$ time ($|T|$ being the length of the target protein, and $|R|$ being the total length of the reference sequences). The algorithm is alphabet-agnostic, however for the purpose of analyzing coding sequences, three alphabets are commonly used: nucleotides, amino acids, and codons (64 nucleotide triplets).

2.1.2 ChimeraMap

The ChimeraMap (cMap) algorithm adapts a target gene to a host according to the same principles, by encoding its amino acids using synonymous codon blocks that appear in endogenous host genes. Moreover, the algorithm minimizes the number of blocks that are required to construct the sequence (Zur and Tuller, 2015). This strategy can be applied to any sequenced host genome, omitting the need for any additional information other than gene annotations.

The ChimeraMap algorithm searches for the largest common prefix between the amino acid sequence of the target protein and all proteins in the reference sequence. Then, the corresponding nucleotide sequence (encoding the same amino acid substring) is copied from the reference to encode a block in the target protein. The search is repeated for the suffix beginning at the end of the last found block of the target amino acid sequence, until the complete target protein is encoded. When a substring appears multiple times in the reference, the most frequent nucleotide substring is used (**Algorithm S2**). Zur and Tuller have shown that the ChimeraMap algorithm can be computed in linear time using suffix trees (Zur and Tuller, 2015). Specifically, the implementation here is based on suffix arrays, which can be searched in $O(|T|\log|R|)$ time.

2.2 Position-specific Chimera algorithms

Some regulatory signals are expected to appear in specific regions of the gene. For example, initiation-related signals will tend to appear in its 5'-end, as well as codes that may relate to ribosome traffic regulation, mRNA transport and early stages of protein folding (Tuller and Zur, 2015); termination-related codes will tend to appear in the 3'-end of the gene, signals related to splicing or co-translational folding may also appear at certain distances from the ends, etc. We therefore propose the Position-Specific ChimeraARS (PScARS) and Position-Specific ChimeraMap (PScMap), that consider both the size of the detected substring and its location within host genes.

The algorithm parameters include a search window that defines the maximal allowed distance between the position in the analyzed gene, and the position of the substring in the host (measured from the 5'- and 3'-ends). The selected substring is the longest that meets these constraints. **Figure 1** demonstrates the procedure for the PScMap algorithm, however the depicted substring search method is common to both PScMap and PScARS.

Note, that the longest found substrings must begin within the search window, but may extend outside of the window bounds. A bounded variant of the algorithm is given in **Algorithm S3**.

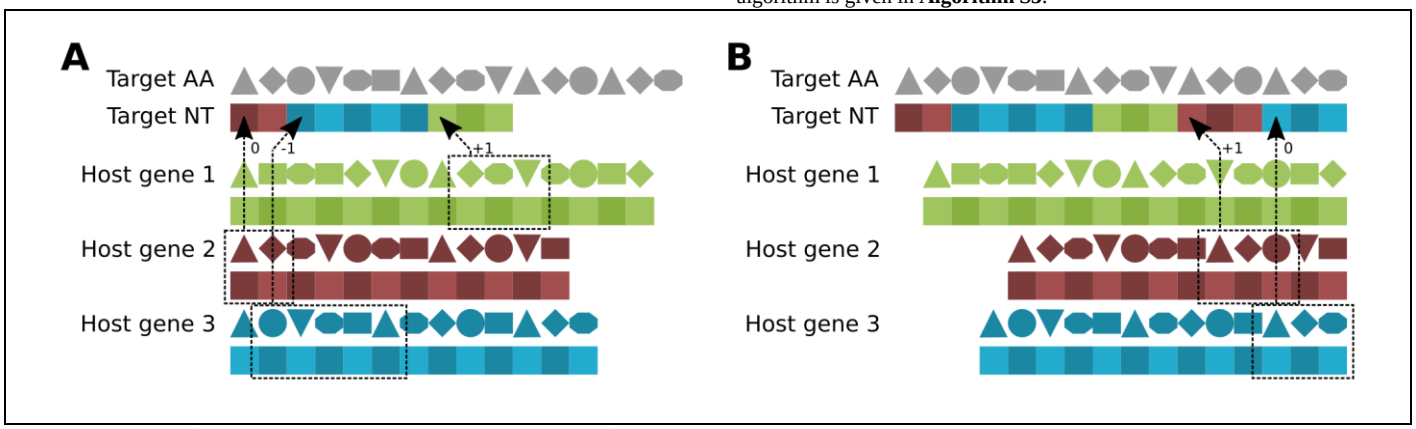

Fig. 1: Position-Specific ChimeraMap. (A) Codon blocks (colored squares) from the host genome are selected to encode the target. In this example, the selected blocks are the largest ones that are within a search window of 3 codons around the location of insertion into the target protein (distance from position denoted next to each arrow). For example, the first block is selected from the 5'-end of gene 2 although a longer block exists downstream the same gene and outside of the search window. **(B)** Backward search diagram, where the search window is positioned in relation to the 3'-end. The diagram depicts the completion of the optimization from panel (A), where forward search may select sub-optimal (smaller) blocks.

