THE PENNSYLVANIA STATE UNIVERSTIY

SCHREYER HONORS COLLEGE

DEPARTMENRT OF AGRICULTURAL AND BIOLOGICAL ENGINEERING

EFFECT OF CODON OPTIMIZATION ON BACTERIAL TRANSLATION ELONGATION RATES

CLAY SWACKHAMER
FALL 2015

A thesis
submitted in partial fulfillment
of the requirements
for a baccalaureate degree in Biological Engineering
with honors in Biological Engineering

Reviewed and approved* by the following:

Howard Salis
Assistant Professor of Chemical Engineering
&Assistant Professor of Agricultural and Biological Engineering
Thesis Supervisor

Ali Demirci
Professor of Agricultural and Biological Engineering
Honors Adviser

*Signatures are on file in the Schreyer Honors College.

ABSTRACT

Microorganisms are used in a variety of industries to produce bioproducts, including fuels, food products, specialty chemicals, and even lucrative biologic therapeutics. These products are created through manipulation of the natural ability of microorganisms to use genetic information to produce proteins. This is known as the central dogma of biology, and is mediated by the steps of DNA being transcribed to mRNA and then translated into protein. It is the task of engineers to use an understanding of these component biophysical processes to control protein synthesis through genetic level controls, and to create new systems for protein production using the principles of rational design. Using engineering knowledge to raise protein expression is a goal in numerous industries, as higher protein expression is a driver of overall profit.

There are several genetic level points of control currently used by engineers to raise protein expression, one of which is codon optimization of the coding sequences used by microorganisms to produce proteins. Since there are 20 amino acids but 64 codons, there are instances were multiple codons are translated into the same amino acid. These are called degenerate codons. However, degenerate codons may not have the same translational efficiency. Traditional codon optimization methods in *E. coli* rely on the preference for certain codons across the entire genome, yet this is not the only possible approach. The principal challenge is that at high translation initiation rates, protein expression may plateau, and it is hypothesized that novel criteria for codon optimization of genes can be used to raise expression plateaus that occur when translation elongation becomes the rate limiting step in protein synthesis, and also allow for fine tuning of expression due to predicted differences in translational efficiency between degenerate codons.

In this research, novel criteria for codon optimization were employed to design and create synthetic variants of a reporter gene that was then characterized in vivo using an expression construct. Fluorescence levels of cells expressing these constructs were measured and results

suggest that protein expression plateaus may still be experienced, even by the sequences optimized for high efficiency. However, the new criteria for codon optimization, for example the statistical correlation between a degenerate codon and its presence in highly translated parts of the genome, are feasible for use in future projects. This may enable future researchers to optimize genes at the codon level with greater fidelity.

TABLE OF CONTENTS

List of Tables

List of Figures

List of Equations

ACKNOWLEDGEMENTS

I would like to recognize the individuals that have assisted me with this project as well as the organizations that provided the resources for me to conduct undergraduate research. First, I would like to thank the International Genetically Engineered Machine Foundation (iGEM), and the undergraduates who were part of the 2014 Penn State team: Samuel Krug, Ashlee Smith, and Emily Sileo.

I would like to extend a sincere thank you to Dr. Howard Salis, my thesis supervisor, who introduced me to the world of synthetic biology and was instrumental in every aspect of the project.

I would like to thank Dr. Ali Demirci, my thesis advisor, as well as Dr. Tom Richard, who co-advised the iGEM 2014 team.

I would like to thank the members of the Salis Laboratory for Synthetic Biology, in particular Chiam Yu Ng, Iman Farasat, Amin Espah Borujeni, Tian Tian, Alex Reis, Grace Vezeau, Sean Halper, and Manish Kushwaha.

I would like to extend a special thank you to Dr. Megan Marshall, Dr. Paul Heinemann, the judges and sponsors of iGEM, and the professors at the University of Alicante, who were very understanding of my travel to the iGEM conference in the middle of an academic semester.

I would like to thank the Agricultural and Biological Engineering and Chemical Engineering departments at Penn State, as well as the Schreyer Honors College. Through this research I have extended classroom knowledge to a real world project, collaborated with incredible individuals, and by doing so gained both knowledge and skills that are the foundation of my professional future.

INTRODUCTION

Numerous bioproducts are important to our daily lives. Examples include medicines, fuels, industrial chemicals, and even components of laundry detergents. Biological engineering has in particular proven to be a driver of growth in the pharmaceutical industry, where the contribution of the biological therapeutic medicine (biologics) industry is estimated at $789 billion [1]. Even though the present economic footprint of this industry is already very large it is recognized that the opportunities continue to grow, as the 8% annual growth rate of biopharma is roughly double that of the conventional pharmaceutical industry, and growth is expected to continue for the predicted future [2]. Due to the size of this industry and the complexity of the production processes that sustain it, there are significant opportunities for fundamental advances in the engineering of these processes to have a large impact. The products that form the profitable base of the biopharma industry include proteins, which are often produced by recombinant DNA technology.

The foundational principle of this technology is that organisms can be programmed to create desired protein products by the introduction of new DNA, and this is accomplished through the principles of rational design [3]. The use of DNA to cause change in an organism's molecular function is based on the underlying principle of molecular biology, the central dogma. The central dogma is a statement of the flow of information in organisms, which is from DNA to RNA to protein. The information which contains all of the instructions for life is found in the sequence of the DNA, but before an organism acts on these instructions, the DNA is transcribed to an intermediate molecule called messenger ribonucleic acid (mRNA), and then the mRNA is translated into proteins, [4].

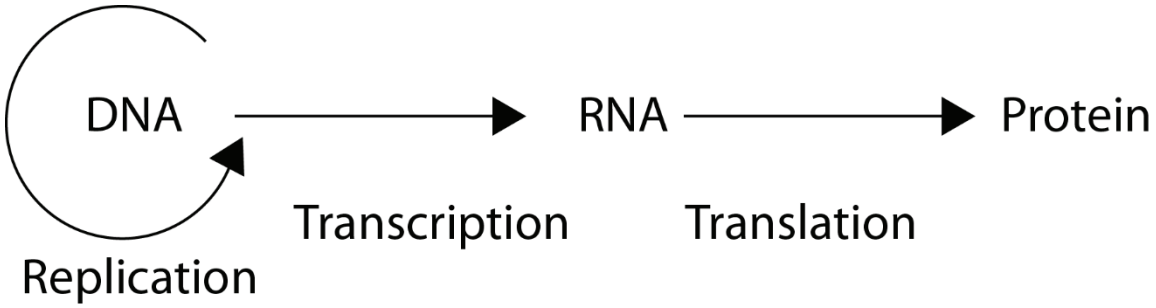


Figure 1: Central Dogma of Biology

It is important to understand some basic concepts about the components that play a role in the central dogma, such as DNA, mRNA, the ribosome, and protein. DNA is a polymer of five-carbon (deoxyribose) sugar molecules, which are covalently bonded to phosphate groups, forming a polymer, or chain, with nitrogenous bases in between. There are four possible bases, Adenine (A), Thymine (T), Cytosine (C), and Guanine (G), which pair in a predictable manner, known as Watson—Crick Base pairing: A – T and G – C [5].

In DNA, the 5-carbon sugar that forms the sugar-phosphate backbone of the molecule is deoxyribose, and in RNA, the sugar is ribose, which can be distinguished from DNA by the presence of the hydroxyl group at the 2' position of the ribose sugar. In addition, the base uracil (U) is found in RNA instead of thymine in DNA. According to the central dogma, mRNA is responsible for acting as an information carrying intermediate between DNA and protein. Messenger RNA is single stranded, but can form elaborate and functional structures through Watson-Crick base pairing of complimentary nucleotides on the same strand [6]. These are referred to as secondary structures.

Production of mRNA occurs during transcription, a process during which an enzyme called RNA polymerase latches onto the DNA and then catalyzes the formation of phosphodiester bonds between nucleoside triphosphate residues which are found in the cellular cytoplasm [7]. The resulting mRNA molecule is called a "transcript" and is complementary to the template DNA.

Transcription begins at a site called the promoter, which is a sequence of nucleotides where RNA polymerase can attach to the DNA and begin translation, and stops at a feature called the terminator, where the enzyme disassociates with the DNA and the finished mRNA strand is released [3].

Translation is the process by which mRNA is read by ribosomes, which are complex catalytic molecules comprised of protein and an rRNA scaffold. This process results in the addition of specific amino acids to a polypeptide chain, which is a polymer of the individual amino acids that are coded for by the mRNA and then joined with peptide bonds at the ribosome.

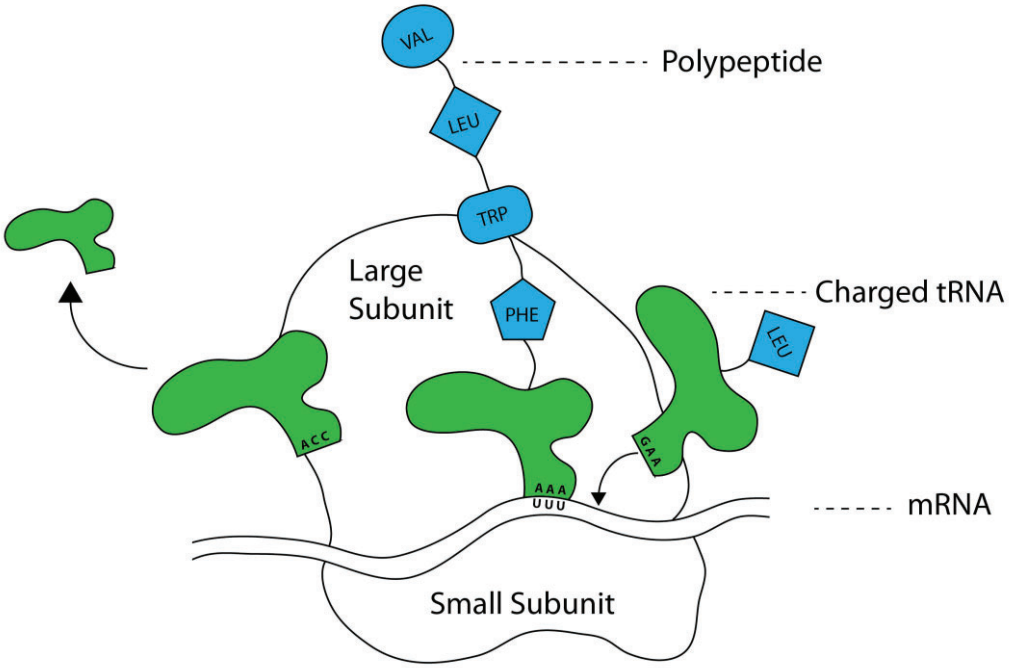


Figure 2: Illustration of translation. Charged tRNA brings amino acids to the translation complex, which is comprised of the ribosome small and large subunits

There are only 20 amino acids coded by DNA, but using these monomers, elaborate and highly functional molecules called proteins can be constructed by the cell. When a protein is created by a cell it is said to be “expressed.” Protein expression levels depend on multiple genetic elements involved in the transcription, translation and regulatory machinery. The molecular

processes that govern protein production in cells are widely conserved across the tree of life, and they can be understood in the context of foundational principles such as thermodynamics and material balances, which are used as guides for engineering DNA to accomplish a specific goal. However, engineers are often interested in producing specific proteins at a level not naturally found in cells. With recent advances in synthetic biology, it is now possible to tune protein expression to a desired level by engineering these genetic elements. Generally, one can modulate protein expression by tuning the gene transcription rate of that protein's coding sequence or by tuning its mRNA translation rate (i.e. initiation and elongation rate).

It is the objective of this project to first introduce the current methods for engineering expression and then investigate a new method based on next-generation criteria. An overview of current genetic control "knobs" is presented in Figure 3.

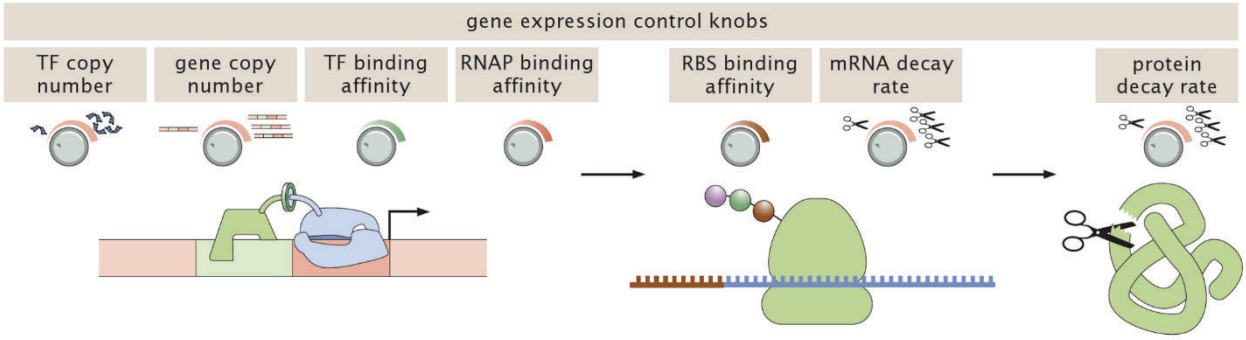


Figure 3: Control Points for Gene Expression (Brewster, Jones and Phillips, 2012)

These regulation points arise at the checkpoints of protein production: transcription and translation, as well as the post translation steps such as folding and decay. More copies of a DNA strand coding for a particular protein will increase expression, and this is shown in Figure 3 as "gene copy number." During transcription of the gene, certain accessory molecules called "transcription factors" are necessary, and thus increasing the number of copies of DNA coding for these transcription factors also will increase expression [8]. This is shown in Figure 3 as "TF copy

number." Increases in protein expression can also be accomplished by reduction of mRNA and protein decay rate [9].

Transcription begins at a sequence of DNA called the promoter, which is a site where RNA polymerase can bind to the DNA and being to create an mRNA molecule. Creating more copies of mRNA leads to higher expression, and thus expression can be raised by modifying the promoter sequence to enhance binding with transcription factors, shown in Figure 3 as "TF biding affinity," as well as binding with RNA polymerase, shown as "RNAP binding affinity" [10]. The strength of a promoter is rated by the number of initiation events per unit time [7]. For this project, the same J23100 promoter (moderate strength) was used in all constructs, the purpose of which was to standardize TF and RNA polymerase binding affinity in all of the constructs and enable translation elongation to be studied.

Increasing the amount of initiations of transcription per unit time results in an increase in the concentration of a particular mRNA. This in turn increases protein expression, which can be visualized using the simple model of protein expression in Equation 3 [9].

$$\frac{dP}{dt} = L \cdot r - U \cdot p$$

Equation 1: Model of Protein Expression

Where:

L	=	Translational constants
r	=	mRNA concentration
U	=	Degradation of protein
P	=	Protein concentration

However, transcription is only one genetic point of control. Another method for increasing protein production is shown in Figure 3 as "RBS binding affinity." This can be considered part of

the translational constants, L, which are shown to contribute to higher protein expression based on the model in Equation 3.

Translation can only occur when the ribosome successfully binds to the mRNA during the translation initiation event. In the intracellular environment, mRNAs and ribosomes are present in the cytoplasm, with the ribosome present in its disassociated form, a large and small subunit [11]. These components come together once the ribosome associates with the mRNA at the ribosome binding site (RBS). The RBS is a sequence of nucleotides near the 5' end of the mRNA transcript (roughly 35 base pairs upstream of the coding sequence and extending up to the start of the mRNA [12]. which can Watson-Crick base pair with a sequence near the 3' end of the rRNA on the small subunit of the ribosome [3]. The specific site of the hybridization between rRNA and mRNA is known as the Shine—Dalgarno sequence, and the complementary site on the rRNA is called the anti-Shine—Dalgarno sequence [13].

The probability of this association can be described using statistical thermodynamics. In this approach, two states are described: the first being the folded mRNA transcript and the free ribosomal small subunit separately, and the second the assembled 30S-initiation complex [14]. The probability of the complex assembling for a particular mRNA is proportional to the difference in Gibbs free energy between the two states, as described in Equation 2 [14].

$$r \; \alpha \; e^{(-\beta \cdot \Delta G_{tot})}$$ Equation 2: Translation Initiation Rate

Where:

r	=	Translation initiation rate (au)
β	=	Boltzman factor
ΔG_{tot}	=	Total Gibbs free energy change (kcal/mol)

An assumption of thermodynamic models is that the processes which result in translation initiation are in quasi-equilibrium, and this assumption allows the use of statistical methods to determine the probability of an event occurring based on the change in free energy necessary for that reaction to take place [15].

From the calculation in Equation 2, a more negative change in Gibbs free energy for a particular mRNA associating with the ribosome will lead to a higher proportional rate at which the translation initiation complex is formed for that particular mRNA. This is known as translation initiation rate (TIR). The calculation of the total change in Gibbs free energy is complex and must incorporate the work needed to unfold the mRNA secondary structure, penalty associate with non-optimal spacing between the RBS and the start codon, as well as other terms [14].