2.2.1 Position-Specific ChimeraARS

The PScARS algorithm (**Algorithm 1**) operates similarly to the cARS algorithm described in the section 2.1.1, with the exception that in each iteration the complete suffix array is first reduced to suffixes that start within the search window in their respective genes. The search window is defined based on the current position in the target protein and can be computed in relation to the beginning of the sequence (Figure 1A) and / or the ending of the sequence (Figure 1B). Reduction of the suffix array is calculated in O(log|R|) using an array denoting the order of sorted suffix positions in relation to the beginning of their respective genes (and an additional array, where positions were calculated in relation to the ending of respective genes). Positions in relation to gene start are defined by the series $i = 1,2,\dots,|g|$ where 1 denotes the first position, while positions in relation to gene end are defined by the series $j = -|g|, -|g|+1, \dots, -1$ where –1 denotes the last position in the gene.

Algorithm 1: Position-Specific ChimeraARS

```
Procedure: PScARS
Input:   target     # target sequence (any alphabet)
         suf_arr    # suffix array for the reference sequences
         pos_arr_s  # suffix position array, relative to gene start
         pos_arr_e  # suffix position array, relative to gene end
         ref        # reference sequences (same alphabet as target)
         window     # search window definition
Output:  ARS
n ← size(target)
subs_lens ← array[n]
for pos from 1 to n:
   suf_arr_slice ← SelectWindow(suf_arr, pos_arr_s, pos_arr_e, window,
                                pos, pos-n-1)
   block ← LongestPrefix(target[pos:n], suf_arr_slice, ref)
   subs_lens[pos] ← size(block)

ARS ← mean(subs_lens)
```

```
Procedure: SelectWindow
Input:   suf_arr, pos_arr_s, pos_arr_e, win
         s,e  # current position in relation to target protein start/end
Output:  suf_arr_slice
valid ← empty set
win_coo ← [0, 0]  # window coordinates
if win.from_start
   # update window coordinates relative to gene start
   win_coo[1] ← ceil(s + win.center - win.size/2)
   win_coo[2] ← ceil(s + win.center + win.size/2 – 1)
   L ← BinarySearch(win_coo[1], pos_arr_s, suf_arr)
   R ← BinarySearch(win_coo[2], pos_arr_s, suf_arr) - 1
   add pos_arr_s[L:R] to valid
if win.from_end
   # update window coordinates relative to gene end
   win_coo[1] ← ceil(e + win.center – win.size/2)
   win_coo[2] ← ceil(e + win.center + win.size/2 – 1)
   L ← BinarySearch(win_coo[1], pos_arr_e, suf_arr)
   R ← BinarySearch(win_coo[2], pos_arr_e, suf_arr) - 1
   add pos_arr_e[L:R] to valid

suf_arr_slice ← suf_arr[valid]
```

```
Procedure: LongestPrefix
Input:   key, suf_arr, ref
Output:  block_aa
idx ← BinarySearch(key, suf_arr, ref)  # first position in the array that is
   greater than the key
block_aa ← the largest string among
         CommonPrefix(key, suf_arr[idx-1], ref)
         and CommonPrefix(key, suf_arr[idx], ref)
```

2.2.2 Position-Specific ChimeraMap

The PScMap algorithm (**Algorithm 2**, **Figure 1A-B**) operates similarly to the cMap algorithm described in section 2.1.2 and has been modified to select a search window as described for PScARS.

2.3 Implementation in ChimeraUGEM

The ChimeraUGEM software includes implementations of all aforementioned Chimera algorithms, and the widely-used CAI model. The software was written in MATLAB, with some parts implemented in C for efficiency. Binaries are provided for macOS, Linux, and Windows.

Algorithm 2: Position-Specific ChimeraMap

```
Procedure: PScMap
Input:   target, pos_arr_s, pos_arr_e, window
         ref_aa     # reference amino acid sequences
         ref_nt     # reference nucleotide sequences
         suf_arr    # suffix array for the amino acids reference
Output:  opt_seq
n ← size(target)
pos ← 1
opt_seq ← empty string
while pos <= n
   suf_arr_slice ← SelectWindow(suf_arr, pos_arr_s, pos_arr_e, window,
                                pos, pos-n-1)
   block_aa ← LongestPrefix(target[pos:n], suf_arr_slice, ref_aa)
   block_nt ← MostFreqPrefix(block_aa, suf_arr_slice, ref_aa, ref_nt)
   append block_nt to opt_seq
   pos ← pos + block_size
```

```
Procedure: MostFreqPrefix
Input:   suf_arr, ref_aa, ref_nt, block_aa
Output:  block_nt
L ← BinarySearch(block_aa, suf_arr, ref_aa)
key ← block_aa + '~'  # ordered last of strings with the prefix in block_aa
R ← BinarySearch(key, suf_arr, ref_aa)
n ← 3*size(key)-1
count ← empty hash table
for i from L to R-1
   pos ← 3*(suf_arr[i]-1) + 1  # position in nucleotide sequence
   key ← ref_nt[pos:pos+n]
   increment count[key]

block_nt ← argmax(count)  # tie-break: first in lexicographic order
```

2.4 Reference filtering based on sequence similarity

When the reference set contains sequences that are very similar to the target sequence, this may result in biased Chimera output. For example, if the most similar reference and the target are identical, cARS will be $(|T|+1)/2$ and thus provide little information on the underlying regulatory signals. If the reference set contains close homologs the cARS score will be biased, however to a lesser extent. To alleviate this, we employed 3 heuristics in our analyses and in ChimeraUGEM. In the first step, identical sequences may be discarded either from the reference set (for a particular target while it is analyzed) or

from the list of targets. Next, the maximal size for a shared sequence block between the target and some reference can be used to filter out the reference (max_len). When the program encounters a block that is larger than max_len, the block is discarded, and the reference will be flagged as non-relevant to that particular target for all future searches. Third, the fraction of positions in the target protein that were associated with the same reference sequence can be limited (max_pos). When the program detects that some reference exceeded the allowed fraction, the reference will be flagged as non-relevant to that particular target and these positions will be re-calculated (*i.e.*, new sequence blocks will be assigned to them).

The parameters utilized in all analyses here were 40 codons for max_len and 50% for max_pos. However, the sensitivity of these filters should be context-dependent and can be user-defined in ChimeraUGEM. For example, including orthologous sequences of the target protein, that may exist in the host organism, as reference for designing a plasmid could have a desirable effect for some purposes.

2.5 Codon Adaptation Index

The Codon Adaptation Index (CAI) (Sharp and Li, 1987) is one of the most widely used models for measuring codon usage bias and optimizing the coding sequence. First, given a reference set of coding sequences, the frequencies of all codons are computed $\{x_i\}_{i=1}^{64}$ (we included in our analyses stop codons and amino acids with no degeneracy, that were originally excluded in Sharp and Li). Next, for each amino acid, the most frequent synonymous codon is regarded as the most optimal and its weight is set to 1, while the rest of the synonymous codons are scaled with respect to the optimal codon. Finally, given a target sequence, its CAI score is the geometric mean of codon weights across all its positions. This may be formulated as $CAI = \left(\prod_{l=1}^{|T|} w_l\right)^{1/|T|}$, where $|T|$ is the length of the target sequence, and $w_i = x_i/\max(x_j)$ so that $j \in AA(i)$, *i.e.* the set of synonymous codons that encode the same amino acid as the i-th codon. In ChimeraUGEM, CAI is calculated by default when an analysis of a sequence (in NT alphabet) is performed. The user may also provide alternative codon weights (*e.g.*, weights derived from a tRNA Adaptation Index model (Reis et al., 2004; Sabi et al., 2017), or Typical Decoding Rates (Dana and Tuller, 2014a)) that will be used to compute the geometric mean of the target sequence. Codon optimization, according to this model, consists of selecting the synonymous codon with the highest weight (*i.e.*, highest frequency, in the case of CAI) at every position within user-specified regions.

2.6 Sequences and gene expression data

We analyzed genome-wide expression levels in 7 organisms (*A. thaliana, C. elegans, C. reinhardtii, D. melanogaster, E. coli, H. sapiens, S. cerevisiae*) using the models described in the previous sections. Gene sequences were obtained from Ensembl (Zerbino *et al.*, 2018) (CDS fasta files from Ensembl release 93 and Ensembl Genomes 40). Protein abundance datasets were downloaded from PaxDB (Wang *et al.*, 2015) (GPM datasets, accessed in August 2018) and (Leufken *et al.*, 2017) (for *C. reinhardtii*, see also **Table S1**). In addition, integrated tissue-specific human datasets were downloaded from PaxDB. In order to reduce noise that might originate from similarity between sequences in the reference set when calculating cARS scores, we kept the sequence of the longest isoform from each set of alternative splice variants.