The result of the calculations is a tunable genetic knob that allows engineers to rationally design synthetic RBS sequences that will result in a desired TIR measured on a proportional scale. Translation initiation is the rate limiting step in almost all cases [14,16,17], and by increasing the RBS strength protein expression has been predictively increased by over 5 orders of magnitude [14,15]. These calculations can be conducted to both determine the predicted TIR of a known mRNA sequence, or to engineer new mRNAs for a desired TIR, and are available through the Ribosome Binding Site Calculator (https://salislab.net/software).

Once the translation complex has been assembled and the mRNA bound, translation elongation begins. In elongation, the mRNA transcript is read in three letter sequences, called codons. Thus, amino acid residues are added to the polypeptide chain at a rate of one for every three base pairs in the mRNA. Transfer RNA (tRNA) serves as the decoding molecule, as it pairs with each specific codon of mRNA and carries a corresponding amino acid residue. This process can occur at rates of roughly 22 amino acids per second in bacteria [18]. Amino acids are re-attached to tRNA through “charging” by an aminoacyl tRNA synthetase [11]. After deposition of the amino

acid and exit from the ribosome, the tRNA can be recharged and participate in additional reactions. Specificity for tRNA binding to the mRNA occurs through Watson-Crick base pairing between the tRNA anticodon and a complimentary section of the mRNA. This reaction occurs inside the ribosome, which serves as a rigid scaffold, optimally positioning each substrate [19].

After reaching a stop codon, the reaction is terminated and the translation complex disassociates, allowing the ribosome and mRNA to diffuse away from the site and participate in additional translation reactions. A simple Markov chain of the overall process of translation as mediated by the rates of the component steps is presented in Figure 4.

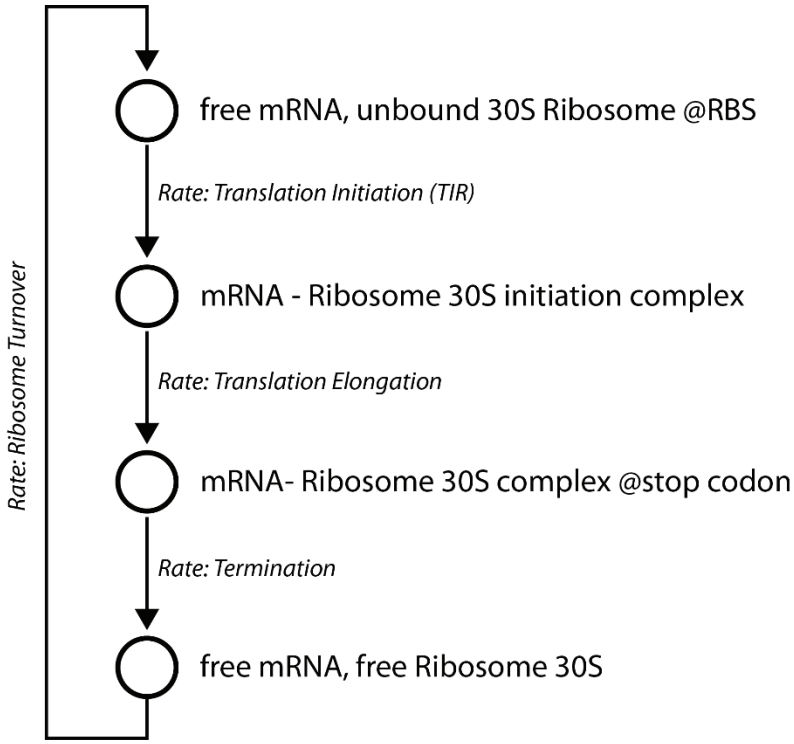


Figure 4: Markov Chain of Translation. Circles are states in which the participant molecules can exist, and arrows are transitions between the states. It is assumed that all major states are known, resulting in a discretized chain.

During translation, each amino acid is connected to the following amino acid by a peptide bond catalyzed by the ribosome. This Gibbs free energy change for this overall reaction is positive, which means that an input of energy is required. This is accomplished by the hydrolysis of guanosine triphosphate (GTP). As the reaction proceeds, the ribosome positions more tRNAs along the mRNA and allows more peptide bonds to be catalyzed between the amino acids that are carried by the tRNA.

Even though TIR is generally the rate limiting step in protein synthesis, there are instances when protein expression plateaus even as TIR is increased. In these cases, it is assumed that initiation is no longer the rate limiting factor, and other methods for raising production are needed. In a recent experiment, the RBS strength in a construct housing the reporter gene GFP mut3b was increased using the RBS Calculator. The expression was then characterized. Unexpectedly,

expression level of the protein plateaued even as the RBS strength (and thus TIR) was increased. It was then detected where the plateau occurred, which is called the "maximum translation rate capacity." Since the maximum translation rate capacity occurs independently of TIR, it is theorized that it is due solely to translation elongation becoming a rate limiting step. These results are shown in Figure 5.

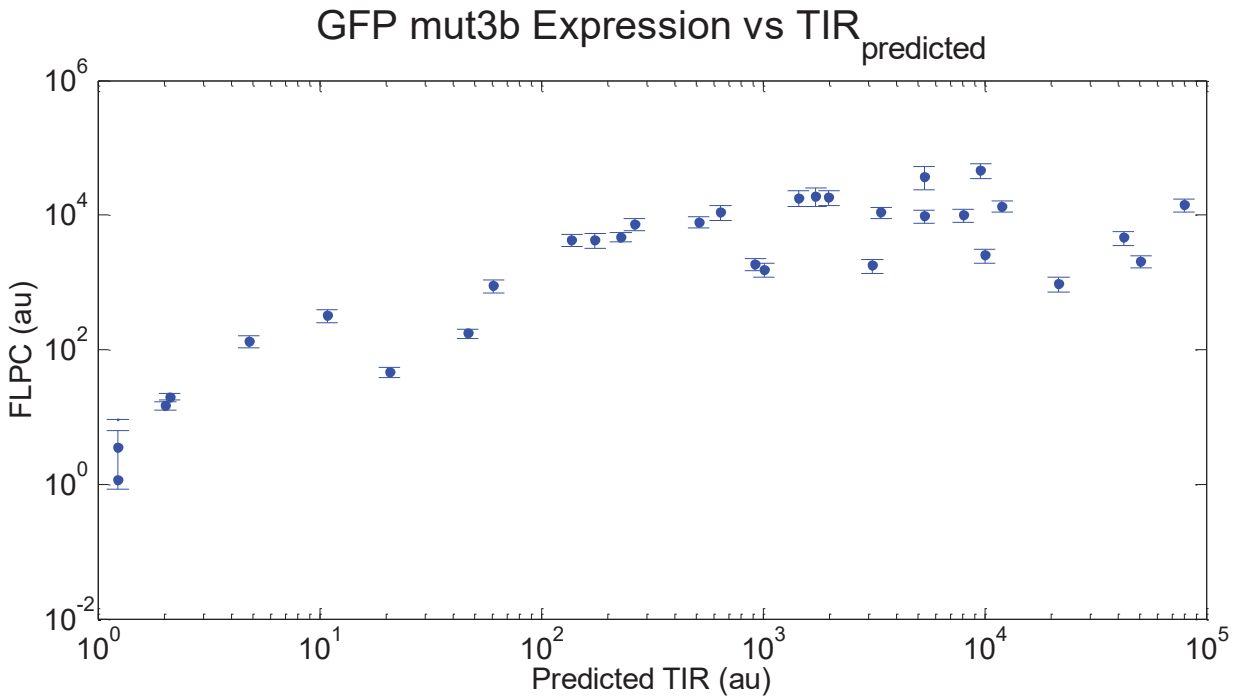


Figure 5: Expression can plateau at high TIR. [20]

The presence of the maximum translation rate capacity has also been predicted by computational models of translation [21]. From this approach it is predicted that at high TIR the rate limiting step becomes the “flow” from codon to codon, and thus the expression converges to a constant value (maximum translation rate capacity) that is set by the elongation rates [21]. This is demonstrated in Figure 6.

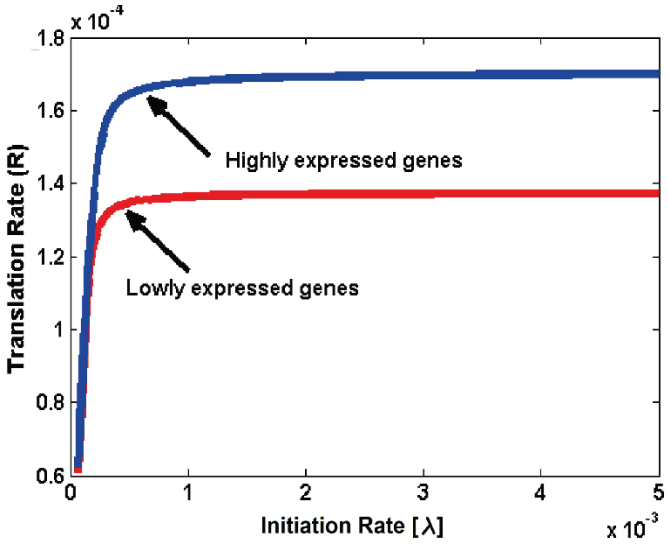


Figure 6: The Maximum Translation Rate Capacity is Reached at High TIR (Reuveni, et. al., 2011)

Once TIR has already been raised to high levels, expression must be raised through accelerating translation elongation rate [16], and control of this process can be accomplished through modification of the codon composition of the mRNA transcript [21].

A codon is formed by all possible permutations of the four nucleotides present in DNA (A,T.C,G) in a three letter series, thus 4^3, or 64 codons are possible. Since there are less amino acids than distinct codons, there is redundancy in the codons, that is, some amino acids are specified by multiple codons, called. Thus, there are numerous possible sequences of DNA that will lead to production of a protein with the same sequence of amino acids.

In the standard genetic code, there are 61 codons which code for 20 amino acids, and 3 which carry no amino acid and instead serve as a signal to stop translation, thus are known as "stop codons" [22]. Even though there is redundancy, there is no ambiguity, meaning that each codon specifies only one amino acid. Codons that code for the same amino acid are called degenerate codons. However, degenerate codons do not necessarily lead to the same expression levels of that amino acid [23].

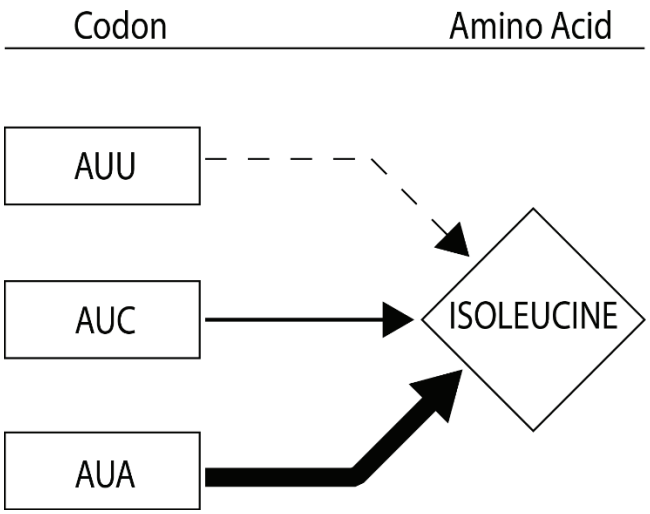


Figure 7: Degenerate codons may not be translated at the same rate

By choosing specific codons from the degenerate set for a particular amino acid, the translation elongation rate for a gene can be modified, and using this approach, improvements in expression of heterologous proteins have been measured. A phenomenological approach is to analyze the coding sequences of an organism, and then choose only the most commonly occurring codon of a degenerate set to be used whenever that amino acid is called by the coding sequence. In this project, this method was used to construct two variant coding sequence for a reporter protein. Although blind to the interaction mechanism of codon choice and translational efficiency, this "common codon" approach has nevertheless proven to be effective, and was used in a recent experiment to raise production of an industrially significant enzyme, α amylase, by up to 2.62 fold verses a wild type gene [24].

In another project that employed the same optimization method, a therapeutic protein for a proposed vaccine was expressed from a gene that was designed using only the most frequently used codons for E. coli, and it was found that expression of the recombinant protein was raised by up to four fold using the codon optimized coding sequence [25]. Using the same approach, a three-

fold increase in expression and significant increase in cellular growth rate were found when a recombinant vaccine protein was produced in E. coli using a codon optimized CDS [26].

These examples show that codon optimization has been proven to be an effective method of raising expression of heterologous proteins. It is theorized that expression was raised due to higher translation elongation rates in optimized genes, stemming for the fact that only rapidly translated, efficient codons were used. As a consequence of this, it is hypothesized that through codon optimization, plateaus in protein expression due to translation elongation becoming the rate limiting step could be lifted, and design of coding sequences could be conducted to ensure that protein production can always be maximized.

Coding Sequence Design

In order to test the hypothesis that codon optimization could be used to lift expression plateaus at high translation initiation rates, an expression construct was designed that employed a reporter protein codified by a codon optimized sequence. The protein that was chosen was Superfolder Green Fluorescent Protein (sGFP), which is a synthetic variant of the naturally occurring GFP (*Aequorea victoria*), that has been optimized for fast post translation folding and high fluorescence [27]. The advantage of using an extremely fast folding protein such as sGFP is that it is very likely that the protein will be able to correctly fold even with a higher translation elongation rate. Most proteins fold on the order of milliseconds [3]. and by using a known fast folder the possibility that high translation rates would negatively affect the activity of the reporter protein was greatly diminished. Additionally, sGFP is about four fold as bright as the commonly used reporter, GFP mut3b, and is more resistant to denaturing [27] which allows for easier fluorescence assays.

The next step in the design was to determine specific criteria for optimization of the sGFP coding sequences (720 base pairs). The first approach was the traditional method of optimization where whichever codon is most commonly used within a degenerate set is used whenever that amino acid is called by the coding sequence.

These “common” and “rare” codons were identified by taking the results of a statistical analysis of the entire *E. coli* K12 genome [28]. Some degenerate codons occur more often in protein coding sequences and some are more infrequent, and bias is strongest in highly expressed genes [23]. This indicates that the codon level composition of coding sequences impacts translational efficiency. Codons that are over or under preferred in the overall entire genome are referred to as

"common" and "rare" codons. Using this distinction, two variant sGFP coding sequences were constructed, one with only common codons, and one with only rare codons. Data for codon usage preference across the entire genome is shown in Table 1.

Table 1: Codon Usage Bias in E. coli

	CODON	AMINO ACID	USAGE RATIO	CODON	AMINO ACID	USAGE RATIO	CODON	AMINO ACID	USAGE RATIO	CODON	AMINO ACID	USAGE RATIO
U	UUU	Phe (F)	51%	UCU	Ser (S)	19%	UAU	Tyr (Y)	53%	UGU	Cys (C)	43%
	UUC	Phe (F)	49%	UCC	Ser (S)	17%	UAC	Tyr (Y)	47%	UGC	Cys (C)	57%
	UUA	Leu (L)	11%	UCA	Ser (S)	12%	UAA	Stop	62%	UGA	Stop	30%
	UUG	Leu (L)	11%	UCG	Ser (S)	13%	UAG	Stop	9%	UGG	Trp (W)	100%
C	CUU	Leu (L)	10%	CCU	Pro (P)	16%	CAU	His (H)	52%	CGU	Arg (R)	42%
	CUC	Leu (L)	10%	CCC	Pro (P)	10%	CAC	His (H)	48%	CGC	Arg (R)	37%
	CUA	Leu (L)	3%	CCA	Pro (P)	20%	CAA	Gln (Q)	31%	CGA	Arg (R)	5%
	CUG	Leu (L)	55%	CCG	Pro (P)	55%	CAG	Gln (Q)	69%	CGG	Arg (R)	8%
A	AUU	Ile (I)	47%	ACU	Thr (T)	21%	AAU	Asn (N)	39%	AGU	Ser (S)	13%
	AUC	Ile (I)	46%	ACC	Thr (T)	43%	AAC	Asn (N)	61%	AGC	Ser (S)	27%
	AUA	Ile (I)	7%	ACA	Thr (T)	30%	AAA	Lys (K)	76%	AGA	Arg(R)	4%
	AUG	Met (M)	100%	ACG	Thr (T)	23%	AAG	Lys (K)	24%	AGG	Arg (R)	3%
G	GUU	Val (V)	29%	GCU	Ala (A)	19%	GAU	Asp (D)	59%	GGU	Gly (G)	38%
	GUC	Val (V)	20%	GCC	Ala (A)	25%	GAC	Asp (D)	41%	GGC	Gly (G)	40%
	GUA	Val (V)	17%	GCA	Ala (A)	22%	GAA	Glu (E)	70%	GGA	Gly (G)	9%
	GUG	Val (V)	34%	GCG	Ala (A)	34%	GAG	Glu (E)	30%	GGG	Gly (G)	13%
	U			C			A			G		

The most common codon in each degenerate set is shown in green, and the least common in red [28].