2.7 Heterologous gene expression

We designed a synthetic gene using PScMap and expressed it in *Chlamydomonas reinhardtii* while comparing the results to cMap- and CAI-optimized variants of the same gene (Supplementary Methods). DNA sequences of engineered variants of a Ferredoxin-Hydrogenase (fd-hyd) fusion protein (Yacoby *et al.*, 2011) were synthesized and cloned into the pSL18 vector, under the control of the endogenous *psaD* promoter (Fischer and Rochaix, 2001), followed by recombination-based cloning. Algal strains were initially screened twice: first, with hygromycin; then, strains were overlaid with engineered H_2-sensing *R. capsulatus*, and scanned for hydrogen production which is catalyzed by fd-hyd (Eilenberg *et al.*, 2016). Following anaerobic induction of the algal cells (Meuser *et al.*, 2012), the amount of fd-hyd was quantified by measuring the concentration of produced H_2 using Methyl-Viologen (Eilenberg *et al.*, 2016). A representative subgroup of clones was selected from the low, medium, and top quantiles of expression. RNA levels were measured using real-time quantitative PCR (Supplementary Methods).

3 Results

3.1 PScARS improves expression prediction upon CAI and cARS

We first tested the ability of our proposed extension to cARS, the position-specific cARS (PScARS), to predict protein levels in an unsupervised setting, where expression levels are unknown prior to prediction. In this scenario, the input to PScARS is only the complete set of coding sequences of the organism. We considered 7 organisms where protein abundance data was available (**Table S1**).

PScARS predictions explained a higher percentage of the variance in expression as measured by Spearman correlation than cARS in all of the studied species and improved upon the traditional CAI in all species but *E. coli* (**Figure 2A**). Particularly, considerable improvement was observed in species where CAI performs poorly (human, *A. thaliana*, *C. elegans*, in agreement with previous studies that analyzed CAI in these organisms (Vogel *et al.*, 2010; Zhang *et al.*, 2012)). This may imply that signals that are longer than single codons have a significant effect on gene regulation in these organisms.

It is possible to represent coding sequences by their amino acid, codon (NT triplets), or nucleotide sequences; the latter alphabet may also apply to analysis of non-coding sequences. **Figure 2** reports the results for the codon alphabet, which showed the best prediction performance in our tests (**Figure S1** reports the results for the other alphabets).

We calibrated the window size parameter of PScARS based on gene expression prediction in *E. coli* and *S. cerevisiae* and observed a peak at 40 codons in both organisms when the input sequences were represented in the codon alphabet (**Figure 2B**). A similar peak was observed at 120nt in both species when the input sequences were represented in nucleotides (**Figure S1**). It is worth noting that the results were not highly sensitive to the selection of window size.

3.2 PScARS detects information-rich regions in the 5'- and 3'-ends of the coding region

Next, we computed meta-gene profiles of the average detected substring length by PScARS at each position in the transcript (**Figure 2C**). We found that all organisms tend to contain longer Chimera blocks at the 5'-end of the gene. These results are compatible with the large number of regulatory signals that are interleaved in the 5'-end of genes, and have been discussed in the literature (Tuller and Zur, 2015). We also found that human and *C. reinhardtii* tend to contain longer Chimera blocks towards the 3'-end of the gene, peaking around the position -30 codons. Furthermore, the correlation between PA and substring length at each position tends to modestly increase towards the 3'-end of the gene in most of the studied organisms. These results are in agreement with previous studies on termination signaling (Bertram *et al.*, 2001; Beznosková *et al.*, 2015), and may imply that this region is more signal-rich than previously thought.

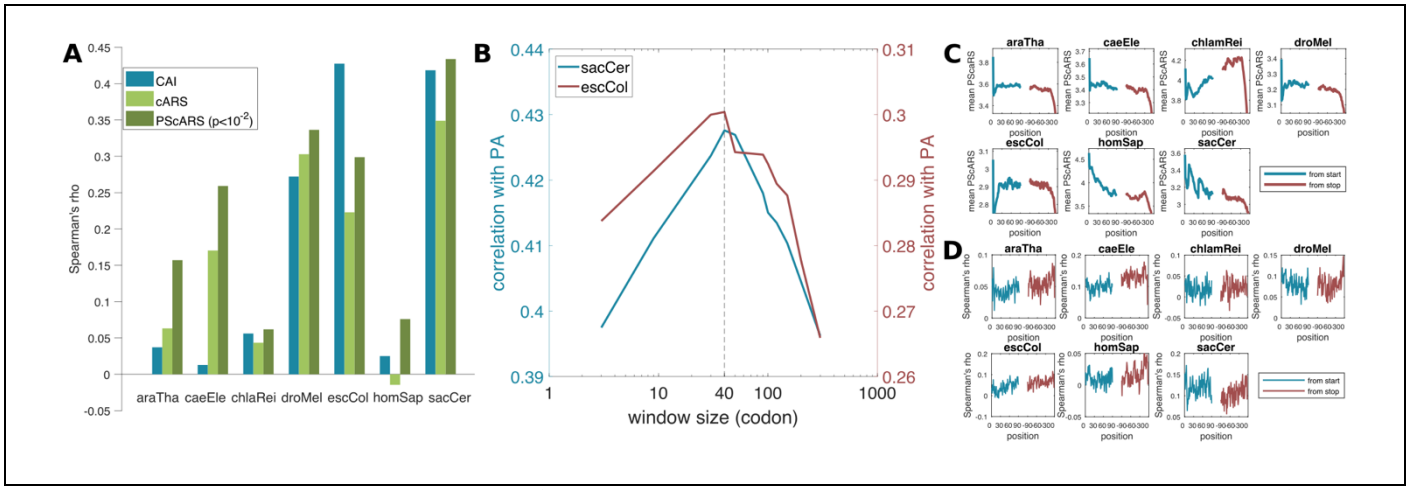

Fig. 2: Unsupervised prediction. (A) Unsupervised prediction of protein abundance (PA) based on CAI, cARS, and PScARS given an all-organism reference set for 7 species (*A. thaliana, C. elegans, C. reinhardtii, D. melanogaster, E. coli, H. sapiens, and S. cerevisiae*) (see also **Figure S1**). **(B)** The algorithm is not highly sensitive to the window size parameter, but shows modest improvement in predictions, and a clear maximum, for a size of 40 codons (see also **Figure S1**). **(C-D)** Meta-gene profiles of PScARS (based on the codon alphabet) in 7 species, positioned relatively to the START / STOP codons. Average PScARS shown in **(C)**. Correlation between PScARS and PA for each position shown in **(D)**.

3.3 PScARS predicts tissue-specific expression

In addition, Chimera algorithms can be used in a supervised manner by selecting a reference set of highly expressed genes. This additional information – 1,000 highly-expressed genes – led to a considerable increase in prediction performance (**Figure S2**), and demonstrates that PScARS can capture relevant regulatory signals that are present in highly expressed genes. The largest increase in correlation for PScARS was observed in human (from 0.08 to 0.25) and in the Arabidopsis plant (from 0.16 to 0.47).

We further tested the algorithm's ability to predict tissue-specific protein abundance (PA) in 7 human tissues. To this end, we generated PScARS and CAI predictions based on reference sets of 1,000 highly expressed genes in each of 8 tissues (inter-tissue correlations between PA datasets appear in **Figure S2**). We then selected one tissue (colon) as a reference tissue, and tested how well differential expression is captured by the fold-change in prediction scores. Strikingly, in all cases differential expression in tissue X are most correlated with differential predictions in the same tissue (the diagonal in **Figure 3A**, where $P<10^{-8}$), while the rest of the tissues are considerably less correlated. Thus, PScARS can capture tissue-specific signals well. In contrast, supervised CAI predictions were less specific to the correct tissue (**Figure S2**).

3.4 PScMap-optimized genes show significantly increased expression

PScMap was tested experimentally by designing a synthetic gene and expressing it in the single-cell green alga *C. reinhardtii* (Supplementary Methods). The gene that was designed is a fusion of Ferredoxin and Hydrogenase (fd-hyd), which has been suggested to have potential biotechnological applications (Yacoby et al., 2011). The expression of the PScMap-optimized gene was compared to that of cMap- and CAI-optimized genes (Weiner *et al.*, 2018). We observed an increase of up to 26-fold in protein abundance in the PScMap-optimized gene compared to cMap, and 15-fold compared to CAI (**Figure 3B**, **Figure S3**, **Table S2**). Interestingly, the increase in gene expression was also related to an increase in transcription levels, demonstrating the model's ability to capture diverse regulatory signals.

3.5 ChimeraUGEM is an easy-to-use tool for modeling and engineering gene expression

We developed the ChimeraUGEM (unsupervised gene expression modeling) software (**Figure 3C**), which provides tools for the analysis of gene sequences (coding and non-coding), as well as the design of protein coding sequences for optimized expression based on the Chimera algorithms and codon usage optimization. It accepts standard fasta files as input and output, generates summary tables in CSV format, and enables batch-processing of genes for analysis and design. All algorithm parameters can be user-defined via a graphical interface.

The program enables the user to analyze and engineer selected regions within genes (e.g., the first 50 codons), or to perform a complete analysis of the sequence. The implementation of cARS and PScARS supports the AA, codon, and NT alphabets. Furthermore, the program generates positional substring length profiles for each gene that can be used, *e.g.*, for meta-gene analysis.

We tested the run-time of ChimeraUGEM by calculating PScARS for one half of *E. coli*'s proteins (2,154 genes) using the other half as reference sequences, which required 36min (1 gene/sec) on a modern laptop, including the pre-processing of the reference set. PScMap on the same gene set required 19min (1.9 gene/sec).

4 Conclusions

We presented a methodology and a software tool for model-free analysis and engineering of gene expression. The reported results may suggest that signals of higher dimensions than single codons – which are captured by the position-specific Chimera algorithms – play a central role in the regulation of expression in the studied organisms. The unsupervised prediction results in section 3.1 demonstrate that the method may be applied to a multitude of organisms where gene expression measurements and models are missing, given only the set of coding sequences.