For example, if the amino acid Phenylalanine was called, the codons UUU (usage ratio 0.51) and UUC were available (usage ratio 0.49). For common GFP, UUU was used for each Phenylalanine, because it had the highest frequency. For rare GFP, UUC was used. It is hypothesized that commonly occurring codons will have faster elongation rates than degenerate rare codons because cells have become optimized through evolution to efficiently translate proteins necessary to their survival.

A second approach to optimization was conducted by taking the results of another recent project, in which all the genes (coding DNA sequences) of *E. coli* were divided into five groups based on the naturally occurring TIR, from lowest to highest [29]. Then, the codon usage profile of

each group of genes was statistically analyzed to determine whether a codon is slow or fast. A fast codon was defined as one with high correlation between TIR and its frequency. Otherwise, it was a slow codon. This principle is demonstrated in Figure 8.

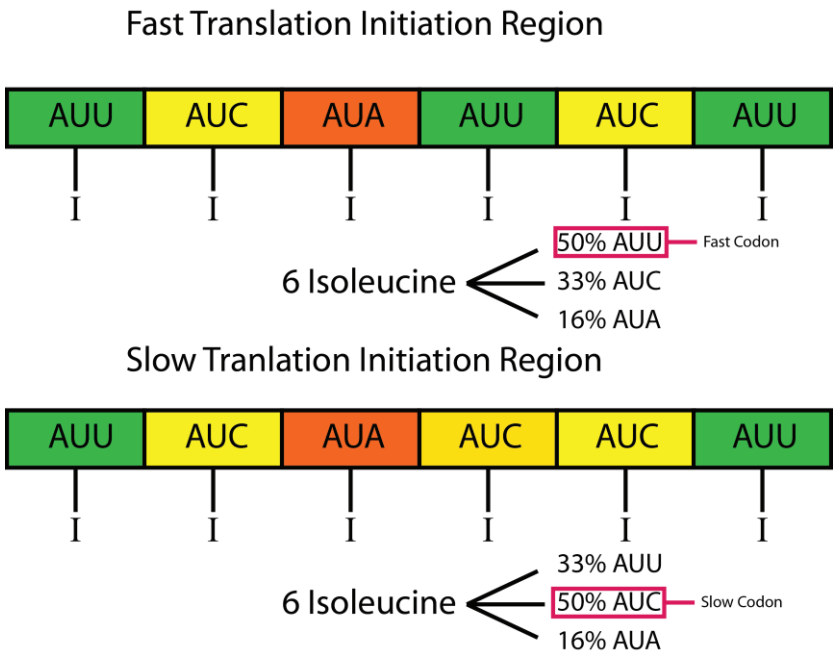


Figure 8: Fast and slow codons identified using codon bias in different TIR regions

This is similar to the common and rare distinction, but is more specific, as coding regions of the genome with low TIR could code for proteins where high expression (and thus fast translation elongation) is not necessary. By analyzing the codon usage profile of individual regions of the genome based on TIR, the distinction between fast and slow codons is made. A simplified example of this analysis is presented in Figure 9.

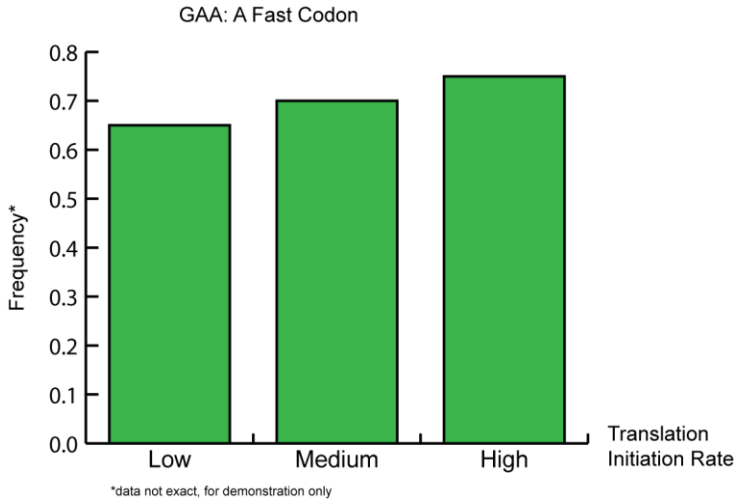


Figure 9: Codons with positive correlation between TIR and frequency are fast codons

It is hypothesized that the groups of CDS with high TIR will hold more "fast" codons, which will lead to higher translation elongation rate and thus higher protein expression, whereas the slow regions will hold more "slow" codons leading to lower expression. The overall analysis of the *E. coli* K12 genome allowed identification of all fast codons, slow codons, and codons with no statistically significant correlation between TIR and frequency (neither fast nor slow), and this is shown in Figure 10. Using this approach, two more coding sequences were constructed, one with only "fast" codons, and one with only "slow" codons.

This approach is similar to existing optimization methods based on codon adaptation index (cAI), but instead of defining fast codons as those with positive correlation between frequency and overall protein expression level, fast codons were defined as those with positive correlation between frequency and TIR. All fast and slow codons are identified in Figure 10.

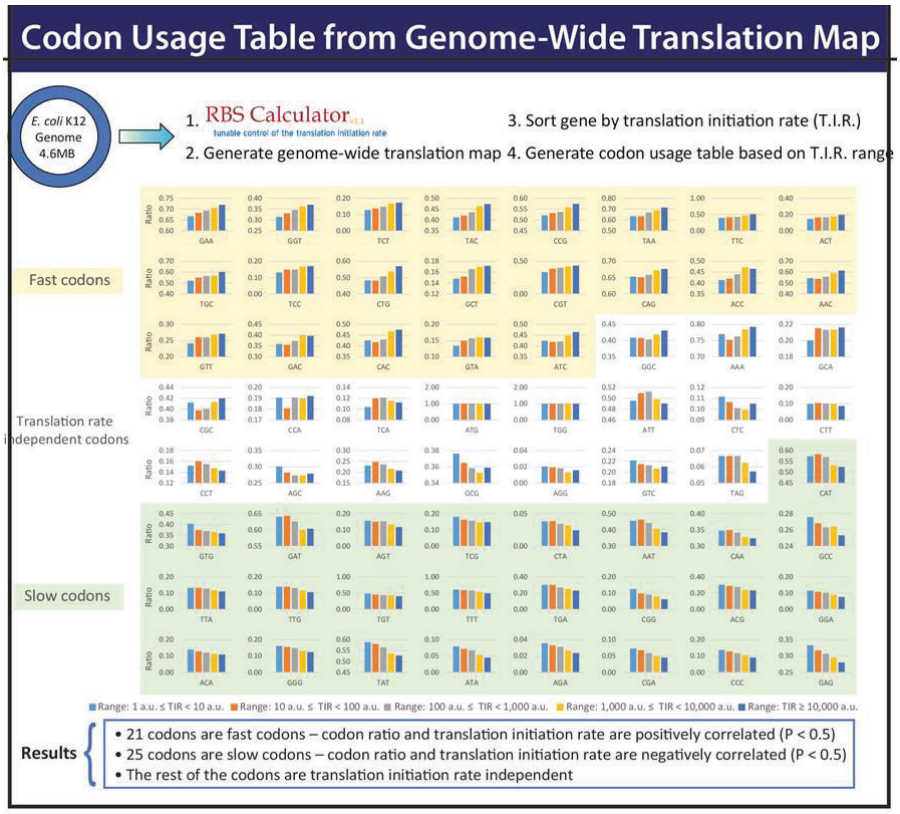


Figure 10: All fast and slow codon were identified [29]

In another project, researchers experimentally determined the time taken by a ribosome to add an amino acid to a growing polypeptide chain. This is known as the "insertion time" for that codon [30]. Using this data, a table of the insertion times for each codon was compiled, and is shown in Table 2.

Table 2: Table of Codon Insertion Times [30]

	CODON	AMINO ACID	Insertion Time (ms)	CODON	AMINO ACID	Insertion Time (ms)	CODON	AMINO ACID	Insertion Time (ms)	CODON	AMINO ACID	Insertion Time (ms)
U	UUU	Phe (F)	136	UCU	Ser (S)	55	UAU	Tyr (Y)	53	UGU	Cys (C)	75
	UUC	Phe (F)	195	UCC	Ser (S)	246	UAC	Tyr (Y)	77	UGC	Cys (C)	109
	UUA	Leu (L)	157	UCA	Ser (S)	106	UAA	Stop	11	UGA	Stop	12
	UUG	Leu (L)	50	UCG	Ser (S)	96	UAG	Stop	19	UGG	Trp (W)	168
C	CUU	Leu (L)	260	CCU	Pro (P)	143	CAU	His (H)	296	CGU	Arg (R)	28
	CUC	Leu (L)	204	CCC	Pro (P)	197	CAC	His (H)	222	CGC	Arg (R)	35
	CUA	Leu (L)	186	CCA	Pro (P)	237	CAA	Gln (Q)	179	CGA	Arg (R)	34
	CUG	Leu (L)	35	CCG	Pro (P)	134	CAG	Gln (Q)	231	CGG	Arg (R)	397
A	AUU	Ile (I)	97	ACU	Thr (T)	55	AAU	Asn (N)	109	AGU	Ser (S)	85
	AUC	Ile (I)	128	ACC	Thr (T)	153	AAC	Asn (N)	161	AGC	Ser (S)	127
	AUA	Ile (I)	128	ACA	Thr (T)	178	AAA	Lys (K)	76	AGA	Arg(R)	190
	AUG	Met (M)	266	ACG	Thr (T)	129	AAG	Lys (K)	102	AGG	Arg (R)	461
G	GUU	Val (V)	26	GCU	Ala (A)	39	GAU	Asp (D)	77	GGU	Gly (G)	35
	GUC	Val (V)	208	GCC	Ala (A)	415	GAC	Asp (D)	116	GGC	Gly (G)	49
	GUA	Val (V)	73	GCA	Ala (A)	83	GAA	Glu (E)	57	GGA	Gly (G)	324
	GUG	Val (V)	42	GCG	Ala (A)	44	GAG	Glu (E)	36	GGG	Gly (G)	81
	U			C			A			G		

It is theorized that codons with longer insertion times will lead to lower overall translation elongation rates and thus lower the maximum translation rate capacity of a protein. Using these results, a coding sequence was constructed that contained only the slowest insertion time (SIT) codon in each degenerate set. For example, if the amino acid needed was Phenylalanine, the codons UUU and UUC were available. The codon UUC was used, as its insertion time of 195 ms was greater than the 136 ms insertion time for UUU.

In total, five total sGFP coding sequences were constructed based on the results of three distinct optimization methods, and the summary of variant coding sequences is presented in Table 3.

Table 3: Summary of Variant sGFP coding sequences

Variant Name	Number	Optimization Criteria
Rare	1	Only codons that are the rarest degenerate codon for an amino acid based on all coding sequences in the genome
Common	2	Only codons that are the most common degenerate codon for an amion acid based on all coding sequences in the genome
Fast	3	Only codons that are more frequent in higher TIR regions of the genome than in lower regions, essentially, common codons in highly translated parts of the genome
Slow	4	Only codons that are more frequent in lowerTIR regions of the genome than in higher regions, essentially, common codons in rarely translated parts of the genome
Slow Insertion Time	5	Only codons that are predicted to have the slowest insertion time of their degenerate set

To design the genes a custom script was created which replaces all degenerate codons in a gene with the desired codons, for example, replacing all rare codons with common degenerate codons or all slow codons with fast degenerate codons. This is provided for use in future projects in Appendix A: Script for Optimizing Genes.

All coding sequences were designed so that there would be no difference between the amino acid profile of the variant sGFP and the original sGFP. This ensured that each gene resulted in the expression of the same protein. However, due to the optimization procedure there was significant difference in the genes at the codon level. In fact, genes designed using orthogonal criteria (rare *or* common, fast *or* slow) showed no similarity except for start codons, stop codons, and the amino acid tryptophan, which is specified by only one amino acid. This is visualized in Figure 11.

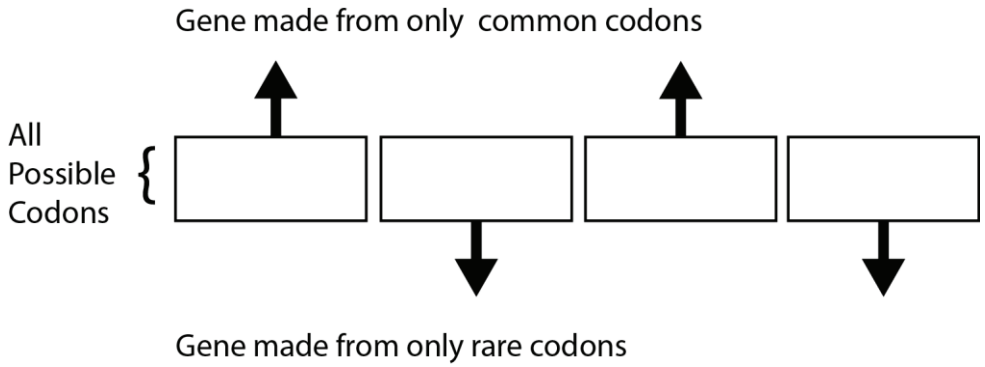


Figure 11: Genes designed with orthogonal criteria have minimal commonality

It some cases the variant sGFPs were designed based on non-exclusive criteria, so there exist some instances where more than one of the variants use the same codons for a particular amino acid. For example, slow sGFP uses UUC whenever Phenylalanine is needed, as does the slow insertion time sGFP.

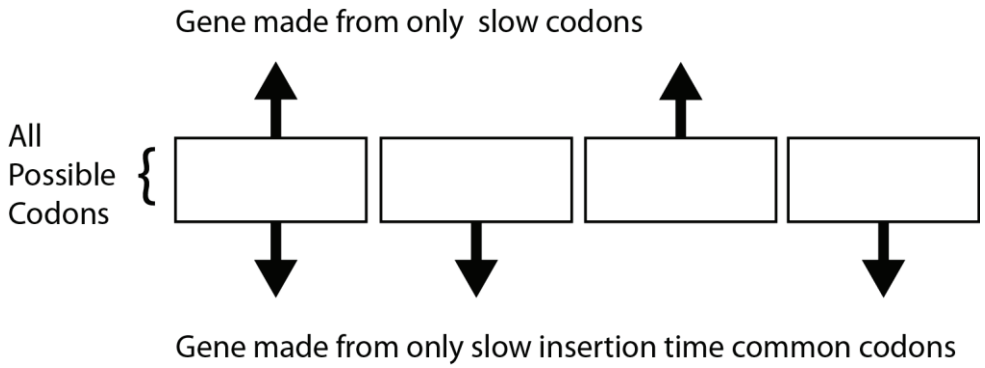


Figure 12: Genes optimized using non-orthogonal criteria have some commonality

The complete designation of each codon as rare, common, fast, slow, or slow insertion time is summarized in Table 4.

Table 4: Complete set of codons used in each variant sGFP

Amino Acid	**Rare Codon**	**Common Codon**	**Fast Codon**	**Slow Codon**	**Slow (Insertion Time) Codon**
Met	ATG	ATG	ATG	ATG	ATG
Trp	TGG	TGG	TGG	TGG	TGG
Phe	TTC	TTT	TTC	TTT	TTC

Thr	ACT	ACC	ACT	ACA	ACA
Ile	ATA	ATT	ATC	ATA	ATA
Leu	CTA	CTG	CTG	TTG	CTT
Val	GTA	GTG	GTT	GTG	GTC
Ser	TCA	AGC	TCT	TCG	TCC
Pro	CCC	CCG	CCG	CCC	CCC
Ala	GCT	GCG	GCT	GCC	GCC
Tyr	TAC	TAT	TAC	TAT	TAC
His	CAC	CAT	CAC	CAT	CAT
Gln	CAA	CAG	CAG	CAA	CAG
Asn	AAT	AAC	AAC	AAT	AAC
Lys	AAG	AAA	AAA	AAG	AAG
Asp	GAC	GAT	GAC	GAT	GAC
Glu	GAG	GAA	GAA	GAG	GAA
Cys	TGT	TGC	TGC	TGT	TGC
Arg	AGG	CGT	CGT	CGA	AGG
Gly	GGA	GGC	GGT	GGG	GGA

From the table, the number of instances where the same codon was used to specify any given amino acid can be determined, for any comparison of two sGFP variants. There is a maximum of 20 instances of similarity, in the case of identical genes, and a minimum of 2, as Methionine is always specified by ATG, and Tryptophan by TGG. This is shown in Figure 13.