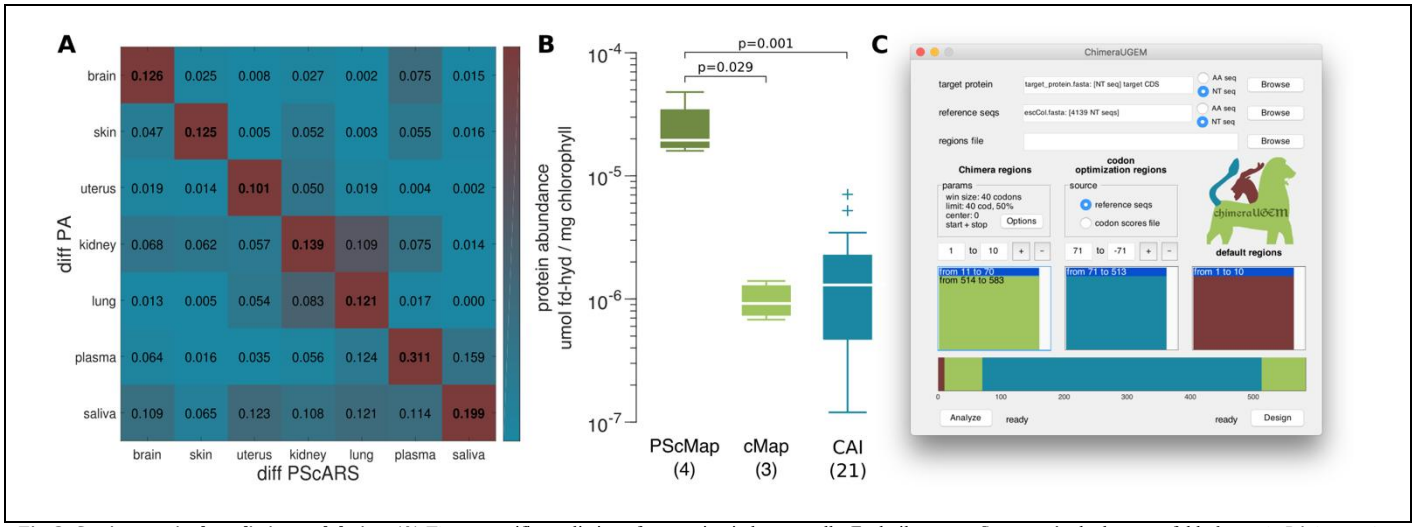

Fig. 3: Semi-supervised prediction and design. (A) Tissue-specific prediction of expression in human cells. Each tile reports Spearman's rho between fold-change in PA in one tissue and fold-change in PScARS in a second tissue. Colors are proportional to the explained variance and scaled by the maximal value in each row. $P<10^{-8}$ for all values on the diagonal (results for CAI appear in **Figure S2**). **(B)** Experimental validation of PScMap in *C. reinhardtii*. One-tailed rank-sum p-values reported above, and the number of clones below (see also **Figure S3**). **(C)** ChimeraUGEM interface. In this example, the first 10 codons are provided by the user; codons 11-70 and 514-583 are optimized using PScMap, and codons 71-513 are selected based on their frequency.

The analysis reported here demonstrate that the Chimera approach can be used as an exploration tool: for example, our results suggest that our approach is important in organisms such as *A. thaliana*, *C. elegans*, and *D. melanogaster* where it is able to significantly predict expression levels much better than codon based measured such as CAI. It will be interesting to perform experiments and modeling in the future to understand the nature of these longer codes in these organisms. Similarly, some organisms (*e.g.*, human and *C. reinhardtii*) tend to contain longer Chimera blocks towards the 3'-end of the gene, suggesting that these organisms include specific codes in these regions that should be further explored in the future. A study of such codes may, for example, enable a better understanding of mechanisms that regulate translation termination and post-termination processes (stop codon readthrough, reinitiation, and others).

The tissue-specific prediction of expression in section 3.3 suggests that PScARS has high sensitivity to regulatory signals. This result also suggests that tissue specific gene expression regulation is partially encoded in the coding region and not exclusively in promoters and enhancers.

Our experimental results in section 3.4 may imply that PScMap can be used to optimize gene expression very efficiently. Specifically, it can be used for non-model organisms when gene expression models are currently partial.

Furthermore, this approach can naturally be integrated with other approaches and models for gene expression analysis and optimization, including model-based optimization (Weiner *et al.*, 2018). The above results, and particularly the effect our coding sequence optimization on RNA levels, also emphasize the complexity of the coding region in terms of overlapping regulatory codes.

Acknowledgements

We would like to thank Prof. Matthew Posewitz for the *hydA*$_{1,2}$ double hydrogenase knockout mutant, and Prof. Matt Wecker and Prof. Maria Ghirardi for the H_2 sensing *R. capsulatus*.

Funding

A.D. is grateful to the Azrieli Foundation for the award of an Azrieli Fellowship. This study was supported in part by a fellowship from the Edmond J. Safra Center for Bioinformatics at Tel-Aviv University. This study was supported by the Israeli Ministry of Science, Technology and Space. This study was supported by a fellowship from the Manna Center for Plant Biosciences.

Conflict of Interest: none declared.

References

Alberts,B. *et al.* (2005) Molecular Biology of the Cell 4th ed. Garland Science.

Barash,Y. *et al.* (2010) Deciphering the splicing code. *Nature*, **465**, 53–59.

Ben-Yehezkel,T. *et al.* (2015) Rationally designed, heterologous S. cerevisiae transcripts expose novel expression determinants. *RNA Biology*, **12**, 972–984.

Bertram,G. *et al.* (2001) Endless possibilities: translation termination and stop codon recognition. *Microbiology*, **147**, 255–269.

Beznosková,P. *et al.* (2015) Translation initiation factor eIF3 promotes programmed stop codon readthrough. *Nucleic Acids Res*, **43**, 5099–5111.

Chu,D. *et al.* (2014) Translation elongation can control translation initiation on eukaryotic mRNAs. *EMBO J.*, **33**, 21–34.

Churchman,L.S. and Weissman,J.S. (2011) Nascent transcript sequencing visualizes transcription at nucleotide resolution. *Nature*, **469**, 368–373.

Cohen,E. *et al.* (2018) A code for transcription elongation speed. *RNA Biology*, **15**, 81–94.

Dana,A. and Tuller,T. (2014a) Mean of the typical decoding rates: a new translation efficiency index based on the analysis of ribosome profiling data. *G3 (Bethesda)*, **5**, 73–80.

Dana,A. and Tuller,T. (2014b) The effect of tRNA levels on decoding times of mRNA codons. *Nucl. Acids Res.*, **42**, 9171–9181.

Demain,A.L. and Vaishnav,P. (2009) Production of recombinant proteins by microbes and higher organisms. *Biotechnology Advances*, **27**, 297–306.

Eilenberg,H. *et al.* (2016) The dual effect of a ferredoxin-hydrogenase fusion protein in vivo: successful divergence of the photosynthetic electron flux towards hydrogen production and elevated oxygen tolerance. *Biotechnology for Biofuels*, **9**, 182.

Ferrer-Miralles,N. *et al.* (2009) Microbial factories for recombinant pharmaceuticals. *Microbial Cell Factories*, **8**, 17.

Fischer,N. and Rochaix,J.D. (2001) The flanking regions of PsaD drive efficient gene expression in the nucleus of the green alga Chlamydomonas reinhardtii. *Molecular Genetics and Genomics*, **265**, 888–894.

Frenzel,A. *et al.* (2013) Expression of Recombinant Antibodies. *Front. Immunol.*, **4**.

Gaspar,P. *et al.* (2012) EuGene: maximizing synthetic gene design for heterologous expression. *Bioinformatics*, **28**, 2683–2684.

Goodman,D.B. *et al.* (2013) Causes and Effects of N-Terminal Codon Bias in Bacterial Genes. *Science*, **342**, 475–479.

Kimchi-Sarfaty,C. *et al.* (2007) A 'silent' polymorphism in the MDR1 gene changes substrate specificity. *Science*, **315**, 525–528.

Kozak,M. (1999) Initiation of translation in prokaryotes and eukaryotes. *Gene*, **234**, 187–208.

Kramer,G. *et al.* (2009) The ribosome as a platform for co-translational processing, folding and targeting of newly synthesized proteins. *Nat Struct Mol Biol*, **16**, 589–597.

Kudla,G. *et al.* (2009) Coding-Sequence Determinants of Gene Expression in Escherichia coli. *Science*, **324**, 255–258.

Leufken,J. *et al.* (2017) pyQms enables universal and accurate quantification of mass spectrometry data. *Molecular & Cellular Proteomics*, **16**, 1736–1745.

Li,G.-W. *et al.* (2012) The anti-Shine-Dalgarno sequence drives translational pausing and codon choice in bacteria. *Nature*, **484**, 538–541.

Meuser,J.E. *et al.* (2012) Genetic disruption of both Chlamydomonas reinhardtii [FeFe]-hydrogenases: Insight into the role of HYDA2 in H_2 production. *Biochemical and biophysical research communications*, **417**, 704–9.

Peden,J.F. (2000) Analysis of codon usage.

Puigbò,P. *et al.* (2008) E-CAI: a novel server to estimate an expected value of Codon Adaptation Index (eCAI). *BMC Bioinformatics*, **9**, 65.

Reis,M. dos *et al.* (2004) Solving the riddle of codon usage preferences: a test for translational selection. *Nucleic Acids Res*, **32**, 5036–5044.