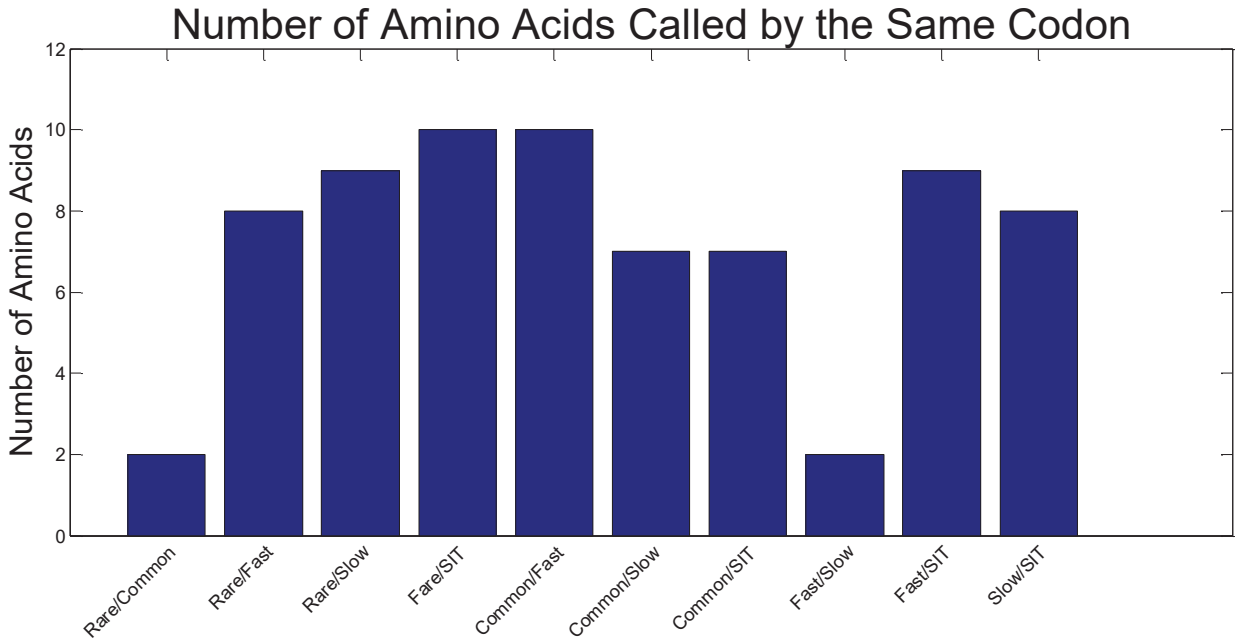


Figure 13: Comparison of sGFP coding sequences. The sGFP coding sequences that were used in this project have between two and 10 instances of amino acid commonality.

Another method of comparing sGFPs is by determining the number of instances of the same codon appearing in the same location in the gene. For example, if the same codon is used at position 30 in both Fast sGFP and Common sGFP, this would be defined as one instance of similarity. This comparison (expressed as a percentage of total number of codons in the gene) is presented in Figure 14.

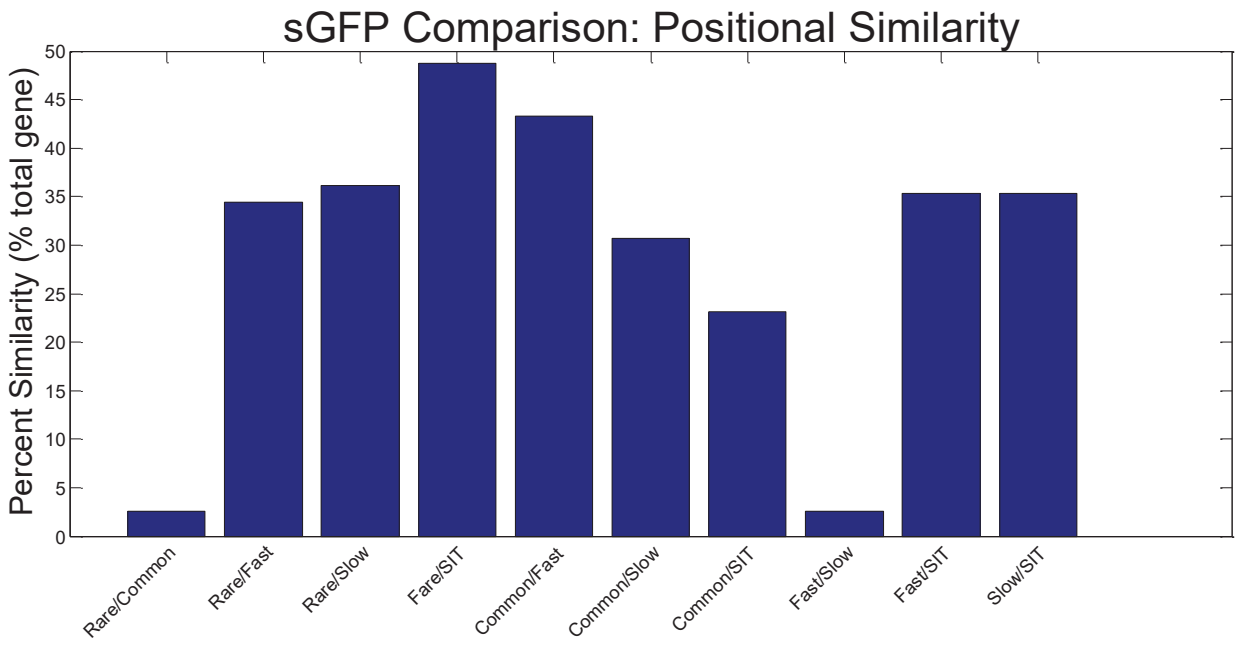


Figure 14: Positional comparison of optimized GFPs. Coding sequences have between 2.5% and 49% commonality. Percent similarity is calculated as percent of total instances where the same nucleotide is present at the same position in both coding sequences.

The variant genes were also compared by computing their codon adaptation index [31], web tool available at http://genomes.urv.cat/CAIcal/. and total predicted amino acid insertion time [30], tool available in: Script for Calculating percent similarity between sGFPs.

The cAI is defined from zero (poorly adapted to *E. coli*) to one (perfectly adapted to *E. coli*), and it is predicted that a gene with a higher cAI will benefit from efficient expression. Total amino acid insertion time is calculated by summing the times for each codon in the each variant sGFP. It is predicted that genes with lower total time will also benefit from efficient expression. These plots are shown in Figure 15.

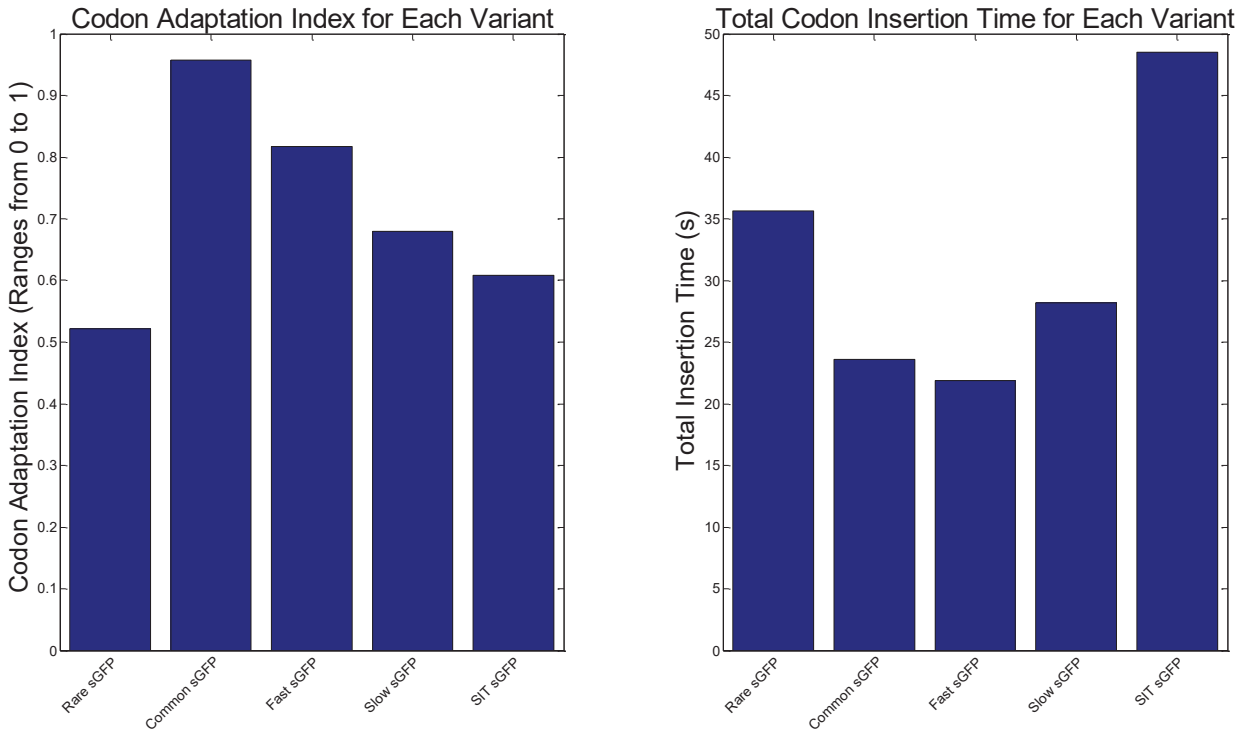


Figure 15: Comparison of cAI and total codon insertion time for the variant sGFPs

These comparisons show what is expected. First, the highest cAI is for Common sGFP, which was optimized to maximize this parameter, and the lowest is Rare sGFP, optimized to minimize this parameter. Similarly, insertion time is low for Common and Fast sGFPs, and highest for SIT sGFP, which was optimized to maximize this parameter. This comparison also reinforces the hypothesis that efficient genes such as Common and Fast sGFPs will have higher expression than the inefficient ones Rare, Slow, and SIT sGFP, as they are superior by the quantifiable metrics

of cAI and insertion time. This also confirms the use of the data on codon preference that was used in design [28], with the results of a different group [31].

Leader Sequence Design

Central to this project is the idea that maximum translation rate capacity is reached in the expression of a protein even as TIR is increased. Since TIR is increased by modifying the RBS, an RBS library was developed where with sequences of increasing TIR. Since design of an RBS is dependent on the first 60 base pairs of a protein coding sequence, it was necessary to develop a "dummy" coding sequence to be the same for all variant GFPs, or else it could not be ensured that an accurate and broad spectrum of TIR would be covered. This sequence was placed directly upstream of the CDS of each sGFP and was called the "leader sequence."

Since the purpose of the leader sequence was to ensure that a broad range of TIR was sampled but not to slow down translation elongation, it was designed with several considerations. First, it was 60 base pairs in length, the maximum amount of nucleotides downstream of an RBS shown to influence TIR. Second, it was designed using only codons that were both fast and common. This was to ensure that the translation of the leader would not become the rate limiting step in translation of the sGFP. Third, no stop codons were present in the leader in any reading frame. In order to reduce secondary structure formation in the leader sequence, there were no instances of three nucleotides in a row which form three hydrogen bonds. This means that it had no instances of GGG, GGC, CCC, or any combination of G and C for three consecutive letters, as this is the only combination of amino acids capable of forming secondary structures strong enough to prevent the ribosome from translating the mRNA. The junctions between the end of the leader sequence and the beginning of the coding sequences were checked using Vienna RNA [32,33]

(http://www.tbi.univie.ac.at/RNA/), a program that determines the extent of RNA structures that are formed, and it was ensured that there were no particularly stable structures that were predicted.

The leader was designed to have a diverse amino acid profile, in order to prevent any potential tRNA depletion or rate decrease from non-cognate tRNA interactions. Lastly, the leader was checked to ensure it did not contain any restriction sites that were to be used in cloning. The leader sequence was flanked with restriction sites that could be used when the RBS library was ligated into the constructs. A schematic of the leader sequence and surrounding features is shown in Figure 16.

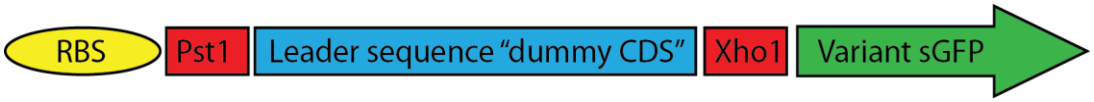


Figure 16: The leader sequence

RBS Design:

In order to accurately span a wide range of translation initiation for the variant GFPs, a library of Ribosome Binding Sites was designed that was ligated into the pFTV vector ahead of the leader sequence that preceded each coding sequence. A library of ribosome binding sites can be included in one degenerate RBS sequence, that is, a sequence that contains several degenerate letters, for example R (A or G), Y (C or T) , or N (A, T, C, or G). Within this single degenerate sequence are multiple distinct sequences, each with different TIR. Through rational design, a library can be designed and packaged in a dRBS that contains a desired number of sequences spanning a desired range of TIR on a proportional scale.

Since translation initiation is dependent on the 60 base pairs of DNA following the RBS it was necessary to ensure that this region of DNA following the RBS was the same for each variant sGFP construct. Since the coding sequences were different, this was accomplished by using a

uniform “leader sequence” that was attached to each sGFP construct directly upstream of the coding sequence.

Once the leader sequence was designed it was possible to design the RBS library using the ribosome binding site calculator. The calculator uses several inputs. First, the pre-sequence is any DNA that precedes the RBS, and although not absolutely necessary to the calculation, including this information improves accuracy of the calculations by providing more information for the calculation of the ΔG terms that make up the statistical thermodynamic equation which relates overall free energy change to translation rate (Equation 2). The “Pre Sequence” used for this design was the six nucleotide restriction site of Sac1, the restriction enzyme used to ligate in the RBS, preceded by the series of 20 nucleotides upstream of the restriction site.

The next input is the Protein Coding Sequence, which is the DNA following the RBS. This is the first DNA to be translated, and thus must begin with a start codon. The purpose of using a leader sequence was to standardize this region across all constructs.

The input for constraints is used to set how long of a sequence may be generated and if there are any features that cannot be mutated by the calculator. In this case, immediately upstream of the calculator is the Sac1 restriction site, which was included, followed by a space of 24 “N’s,” which are interpreted by the calculator as mutable nucleotides. Finally, the restriction site for Pst1 was included, as this was the other site that would be needed to ligate the dRBS into the backbone.

The next input is the Range of Translation Initiation Rates, which were set from 1 to 250,000 au. The library resolution was set to maximum (approximately 25 to 50 sequences), and *E. coli* K12 dh10b selected as the organism of interest. The current version of this software is the Ribosome Binding Site Calculator V2.0 [34], available at (https://salislab.net/software) The inputs used for the dRBS design in this project are summarized in Table 5.

Table 5: Inputs to RBS calculator

Input	**Specification**
Pre-sequence	TCTAGAGACTGAATTCAACGGAGCTC
Protein Coding Sequence	ATGCACAAAACTGTTCGTGCTGTTCGTCAGAAAGTTCACAAATC TACTGTTCAGACT
Constraints	GAGCTCNNNNNNNNNNNNNNNNNNNNNN CTGCAG
Minimum TIR	1.0
Maximum TIR	250,000.0
Resolution	25-50 sequences
Organism	Escherichia coli str. K-12 substr. DH10B (ACCTCCTTA)

Initial attempts used 30 N's, followed by the Pst1 site sequence for the field "Initial RBS Sequence with Optional Constraints," but this approach was soon modified as no libraries were able to create a sufficiently high TIR, with most maximizing at approximately 50,000 au. To test whether a high enough TIR was possible for the leader sequence that had been designed, the forward engineering mode of the Ribosome Binding Site Calculator was used. This mode determines a single sequence with a desired TIR, and allows a goal of "maximize" to be selected.

Using the same inputs for leader sequence and pre sequence that had been used for the RBS library calculations, and a TIR goal of "maximize", several sequences were compiled that displayed sufficiently high TIR. These high TIR sequences were then used as an initial condition for the more computationally intensive RBS library calculator by placing them into the "Initial RBS Sequence with Optional Constraints" field instead of N's. Using this approach, a viable RBS library was generated, and is shown in Figure 17.

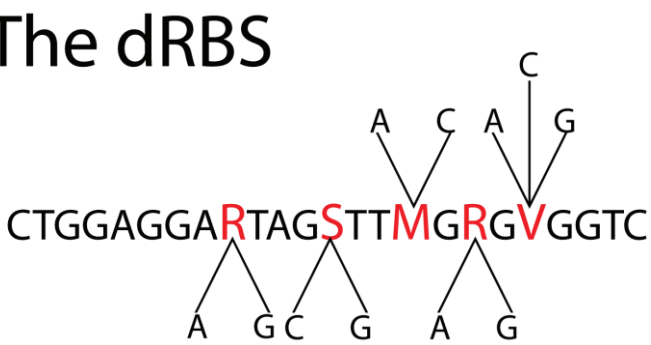


Figure 17: The dRBS library sequence. A library of ribosome binding sites was designed for this project and packaged in a single degenerate sequence.

The designed dRBS contained five degenerate letters, with four specifying one of two possible nucleotides, and the other specifying one of three, thus the number of possible sequences in the library was $2^4 \cdot 3^1 = 48$ total sequences. These sequences are plotted by ascending predicted TIR in Figure 18.

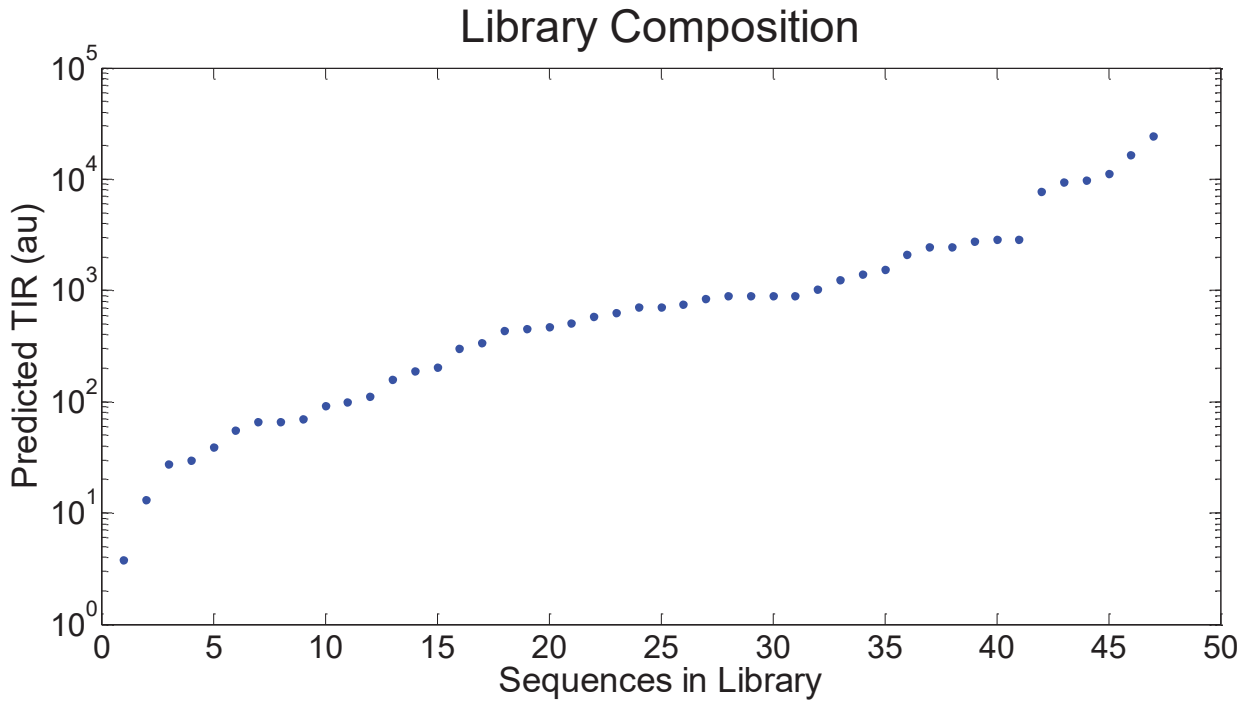


Figure 18: TIR for each sequence in the RBS library

A different visualization of the RBS library that was used in this project is shown in Figure 19, where the 48 total sequences in the library are shown on five histograms displaying the sub

ΔG terms that are calculated by the RBS calculator, as well as one histogram for the summation of these terms, ΔG_{total}. This shows which free energy terms are the same for all sequences in the library, as well as gives a range for the thermodynamic terms that did change from sequence to sequence.

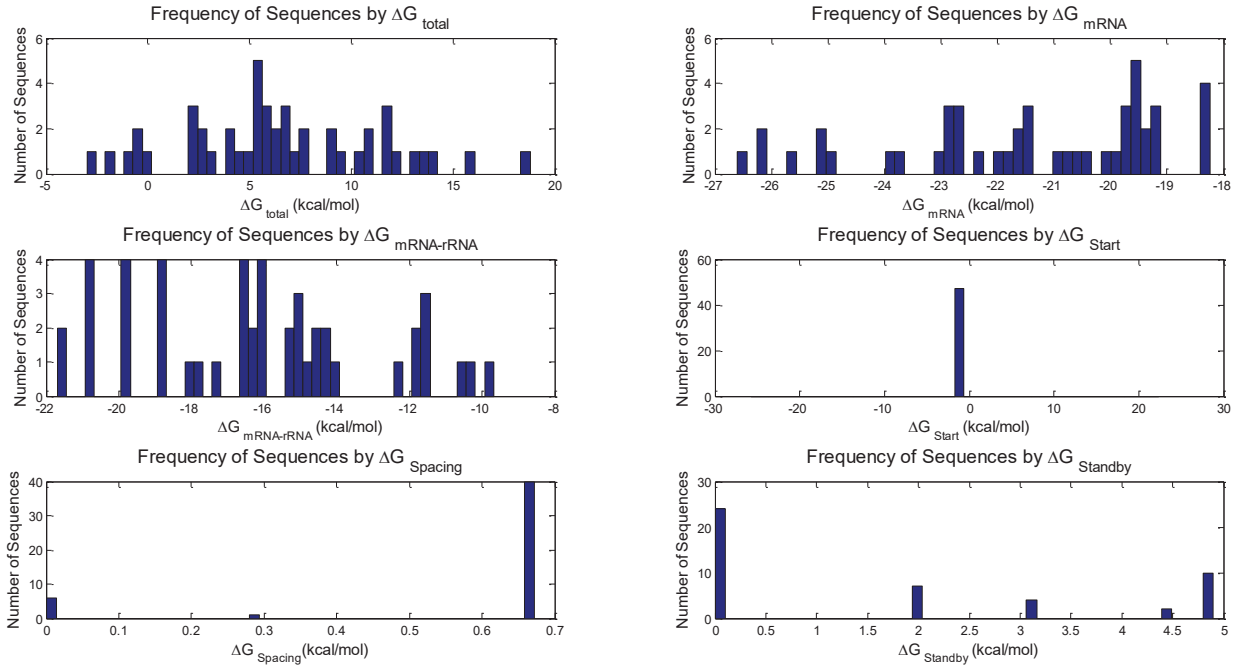


Figure 19: RBS library sub-term analysis. Histograms show the frequency of RBS sequences in the library by the values of Delta G for all sub terms in the RBS calculator (RBS calculator V2.0)

First, this shows that in the case of ΔG_{start} all of the sequences in the library have the same energetic value. This is because the free energy change from hybridization of the start codon to the first tRNA (Fmet) does not change as long as the same start codon is used [12]. The next term is $\Delta G_{spacing}$. This is an energy penalty that becomes increasingly severe as spacing between the start codon and the Shine-Dalgarno seqeunce deviates from the optimal spacing [14]. For all sequences in the library there are only three values of $\Delta G_{spacing}$ predicted. This is likely because the calculator is interpreting several positions within the dRBS as potential consensus sequence sites (due to the presence of the degenerate nucleotides), and some of these result in non-optimal spacing. The

terms $\Delta G_{standby}$ and $\Delta G_{mRNA\text{-}rRNA}$ vary more widely. In the case of $\Delta G_{mRNA\text{-}rRNA}$, which is the energy released from hybridizing the mRNA and 16S rRNA, the value is very sequence specific, and the change of some of the degenerate letters in the dRBS can greatly impact the result. In the case of $-\Delta G_{standby}$, which is defined as the energy needed to unfold the standby site (four nucleotides upstream of the site of 16SrRNA-mRNA interaction), the value is also found to vary widely between the sequences that make up the RBS library. The last sub term, ΔG_{mRNA}, is the free energy associated with the initial, folded state of the mRNA.

The histogram displaying frequency of sequences by their ΔG_{total} shows that the sequences in the library span a wide range of free energy change, and thus should be effective in producing a wide dynamic range in TIR. It also shows that the largest number of the sequences are roughly in the middle of the range, but that there are sequences present that have both very negative ΔG_{total} (high TIR) and very positive ΔG_{total} (low TIR). The library that was used in this project was designed and synthesized *De Novo*, but the pre-experiment analysis lead to confidence that it would allow the expression of the optimized coding sequences to be measured across a wide enough range of TIR to notice any plateau in expression.

The library was ordered from IDT as a degenerate Ribosome Binding Site sequence, which was then ligated into the vector containing each individual variant GFP. Because it was not known which specific RBS was inserted into each vector, colonies were picked and sequenced in order to know exactly which RBS was taken.

Cloning:

In order to test the effect of optimizing the sGFP coding sequence, an expression construct was designed. Several considerations were taken when choosing a vector into which the variant GFPs would be inserted. First, any vector to be used would need to have sufficiently high copy

number in *E. coli* K12 dh10B so that expression of the protein could be measured. Second, any incompatibility with the vector and the GFPs, such as long repetitive sequences that could influence the success of PCR or cloning, would need to be avoided, and were searched for by inspecting the sequences of several candidate vectors in Ape plasmid editor. Another possibility for incompatibility were restriction enzyme recognition sites that would be used to insert the RBS into the construct, or to ligate the RBS into the constructs. These could not be present anywhere in the vector backbone. This consideration was met by searching all sequences in Ape plasmid editor using the "find enzymes" feature. Third, an antibiotic resistance marker was necessary so that colonies could be grown on plates without contamination.

The vector pFTV (2.8 kb) was chosen, due to its adherence to all the design constraints as well as its availability. Vector pFTV confers resistance to Chloramphenicol. Another consideration that was made when choosing pFTV was that it had been used successfully in previous experiments, and this led to confidence that cloning would be successful. For a detailed description of all protocols mentioned in the following section, see Protocols.

Construction of the vector was accomplished in several steps. First, the five variant genes were constructed as double stranded DNA (gblocks) by Integrated DNA Technologies (https://www.idtdna.com/pages/products/genes/gblocks-gene-fragments). They were resuspended from the lyophilized gblocks and amplified using PCR. A photograph of the gel where these amplified genes were run is shown in Figure 20.

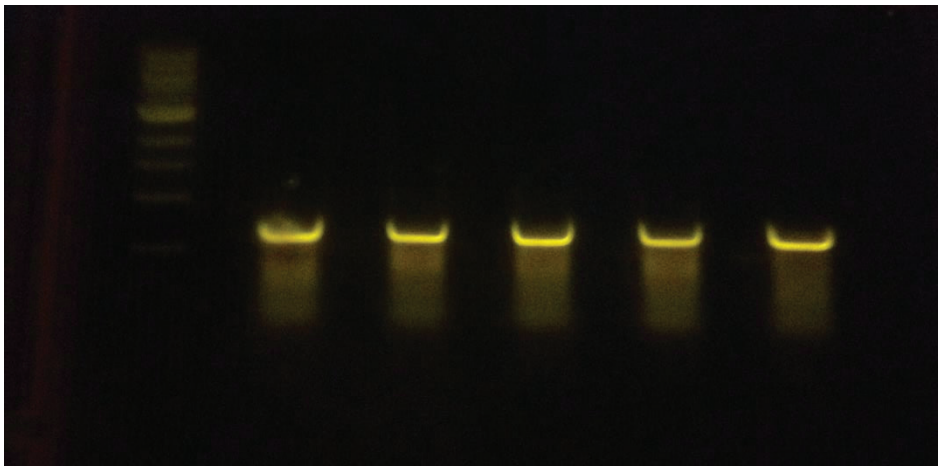


Figure 20: Gel electrophoresis shows bright bands for each optimized gene

Next, the pFTV backbone was processed using Inverse PCR, a process in which the old insert gene was removed and three new restriction sites were introduced. Inverse PCR is visualized in Figure 21.

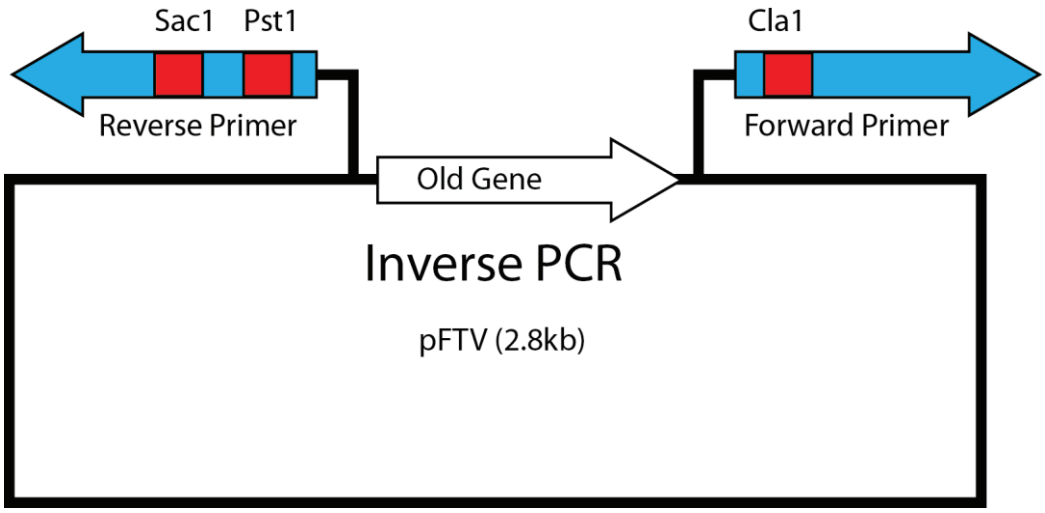


Figure 21: Inverse PCR

After inverse PCR, the backbone contained three new restriction sites. The new sites were chosen to match the designs of the dRBS and the variant genes, which would allow them to be inserted via digestion and ligation. The products of inverse PCR were processed using gel

electrophoresis to collect the backbone and discard the old insert. These products are shown in Figure 22.

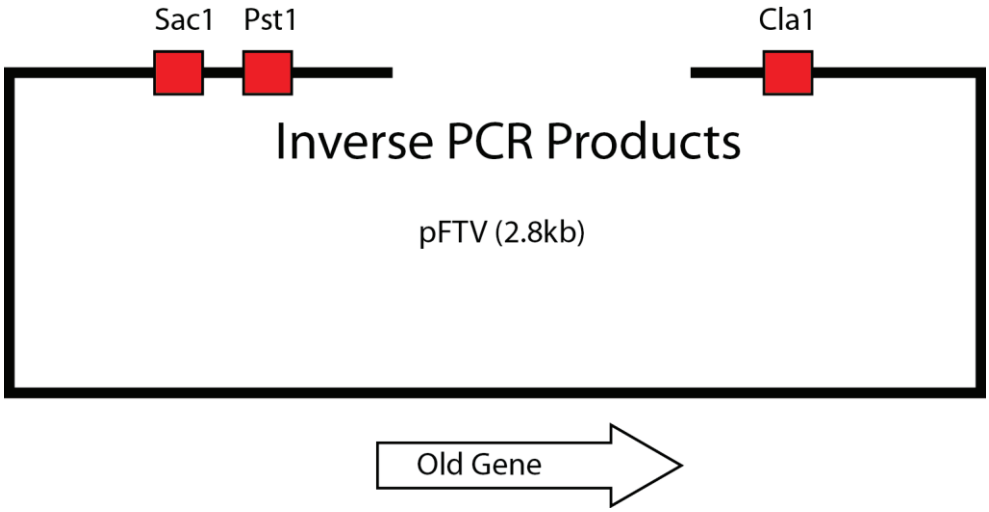


Figure 22: Inverse PCR products

After inverse PCR, the resuspended and digested gblocks containing the leader sequence and variant GFPs were inserted. This was accomplished using a ligation reaction, which is illustrated in Figure 23.

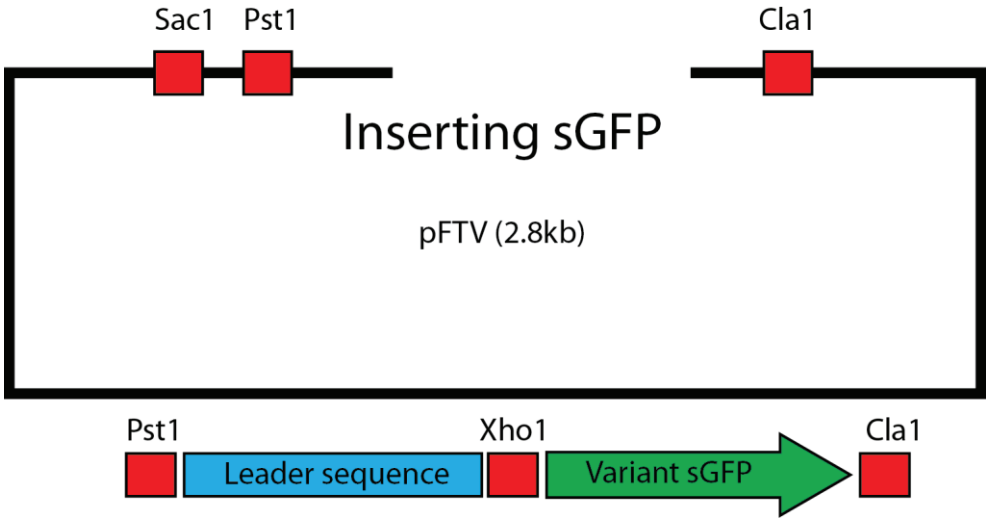


Figure 23: Inserting sGFPs

After insertion of the coding sequences, the backbone was once again digested, this time to make room for the dRBS. The dRBS sequence was constructed using annealed oligos for some

of the constructs, but later was constructed using PCR assembly due to difficulty with the annealed oligo method. In the PCR assembly case, the dRBS was digested back to the restriction sites that flanked it, and then ligated into the backbone. This is visualized in Figure 24.