Sabi,R. *et al.* (2017) stAIcalc: tRNA adaptation index calculator based on species-specific weights. *Bioinformatics*, **33**, 589–591.

Sabi,R. and Tuller,T. (2015) A comparative genomics study on the effect of individual amino acids on ribosome stalling. *BMC Genomics*, **16**, S5.

Sabi,R. and Tuller,T. (2017) Computational analysis of nascent peptides that induce ribosome stalling and their proteomic distribution in Saccharomyces cerevisiae. *RNA*, **23**, 983–994.

Sharp,P.M. and Li,W.-H. (1987) The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. *Nucl. Acids Res.*, **15**, 1281–1295.

Stadler,M. and Fire,A. (2011) Wobble base-pairing slows in vivo translation elongation in metazoans. *RNA*, **17**, 2063–2073.

Stergachis,A.B. *et al.* (2013) Exonic Transcription Factor Binding Directs Codon Choice and Affects Protein Evolution. *Science*, **342**, 1367–1372.

Terpe,K. (2006) Overview of bacterial expression systems for heterologous protein production: from molecular and biochemical fundamentals to commercial systems. *Appl Microbiol Biotechnol*, **72**, 211.

Tuller,T., Carmi,A., *et al.* (2010) An Evolutionarily Conserved Mechanism for Controlling the Efficiency of Protein Translation. *Cell*, **141**, 344–354.

Tuller,T. *et al.* (2011) Composite effects of gene determinants on the translation speed and density of ribosomes. *Genome Biology*, **12**, R110.

Tuller,T., Waldman,Y.Y., *et al.* (2010) Translation efficiency is determined by both codon bias and folding energy. *PNAS*, 200909910.

Tuller,T. and Zur,H. (2015) Multiple roles of the coding sequence 5′ end in gene expression regulation. *Nucl. Acids Res.*, **43**, 13–28.

Vogel,C. *et al.* (2010) Sequence signatures and mRNA concentration can explain two-thirds of protein abundance variation in a human cell line. *Mol. Syst. Biol.*, **6**, 400.

Wang,M. *et al.* (2015) Version 4.0 of PaxDb: Protein abundance data, integrated across model organisms, tissues, and cell-lines. *Proteomics*, **15**, 3163–3168.

Weiner,I. *et al.* (2018) Enhancing heterologous expression in Chlamydomonas reinhardtii by transcript sequence optimization. *The Plant Journal*, **94**, 22–31.

Welch,M. *et al.* (2009) Design Parameters to Control Synthetic Gene Expression in Escherichia coli. *PLOS ONE*, **4**, e7002.

Wu,G. *et al.* (2005) Predicted highly expressed genes in the genomes of Streptomyces coelicolor and Streptomyces avermitilis and the implications for their metabolism. *Microbiology*, **151**, 2175–2187.

Wurm,F.M. (2004) Production of recombinant protein therapeutics in cultivated mammalian cells. *Nature Biotechnology*, **22**, 1393–1398.

Xia,X. (1996) Maximizing Transcription Efficiency Causes Codon Usage Bias. *Genetics*, **144**, 1309–1320.

Yacoby,I. *et al.* (2011) Photosynthetic electron partitioning between [FeFe]-hydrogenase and ferredoxin:NADP+-oxidoreductase (FNR) enzymes in vitro. *PNAS*, **108**, 9396–9401.

Yordanova,M.M. *et al.* (2018) *AMD1* mRNA employs ribosome stalling as a mechanism for molecular memory formation. *Nature*, **553**, 356–360.

Zafrir,Z. and Tuller,T. (2015) Nucleotide sequence composition adjacent to intronic splice sites improves splicing efficiency via its effect on pre-mRNA local folding in fungi. *RNA*, **21**, 1704–1718.

Zafrir,Z. and Tuller,T. (2017) Unsupervised detection of regulatory gene expression information in different genomic regions enables gene expression ranking. *BMC Bioinformatics*, **18**, 77.

Zerbino,D.R. *et al.* (2018) Ensembl 2018. *Nucleic Acids Res*, **46**, D754–D761.

Zhang,G. *et al.* (2009) Transient ribosomal attenuation coordinates protein synthesis and co-translational folding. *Nat. Struct. Mol. Biol.*, **16**, 274–280.

Zhang,Z. *et al.* (2012) Codon Deviation Coefficient: a novel measure for estimating codon usage bias and its statistical significance. *BMC Bioinformatics*, **13**, 43.

Zur,H. and Tuller,T. (2015) Exploiting hidden information interleaved in the redundancy of the genetic code without prior knowledge. *Bioinformatics*, **31**, 1161–1168.

Zur,H. and Tuller,T. (2013) New Universal Rules of Eukaryotic Translation Initiation Fidelity. *PLOS Computational Biology*, **9**, e1003136.