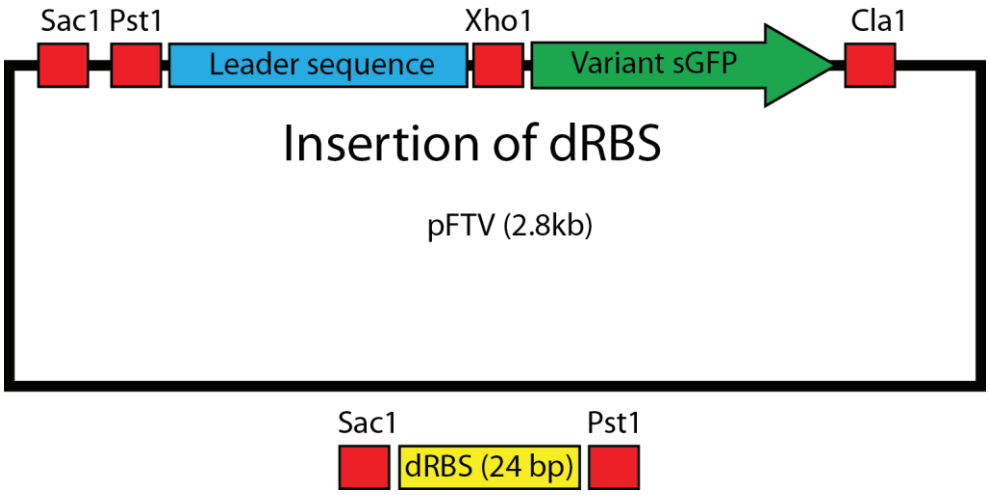


Figure 24: Insertion of dRBS

Finally, the completed construct was introduced into cells through electroporation. The finished product is illustrated in Figure 25.

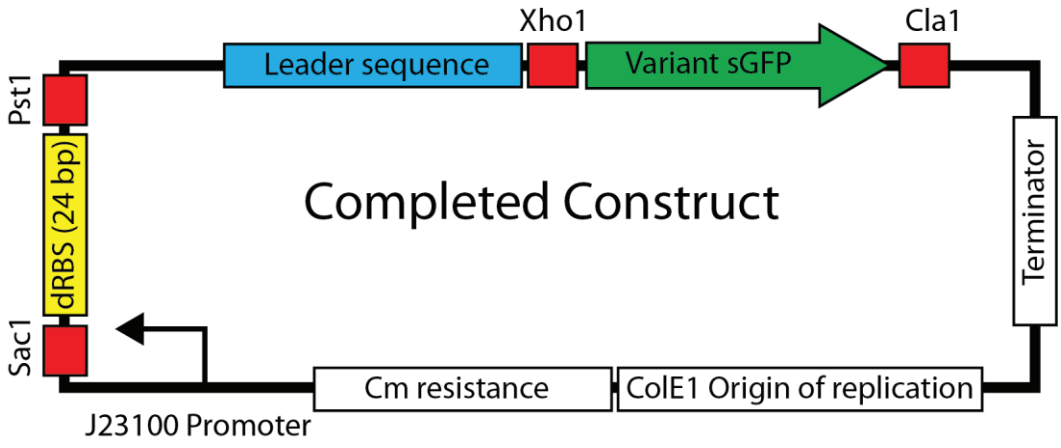


Figure 25: The finished construct

Data Collection

Following cloning, bacteria were transformed with the completed constructs and then plated on Chloramphenicol (Cm) agar plates. After an incubation period, colonies from these plates were selected and used to inoculate the TECAN overnight plate. The TECAN was then run for three plate cycles for each construct, and data were collected and pre-processed.

Once the average fluorescence per cell (FPLC) was determined, some of the TECAN wells were used to inoculate cultures for sequencing and preservation as cryogenic stock. Sequencing was used to determine which RBS in the library was being used by the cells in that well, and finally the results of FPLC were assigned to not only the coding sequence that was expressed, but also the specific RBS in the construct. Using this approach, the results of the experiment were collected.

Protocols

In this section, the protocols used in this project are provided. These protocols are from the Salis Laboratory for Metabolic Pathway Engineering at Penn State University (http://salislab.net/), but they have been modified for specificity to this project.

1. **Minipreparation**

- **Summary**: Minipreparation is the process by which plasmids from bacteria are extracted from living cells grown in laboratory cultures. This can be executed at a variety of scales, and miniprep is the scale chosen for research applications. The general process of miniprep is to weaken the cell wall, lyse the cell, precipitate out cellular components such as lipids and proteins, and remove chromosomal DNA as well as RNA [35]. The plasmid DNA is then be bound to a silica matrix in a small scale column where it is purified through washing. These wash steps remove salts and any remaining cellular components, as well as exchange buffer so that the plasmid is eluted into water in the last step. This procedure follows the

E.Z.N.A. plasmid DNA Mini Kit 1 from the Omega Bio-Tek company, and references the stock reagents that are standard with the kit, available at: http://omegabiotek.com/store/product/plasmid-mini-kit-1-q-spin/

- **Procedure**:
    1. A culture was inoculated and grown for 10 – 14 hours in a shaker at 37°C at 300 revolutions per minute (rpm).
    2. The culture was centrifuged at 10,000 g for 5 minutes at room temperature.
    3. The culture medium was decanted being careful not to discard any of the cell pellet. The tube was tapped upside down onto a paper towel to ensure the pellet was dry.
    4. 250 µL Solution I was added and mixed by pipetting up and down to thoroughly re-suspend the pellet. Solution I was used in order to weaken the cell wall and break down RNA (contains RNase A).
    5. After the pellet was re-suspended, the contents were added to a clean 1.5 mL microcentrifuge tube.
    6. 250 µL Solution II was added and the microcentrifuge was gently inverted several times until a clear lysate formed. The solution was allowed to incubate for 2 minutes at room temperature. Solution II was used in order to lyse the cell. Vigorous mixing was avoided to prevent shearing of chromosomal DNA.
    7. 350 µL Solution III was added to the microcentrifuge tube, which was then inverted several times until a flocculent white precipitate formed. Solution III was used to bind proteins and lipids from the cell for removal.
    8. The solution was centrifuged at maximum speed for 10 minutes at room temperature.

9. The supernatant was transferred to a HiBind DNA Mini Column, being careful not to pick up any of the cellular debris. The HiBind DNA Mini Column was then attached to the vacuum manifold.
10. The vacuum was turned on to draw the supernatant through the mini column.
11. 500 µL HBC Buffer was added to the column and vacuumed through.
12. 700 µL DNA wash buffer was added to the column and vacuumed through.
13. Step 12 was repeated.
14. The column was transferred to a 2 mL collection tube and then centrifuged for 2 minutes at maximum speed to dry the column matrix.
15. The column was transferred to a clean 1.5 mL microcentrifuge tube.
16. 30 µL sterile deionized water was added to the column and incubated for 3 minutes.
17. The microcentrifuge tube with the column was centrifuged for 1 minute at maximum speed to elute the plasmids.
18. The column was discarded and the concentration of the DNA in the microcentrifuge tube was measured using the NanoDrop spectrophotometer and recorded on the tube.
19. The plasmid product was stored at -20° C.

2. Restriction Digest

- **Summary**: Restriction digest is the process by which an enzyme is used to cleave DNA at a target site. This experiment is based on the natural functionality in living bacteria of restriction endonucleases, enzymes which cleave DNA at certain recognition sites in order to protect the host cell from foreign DNA infection [11]. In order to employ restriction digest on synthetic DNA, these known sites are included in the design of the DNA at desired

locations. Restriction digest is often utilized to cut plasmid DNA at specific locations so that new genetic material can be introduced [3]. Restriction sites are usually a hexanucleotide, or six nucleotide sequence, and it is important to choose restriction sites that are not found elsewhere in the vector if cloning is being attempted; otherwise it is possible that the plasmid will be cleaved in multiple locations [36]. Restriction digests are also used in order to detect if a plasmid contains desired genes. This is conducted by choosing an enzyme that will cut the plasmid into fragments of predicted size, which are then run through gel electrophoresis to see if the experimentally observed bands match the predicted lengths of DNA [36].

- **Procedure**:
    1. The necessary restriction enzymes were identified using A Plasmid Editor (APE), and cross checked with other sites present in the vector to be sure that no cuts were made at non-desired locations.
    2. The following reagents were added to a micro PCR tube, in the order they are depicted in Table 6 (restriction enzyme added last).

Table 6: Reagents in Restriction Digest

Reagent	**Quantity**
10x CutSmart Buffer*	5 µL
PCR Product	2 µg
Restriction enzyme 1	1 µL
Restriction enzyme 2	1 µL
dd H20	To 50µL

3. The New England BioLabs Double Digest Finder was used to determine the temperature for the digestion to be incubated, available at (https://www.neb.com/tools-and-resources/interactive-tools/double-digest-finder).
4. The PCR tube was mixed by flicking, then centrifuged on the table top micro centrifuge in order to break any bubbles.
5. The PCR tube was placed into the C1000 Thermal Cycler for incubation of 6-9 hours at the temperature prescribed by the double digest finder.

   *The optimal buffer may differ based on the restriction enzymes that are chosen. Consult the Double Digest Finder.

3. Polymerase Chain Reaction (PCR)

- **Summary**: The PCR reaction is a technique used to amplify desired sections of DNA molecules. The product DNA can then be used in a variety of downstream applications such as cloning. This technique relies on the in vitro exponential amplification of DNA by iterating the cycle of denaturing, annealing, and extension [37]. Under the tightly controlled temperatures of PCR, the time for DNA to replicate is reduced from hours (in vivo) to less than five minutes [38]. During each cycle, the amount of DNA is doubled, and thus after roughly 30 cycles a sample can be generated that is virtually purely composed of the target DNA. This protocol is adapted from the suggested protocol by New England BioLabs Inc.
- **Procedure**:
    1. A bucket of ice was gathered to ensure that all materials remained cold during setup.
    2. The following components were added to a micro PCR tube, in the order they are depicted in Table 7 (DNA polymerase enzyme added last).

Table 7: Reagents in PCR Reaction

Reagent	**Amount (50 µL Reaction)**
Nuclease-free water	32.5 µL
5x Phusion HF Buffer	10.0 µL
10 mM dNTPs	1.0 µL
10 µM Forward Primer	2.5µL
10 µM Reverse Primer	2.5 µL
Template DNA	(1-5) ng, generally ~ 0.5 µL
Phusion DNA Polymerase	0.5 µL

3. The S1000 thermal cycler was programmed with the protocol described in Table 8.

Table 8: Thermal Cycler Settings for PCR

Step	**Temperature**	**Time**
Initial Denaturation	98° C	30 seconds
Denature	98° C	5-10 seconds
Annealing	45-72° C	10-30 seconds
Extension	72° C	15-30 seconds per kilobase (kb)
Repeat Denature, Annealing, and Extension Cycle 25-35 times		
Final Extension	72° C	5 -10 minutes
Hold	4° C	Forever

4. The micro PCR tube was removed from the thermal cycler and then processed by PCR clean up.

4. DNA purification

- **Summary**: DNA purification is conducted after digestion and ligation reactions. The objective of this experiment is to purify and concentrate DNA by removing salts and enzymes as well as concentrating the product of digestion and ligation reactions [36]. By using a small amount of ddH20 for the elution of the final product it is possible to create product at a higher concentration than if no concentration step was included, since most of the reactions in PCR reaction tubes are volume limited. Product at a higher concentration is more useful for downstream applications, in particular cloning, where it may lead to more colonies being present after transformation [36].

- **Digestion Clean up Procedure**:

1. 250 µL DNA Binding Buffer was added to the PCR tube with the digestion/ligation/PCR products and was mixed by pipetting up and down.
2. This solution was transferred to a Zymo spin column, which was transferred to a 2 mL collection tube.
3. The tube was centrifuged at 14,000 rpm for 1 minute.
4. The flow-through was discarded and 600µL DNA wash buffer was added to the center of the column.
5. The tube was centrifuged at 14,000 rpm for 1.5 minutes to dry the column matrix.
6. Steps 4 and 5 were repeated.

7. The flow through was discarded and then the column was centrifuged at 14,000 rpm for 2 minutes. This was done to ensure minimal residual wash buffer was removed, as ethanol in wash buffer can interfere with downstream applications [36].
8. The column was transferred to a clean 1.5 mL microcentrifuge tube and 10 µL ddH20 was added to the center of the column.
9. The column and tube assembly was incubated for 1 minute at room temperature.
10. The assembly was centrifuged at 14,000 rpm for 2 minutes and the concentration of DNA in the flow through measured using the NanoDrop Spectrophotometer.
11. Products were either used immediately used or stored at -20 °C.

- **Ligation Clean Up Procedure**

1. 200 µL DNA Binding Buffer was added to the PCR tube with the digestion/ligation/PCR products and was mixed by pipetting up and down.
2. This solution was transferred to a Zymo spin column, which was transferred to a 2 mL collection tube.
3. The tube was centrifuged at 14,000 rpm for 1.5 minute.
4. The flow-through was discarded and 600µL DNA wash buffer was added to the center of the column.
5. The tube was centrifuged at 14,000 rpm for 1.5 minutes to dry the column matrix.
6. Steps 4 and 5 were repeated.

7. The flow through was discarded and then the column was centrifuged at 14,000 rpm for 2 minutes. This was done to ensure minimal residual wash buffer was present, as ethanol in wash buffer can interfere with downstream applications.
8. The column was transferred to a clean 1.5 mL microcentrifuge tube and 4 µL ddH20 was added to the center of the column.
9. The column and tube assembly was incubated for 1 minute at room temperature.
10. The assembly was centrifuged at 14,000 rpm for 2 minutes and the concentration of DNA in the flow through measured using the NanoDrop Spectrophotometer.
11. The products were either used for transformation immediately or stored in -20° C freezer for later use.

5. Gel Electrophoresis

- **Summary**: Gel electrophoresis is a separation process during which DNA is introduced into a gel substrate while an electric field is superimposed. The DNA bears a negative charge, and is thus attracted to the positive terminal which is set at one end of the gel [36]. Separation is possible because the rate at which DNA molecules migrate through the gel varies by the size of the molecule (number of base pairs), and by understanding this relationship, different size DNA molecules can be introduced simultaneously in a mixed solution and then separated after a time is allowed for their different migration rates to lead to different positions in the gel. To make the DNA bands visible, a dye is introduced, and the gel is observed under blue light. After excising the gel bands and purifying, the desired DNA can be recovered and stored. This protocol is modified from the AddGene agarose gel electrophoresis protocol, available at https://www.addgene.org/plasmid-protocols/gel-electrophoresis/.

- **Setup Procedure**:

    1. An appropriate size gel cast and comb were chosen for the experiment (large size holds 150 mL gel and small size holds 50 mL gel).
    2. Cast and comb were rinsed thoroughly with distilled water to remove any residue from previous experiments.
    3. 150 mL 1xTAE buffer was heated in the microwave in a flask.
    4. An agarose gel solution was made by mixing 1.2 g agarose (makes 0.8% agarose solution) with the TAE buffer and allowing the solution to cool until it was no longer too hot to touch.
    5. 1.5 µL GelStar dye (10,000X) was added to the agarose solution.
    6. The gel was cast by adding the agarose and dye solution to the mold.
    7. The comb inserted into the mold to create wells for inserting DNA.
    8. After solidifying, gel tray spun so that one end pointed towards the positive terminal, and one toward the negative terminal.
    9. The gel mold was filled with 1XTAE buffer until the gel was covered.
    10. 20 µL appropriate size DNA ladder added to one of the side wells in the gel.
    11. Purple dye (6X) was added to the DNA containing micro PCR tubes and the ladder so that the volume of the DNA or ladder was five times the volume of the dye (ex. 10 µL dye added to 50 µL PCR reaction tube. 4 µL dye added to 20 µL DNA ladder).
    12. Gel was loaded by gently pipetting the mixed DNA/dye samples into the appropriate wells. Position of each sample recorded.
    13. Terminals were connected and device ran at 110 Volts until sufficient separation had occurred for excision of bands without cross-contamination.

14. Gel was photographed for record keeping.

- **Excision Procedure**:

1. A number of clean, 1.5 mL microcentrifuge tubes were massed using an electronic balance equal to the number of bands of DNA that were to be excised.
2. A scalpel was cleaned with ethanol and used to excise each desired band from the agarose gel, under blue light for visibility.
3. Each band of DNA was transferred to a labeled microcentrifuge tube.
4. Each microcentrifuge tube containing gel was massed.
5. The mass of each gel band was found by subtracting the mass of the empty microcentrifuge tube from the mass of the filled tube.
6. Three masses of ADB (agarose dissolving solvent) were added to each mass of agarose excised. For example, if gel mass was 0.25 g (250 mg), 750 µl ADB (750 mg) were added. This protocol estimates that the density of ADB is 1000 kg/m^3.
7. All gel containing tubes were incubated at 60° C in an oven until the gel band was totally melted.
8. Each solution as transferred to a Zymo-Spin column in a collection tube.
9. Each tube as centrifuged for 60 seconds at 1100 rpm. The flow through was discarded.
10. 600 µl DNA wash buffer to the column and centrifuge for 30 seconds. The flow through was discarded.
11. Step 10 was repeated.
12. The tube was centrifuged for 1 minute to dry the column. Flow through was discarded.

13. The column was transferred to a new 1.5 mL microcentrifuge tube.
14. 20 µl sterile water was added to the center of the column.
15. The column was incubated at room temperature for 1 min.
16. The column and microcentrifuge tube assembly was centrifuged for 1 min.
17. The concentration of DNA in the solution was measured using nanodrop. The column was discarded.
18. The microcentrifuge tube was transferred to -20°C freezer for storage. Procedure modified from Zymo Research, INC. http://www.zymoresearch.com/dna/dna-clean-up/gel-dna-recovery/zymoclean-gel-dna-recovery-kit

6. Ligation

- **Summary**: Ligation is the process by which a short segment of DNA (the insert) is added into another, larger section of DNA (the backbone) using the presence of compatible digested restriction enzyme sites on both molecules [36]. During digestion, DNA is cut, revealing a "sticky end," at each cut site that is complementary to the sticky end produced by a digestion on the molecule that is desired to be joined [3]. During ligation, these sites are joined in order to connect the two molecules and create a new plasmid that carries the desired insert. The completed plasmid can then be purified (See section: DNA purification) and used for downstream applications such as transformation.
- **Procedure**:
    1. The following reagents were added to a micro PCR tube, in the order they are presented in Table 9.

Table 9: Reagents for Ligation Reaction

Reagent	**Quantity**
1000 fmole** insert	Varies
10 fmole cut plasmid	Varies
T4 Ligase Buffer	2µL
T4 Ligase	1µL
ddH20	To 20 µL

**Note: DNA (fmoles) = DNA (ng/µL) * 1515/ (# bp in DNA)

2. The reaction was incubated for 10 minutes at room temperature.
3. The ligated product was purified immediately. Short run times and immediate purification was conducted to prevent generation of non-specific ligation products (for example re-circularized backbone).

7. Annealing of Oligonucleotides

- **Summary**: Oligonucleotides (Oligos) are short, single or double stranded DNA molecules which can be designed and chemically synthesized at low cost, making them an attractive method for introducing short functional regions of DNA into larger constructs [36]. They are used in this project for the purpose of constructing a degenerate ribosome binding site library. This library was designed using the Ribosome Binding Site Calculator and constructed by ordering two single stranded oligonucleotides from Integrated DNA Technologies (https://www.idtdna.com/site) and then annealing them together. By this method, a double stranded section of DNA containing degenerate letters was constructed, and was used as a library of ribosome binding sites.

- **Procedure**:

    1. Each oligonucleotide was resuspended to 100 μM in ddH2O. 5ul of each oligo and 90 ul ddH2O were transferred to a clean 1.5 mL microcentrifuge tube to create a 5 uM solution of the two oligos.
    2. 500 mL water was boiled over a hot plate. The tube was placed in a float in the water so that the bottom of the tube, containing the oligo solution, was immersed in the hot water.
    3. The tube was left in the boiling water for 3 minutes. The heat was then shut off and the water was allowed to cool on the bench (~6 hours), being careful not to disturb the beaker.
    4. Annealed oligos were diluted by 10X by adding 10 μL annealed oligo solution to 90 μL ddH2O. The final concentration of the diluted annealed oligo solution was 0.5 μM (500fmoles/μL).
    5. The product was stored at -20°C.

8. PCR Assembly

- **Summary**: Attempts at cloning using the dRBS that was constructed by the method of annealed oligonucleotides (see Annealing of Oligonucleotides) frequently resulted in too few colonies to proceed to fluorescence measurement. It was theorized that using PCR assembly to construct the degenerate ribosome binding site library would be a suitable alternative method, as it is known that the PCR reaction can be used to efficiently generate high fidelity product [36] even when the participating molecules are designed to contain degenerate letters. In assembly PCR, a short primer is used to extend a long strand and create double stranded DNA. The product is then submitted for a normal “rescue” PCR to simply amplify the DNA, and then purified, digested, and used in cloning.

- **Procedure**:

5. A bucket of ice was gathered to ensure that all materials remained cold during setup.
6. The following components were added to a micro PCR tube, in the order they are depicted in Table 10 (DNA polymerase enzyme added last).

Table 10: Reagents in Assembly PCR Reaction

Reagent	**Amount (50 µL Reaction)**
Nuclease-free water	Up to 50 µL
5x Phusion HF Buffer	10.0 µL
10 mM dNTPs	1.0 µL
Long Strand (10 µM resuspension)	2.5µL
Short Strand (Primer) (10 µM resuspension)	2.5 µL
Phusion DNA Polymerase	0.5 µL

7. The S1000 thermal cycler was programmed with the protocol described in Table 11.

Table 11: Thermal Cycler Settings for Assembly PCR

Step	**Temperature**	**Time**
Initial Denaturation	98° C	30 seconds
Denature	98° C	10 seconds
Annealing	71° C	30 seconds
Extension	72° C	15-30 seconds per kilobase (kb)

Repeat Denature, Annealing, and Extension Cycle 15 times		
Final Extension	72° C	5 -10 minutes
Hold	4° C	Forever

8. The micro PCR tube was removed from the thermal cycler and then processed by PCR clean up.

9. Transformation

- **Summary**: For creation of cells with synthetic DNA, new genetic material must be added to the cells themselves. This is done by transforming cells with the desired DNA. A procedure called "electric-field-mediated membrane permeabilization," or electroporation, is used, during which a transient pulse of voltage is created across a culture containing live cells [3]. It is theorized that this results in the formation of temporary pores in the cell wall and which allows the desired DNA to enter [36].
- **Procedure**:

1. Electrocompetent ells were taken from storage at -80° C and put on ice.
2. A fresh electroporation cuvette was also placed on ice.
3. 2-3 μL concentrated ligation product was added to the electrocompetent cells and carefully mixed by pipetting to prevent air bubble formation.
4. Cells and DNA were transferred to an electroporation cuvette.
5. Metal sides of the cuvette were wiped perfectly dry, and the cuvette was placed into the electroporator.
6. Electroporator settings were placed at 2500 Volts.

7. The “pulse” button was selected twice in rapid succession. The time constant was recorded (between 4.6 and 5.6 for optimal efficiency).
8. 600 µL SOC medium added to cuvette and mixed by pipetting. Solution transferred to a 5 mL test tube for incubation. Growth medium without antibiotics for selection must be added to the cuvette immediately following electroporation as the cells are weakened by the procedure and need time to recover and begin synthesizing the antibiotic resistance enzymes that are coded by the new DNA.
9. Cells incubated at 37° C for 30 minutes. This incubation time depends on the antibiotic used for selection. Chloramphenicol requires 30 min (iGEM open protocols, available at http://parts.igem.org/Help:Protocols/Transformation).
10. 200 µL transformant broth was added to an agar plate. The solution was spread evenly using a flame sterilized hockey stick and plate spinner.
11. Plates were incubated in the oven at 37°C until the appearance of colonies (roughly 12 – 16 hours after plating).

10. Preparation of Electrocompetent Cells

- **Purpose**: Electroporation is one technique used to introduce DNA into bacterial cells. It is most commonly used in bacterial cells because electroporation is highly efficient at incorporating the new DNA into the bacterial cells that will then be used for cloning. Electrocompetent cells have a weaker cell wall than non-competent cells so that electroporation will have a higher efficiency.
- **Procedure**:
    1. 5 mL LB Miller medium was added to a test tube.
    2. 5 µL Streptomycin was added to the same test tube.

3. *E. coli* cells of the desired strain were used to inoculate the media. In this research, dh10B was used.
4. The cells were allowed to grow overnight in a shaker at 37°C and 300 rpm (until saturation was reached).
5. 100 mL LB Miller medium was added to 2 Erlenmeyer flasks (total 200 mL media)
6. 100 µL Streptomycin was added to each of the Erlenmeyer flasks.
7. The optical density (OD) of cells grown overnight culture was measured using nanodrop.
8. The Erlenmeyer flasks were inoculated with enough of the overnight media so that the OD in each Erlenmeyer flask was 0.01.
9. The two flasks were transferred to the shaker running at 30°C and 300 rpm. Cells were grown at 30°C instead of 37°C in order to alter the formation of the cell membrane.
10. Flasks were removed from the shaker once OD = 0.5 was achieved.
11. Approximately 50 1.5 mL microcentrifuge tubes were labeled and transferred to a box in the -80°C freezer.
12. The tabletop centrifuge was turned on and cooled to 4°C.
13. The contents of the Erlenmeyers flasks were transferred to 4x50mL Falcon Tubes, while making sure the tubes remained on ice.
14. The tubes were transferred to the pre-chilled centrifuge and centrifuged at 4500 rpm for 10 minutes.
15. The supernatant was decanted while being careful not to disturb the pellet.

16. 25 mL 10% glycerol solution was added to each of the tubes and used the gently resuspend the pellet.
17. The contents of the 4 x 25mL Falcon tubes were combined to create 2 x 50 mL Falcon tubes.
18. These two tubes were transferred back to the chilled centrifuge and centrifuged at 4°C and 4500 rpm for 10 minutes.
19. The supernatant was decanted and the pellets were resuspended in 25 mL 10% glycerol solution.
20. The contents of the 2 x 25mL Falcon tubes were combined to 1 x 50mL Falcon tube.
21. The tube was centrifuged again (making sure to balance the centrifuge) at 4°C and 4500 rpm for 10 minutes.
22. The supernatant was discarded.
23. The cells were resuspended with 2 mL 20% glycerol solution.
24. Quickly, the solution was aliquoted in 55 µL amounts into the pre-chilled centrifuge tubes, until all solution had been dispensed.
25. The cells were stored at -80°C fridge until use.

11. Preparation of Cryogenic Stock

- **Summary**: It may be desirable to preserve certain cell lines, often for their usefulness or simply to allow the possibility of conducting additional study in the future. Preserving a microorganism for long periods of time requires slowing down the cellular metabolism. This preserves the DNA from the possibility of mutation, thus preserving the genetic identity of the cell line. It also allows future research to pick up with essentially the same cells as research that occurred when the cell line was first stored. There are several methods

of preserving cell lines at low temperature, but the method used in this research was to suspend the cells in a glycerol solution, which does not form a crystal structure even at very low temperatures. The use of a crystal forming liquid, such as water, will break the cell walls as crystallization occurs [39].

- **Procedure**:

    1. Several cryo tubes were selected, and labeled with the following information: strain, initials of scientist, date of preparation, strain traits (or plasmids carried), number of strain if in a series or progression, medium used, antibiotic resistance of cells.

    Example:
    Dh10B
    CS
    5/4/2015
    pFTV+GFP4
    001
    SOC
    Cm

    2. The table top centrifuge was set to 4° C.
    3. Cell culture was transferred to 5 ml culture tubes.
    4. Tubes were centrifuged using table top centrifuge at 10,000 rpm for 5 minutes.
    5. Tubes were decanted near flame to ensure aseptic conditions.
    6. Pellets were resuspended using 750 µL appropriate medium.
    7. 750 µL resuspended cell containing medium was added to cryo tube.
    8. 750 µL 50% glycerol solution was added to the cryo tube.
    9. Cryo tubes were briefly vortexed.
    10. Cryo tubes were stored at -80° C.

12. Sequencing

- **Summary**: Sequencing is the process by which DNA molecules are analyzed to determine the order, or sequence, in which the individual bases Adenine (A), Thymine (T), Cytosine (C), and Guanine (G) occur. This is done to determine the composition of an unknown DNA molecule, the effectiveness of cloning, or as a method of screening for cells bearing a specific construct after a cloning procedure in which bacteria were transformed with multiple versions of a construct simultaneously. Sequencing relies on primers which bind to the desired construct at known locations, and then allow the construct to be "read" by instruments after a step analogous to PCR.

- **Procedure**:

    1. Cells of interest were grown overnight in a test tube containing 5mL LB Miller CM 50 medium. The antibiotic was added to select for cells bearing the desired construct.

    2. Plasmids were harvested from the cells using the procedure outlined in Minipreparation.

    3. 10 μL harvested plasmid solution for each desired sequencing reaction was added to a micro PCR tube. This assumes that the harvested DNA solution has concentration of at least 200 ng/μL DNA (see protocol: Measurement of DNA concentration).

    4. 5 μL appropriate sequencing primer at concentration 10 μM (working stock) sequencing was added to the micro PCR tube (see protocol: Primer Resuspension).

5. An order was created at Quintara Biosciences (http://www.quintarabio.com/) and label printed to be shipped with the sequencing order.

13. TECAN Fluorescence Measurement

- **Summary**: To measure the effectiveness of optimization methods, the expression of the fluorescent green protein that is coded for in the expression construct must be quantified. This is done by allowing samples to grow inside the TECAN M1000, a spectrophotometer that also serves as a monochromator and incubator. The TECAN is controlled by a Magellan software program, which is standardized so that samples can be run under nearly identical conditions each time. The TECAN functions by exciting cells grown in small wells with a laser at a certain wavelength, and then measuring the emission of light from those wells. The instrument must be calibrated with both the emission and excitation wavelengths of light, which are found to be 511 nm and 492 nm, respectively (Table of fluorescent proteins: http://nic.ucsf.edu/dokuwiki/doku.php?id=fluorescent_proteins).
- **Procedure**:
    1. A deep well plate is inoculated for overnight growth by carefully transferring 700 µL LB Miller medium and one colony from a plate to each deep well in the plate. The medium is prepared with Chloramphenicol 50 mg/ml to select for bacteria expressing the desired construct.
    2. The overnight plate is also inoculated with two wells housing only standard dh10b *E. coli* to serve as a control for the natural background emission of this cell line. These cells are not expressing the designed construct so they must be selected by using medium with Streptomycin.
    3. The deep well plate must also have some wells filled with un-inoculated medium. This allows for checking for cross contamination, because if these

wells appear turbid by the next morning it is likely that cross contamination has occurred on the plate (possibly agitation was too vigorous).

4. TECAN plate #1 was inoculated. TECAN plates are shallower than the deep well plate, have glass bottoms, and are stored in ethanol solution. Because of this, care was taken to ensure that the plates were sufficiently rinsed and dry, as residual ethanol will impede growth of the cells.
5. The TECAN plate was filled using 200 µL LB Miller CM50 medium, except for the cells to house the dh10b control, which were filled with 200 µL LB Miller Step 50 medium.
6. The TECAN plate was inoculated from the overnight deep well plate.
7. The overnight deep well plate was saved for creation of cryogenic stocks.
8. The TECAN protocol "iGEM 2014 GFP superfolder" was selected, and the plate was inserted into the instrument.
9. The optical density (OD) of TECAN wells was monitored, and when it reached 0.2, the plate was exchanged for a new plate, inoculated using the same method as the first TECAN plate.
10. Step 9 was repeated.
11. The data from the TECAN was recovered and processed to find the average fluorescence of each well. By sequencing these cells and finding which RBS was incorporate into their expression construct, the expression level data was then used to fill the space made by RBS strength, optimization scheme.

14. Measurement of DNA concentration

- **Summary**: In order to proceed through experimental steps where precise amounts of DNA are used, the concentration of DNA in solution must be measured. In order to measure DNA concentration in solution, an instrument called Nanodrop was used.
- **Procedure**:
    1. The setting "nucleic acids" was selected.
    2. The machine was blanked by pipetting 2.0 μL ddH2O onto the measurement platform and then selecting "blank."
    3. 1.0 μL purified, concentrated DNA was placed onto the measurement platform and then "measure" was selected.
    4. DNA concentration in ng/μL was recorded in the laboratory notebook, as well as on the tube containing the DNA of interest.

15. Measurement of Biomass Concentration

- **Summary**: In order to proceed through experimental steps such as creation of electrocompetent cells or plasmid harvest, the concentration of cells (biomass) in growth medium must be measured. Biomass is measured indirectly by measuring the optical density (OD) of cell cultures. This optical density can be understood as the degree of turbidity of the culture medium, and can be related to the biomass through established equations as long as the culture OD is below a threshold (above which the linear relationship between OD and biomass breaks down) [39]. This was conducted using an instrument called Nanodrop.

- **Procedure**:
    1. The setting "cell culture" was selected.

2. The machine was blanked by placing an optical density cuvette filled with sterile culture medium into the measurement well and then selecting “blank.”
3. Culture medium containing cells was diluted 10 fold by pipetting 900 µL sterile medium into a cuvette and then adding100 µL culture.
4. The dilute culture medium was added to the measurement well, and “measure” was selected.
5. The value of OD600 was recorded, as this is the optical density of the culture at 600 nm light, which is the wavelength used for calibration and relation to biomass.

16. gBlocks

- **Summary**: gBlocks Gene Fragments are chemically synthesized, double-stranded DNA. This is the same form as DNA in living cells, which means that gBlocks are compatible with most applications that require double-stranded DNA. In general, gBlocks are handled in the same way as linear, double stranded DNA. gBlocks Gene Fragments are delivered at 200 ng total mass lyophilized DNA, regardless of the length of the synthetic sequence.
- **Procedure**:
    1. Tubes containing gBlocks were centrifuged at 11,000 rpm for 30 seconds to ensure that the DNA pellet was at the bottom of the tube.
    2. 20 µL ddH2O was added to the tubes, for a final concentration of 10 ng/µL.
    3. Tubes were briefly vortexed and centrifuged.
    4. Resuspended gBlocks were stored at −20° C.

17. Primer Resuspension

- **Summary**: In order to proceed to PCR, primers shipped as lyophilized DNA must be resuspended and diluted to a known concentration. This allows consistent and correct

amounts of primers to be added to PCR reactions. It also ensures that stock solutions are protected for future use.

- **Procedure**:
    1. Tubes containing lyophilized primers were centrifuged for 30 seconds at 11,000 rpm to ensure the pellet was in the bottom of the tube.
    2. Primers were resuspended with 10 µl water for each nMol DNA in the tube. For example, a tube containing 38.2 nMol primer was resuspended by adding 382 µl H2O to create a100 µM primer stock solution.
    3. Master primer solutions at 100 µM were incubated at room temperature for 10 minutes, then mixed well before making working stock dilutions.
    4. Working primer solutions at 10 uM were created by dilution of 100 uM stocks. This reduces the number of freeze/thaw cycles that the master primer stock goes through and reduces the chances of contaminating the primary source of the primer. Master stocks were diluted 10 fold in a sterile microcentrifuge tube with ddH2O.
    5. Stock solutions and working solutions at -20° C.
    6. This procedure is modified from White, Resuspending PCR Primers, available at (http://fg.cns.utexas.edu/fg/protocol__resuspending_PCR_primers.html).

18. TECAN Data Analysis

- **Summary**: After measurement in the TECAN, the data for cell fluorescence were exported to Microsoft Excel and pre-processed. The purpose of this processing step is to select only data that corresponds to the log phase growth of the cultures in the TECAN. Equation 3 is used to correct the FLPC for the background fluorescence, as well as the fluorescence of

wild type *E. coli* K12 dh10b cells, and to correct for the baseline OD of the medium. This allows the effects of the optimized sGFP to be studied.

- **Procedure**:

$$FLPC = \frac{fl_{GFP}}{OD_{600}} = \left(\frac{fl_{sample} - fl_{medium}}{OD_{sample} - OD_{medium}}\right)$$

Equation 3: Calculation of FLPC

Where:

$FLPC$	=	Average Fluorescence per cell
fl_{GFP}	=	Fluorescence due to GFP
OD_{600}	=	Optical density of sample due to cells
fl_{sample}	=	Overall fluorescence of sGFP sample
fl_{medium}	=	Background fluorescence due to medium
OD_{sample}	=	Overall optical density of sGFP sample
OD_{medium}	=	Baseline optical density due to medium

Results

The primary result that was desired in this project was the expression level of each of the variant superfolder green fluorescent protein (sGFP) coding sequences. The gene for superfolder GFP was chosen for optimization as it is a reporter protein, that is, a protein that is easily assayed using fluorescence measurements taken by a spectrophotometer. Higher fluorescence indicates higher expression of the protein.

The experiment was designed so that each variant sGFP would be expressed using a library of ribosome binding sites in order to cover a large dynamic range of translation initiation rates. This way, a plateau in expression could be quantified by finding the approximate value of translation initiation rate (TIR) at which the slope of the expression curve became zero. This would be indicative of translation becoming the rate limiting step in protein synthesis. The expression level at which this occurs is known as the maximum translation rate capacity. By comparing the maximum translation rate capacity of each of the sGFPs, it could then be determined if any one optimization method resulted in a significantly higher maximum translation rate capacity, and could thus be used in future projects to ensure that protein expression can always be maximized.

It was hypothesized that the sequences which were optimized for only common codons or only fast codons (positive correlation between frequency and TIR in the genome) would have either a higher maximum translation rate capacity than the sGFPs optimized for only rare or slow codons, or perhaps no plateau at all.

There was difficulty in the cloning stage that required troubleshooting. Unfortunately, the coding sequence “Slow-optimized sGFP” was never successfully introduced into cells, and therefore the total number of variants was reduced to four.

Measurement of Average Fluorescence per Cell (FLPC) for the remaining four coding sequences was conducted using TECAN. Average FLPC for all colonies expressing Rare-optimized sGFP (but a variety of RBSs) is shown in Figure 26. Error bars of length ± one standard deviation in the FLPC results are shown. The control was E. coli K12, sub strain dh10B with no construct, which showed no fluorescence.

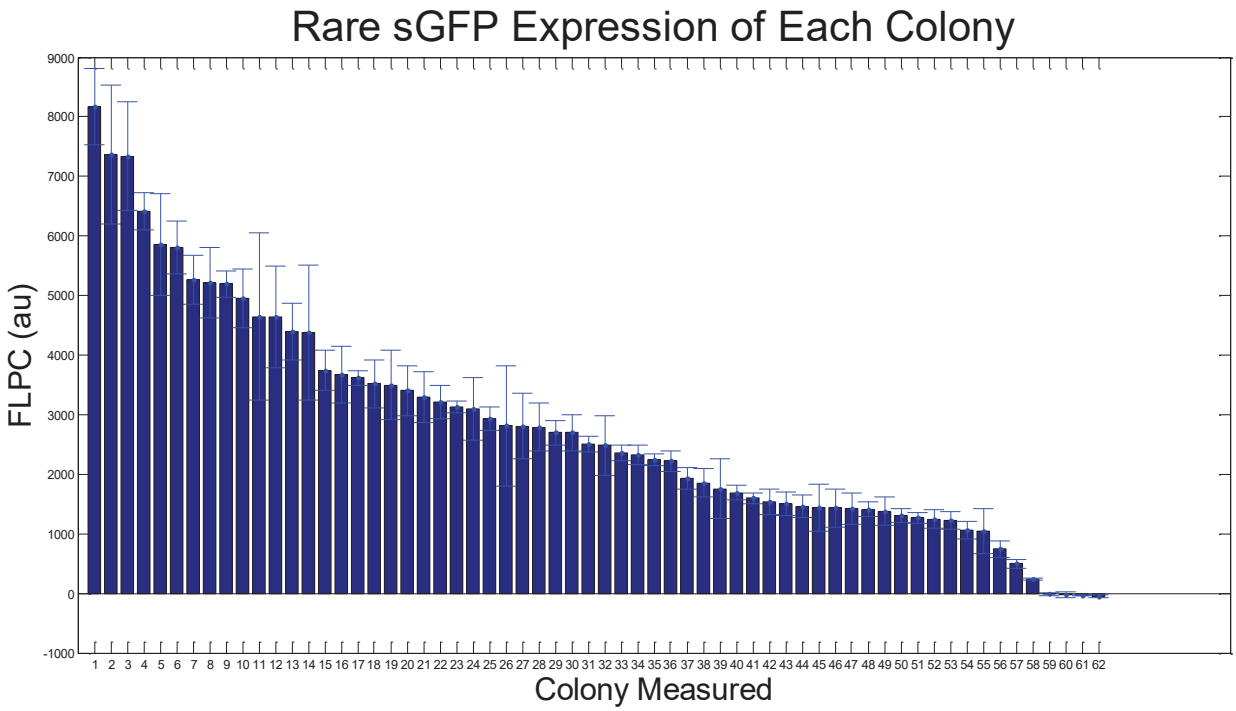


Figure 26: Ranked FLPC for Rare sGFP shows an expression range from (0 to 8,169)

The results in Figure 26 show each colony that was measured using TECAN. Sixteen colonies were sequenced, with most toward the higher range of expression. Cloning resulted in strains that all harbored the same coding sequence, but using different ribosome binding sites. Thus, the specific RBS used by each colony was determined by sequencing. In this screening process, it was discovered that several of the colonies had the same RBS. For example, approximately 16 colonies were sequenced for each coding sequence, but less than 16 distinct RBSs were verified from each batch because some had the same RBS. The results from TECAN for common-optimized sGFP are shown in Figure 27.

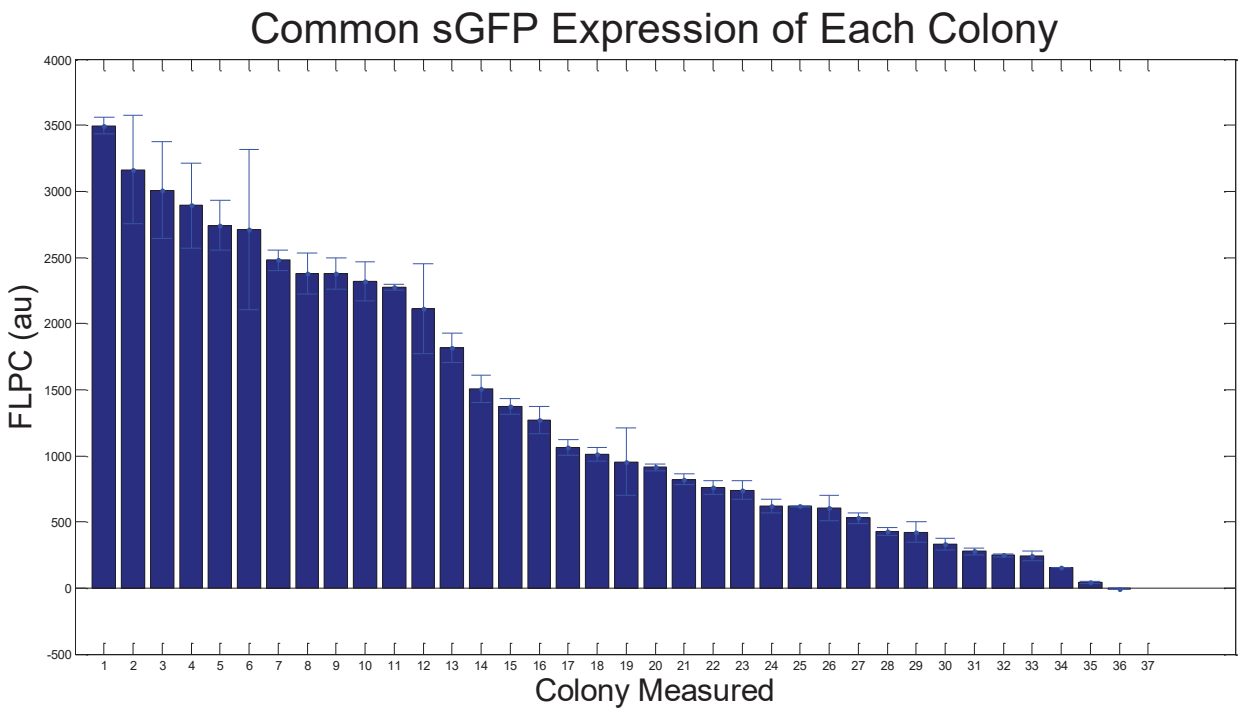


Figure 27: Ranked FLPC for Common sGFP shows an expression range from (0 to 3,642)

The results of the expression of Common-optimized sGFP were used to choose colonies for screening. Results for Slow insertion time (SIT)-optimized sGFP are shown in Figure 28.

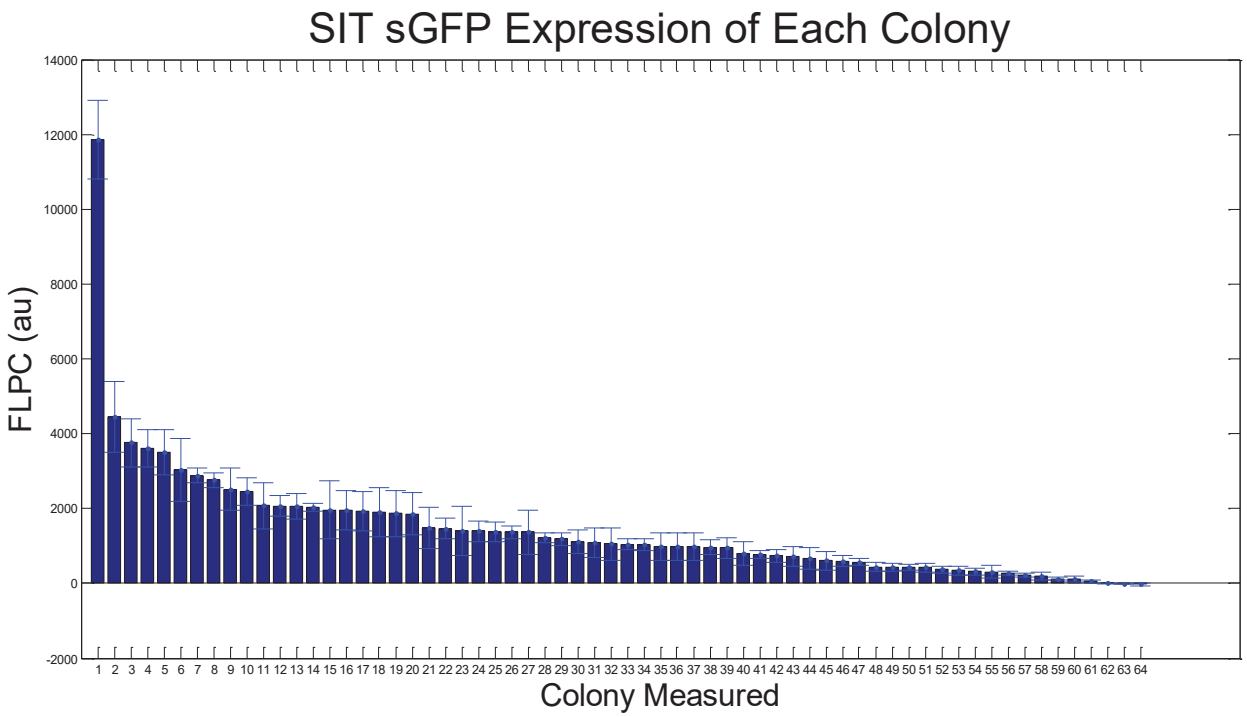


Figure 28: Ranked FLPC for SIT sGFP shows an expression range from (0 to 11,851)

The results for Fast-optimized sGFP are shown in

Figure 29.

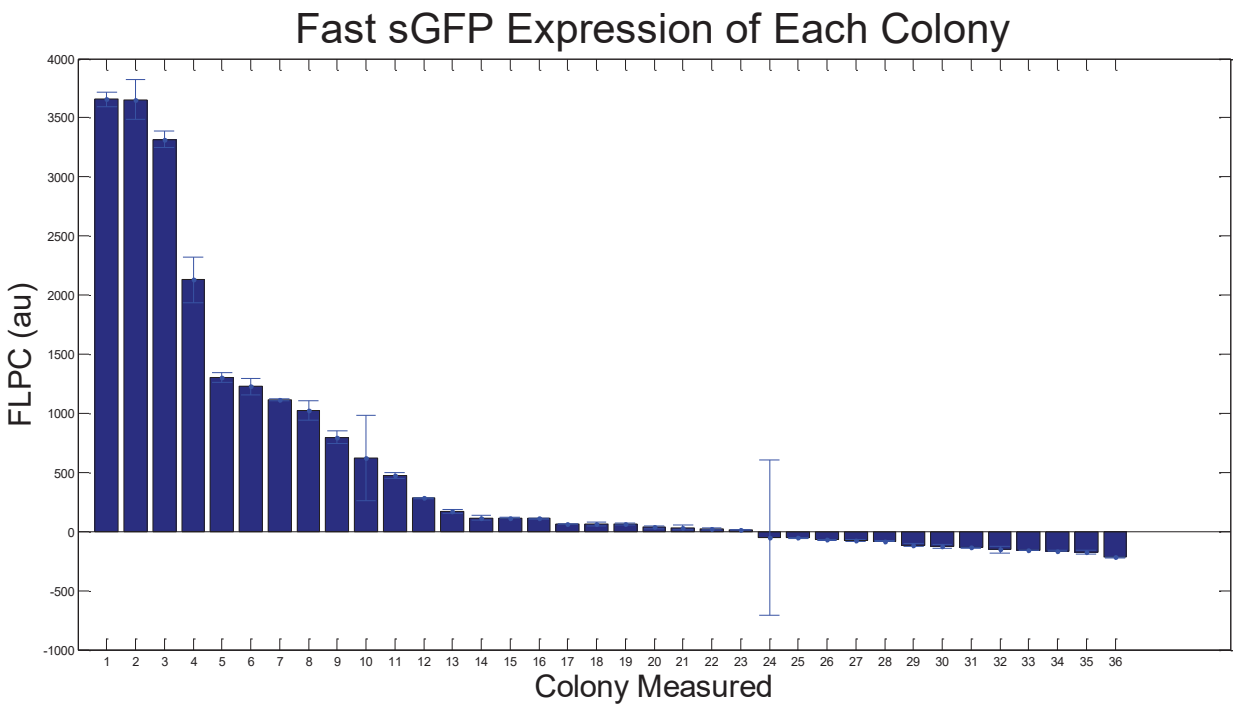


Figure 29: Ranked FLPC for Rare sGFP shows an expression range from (0 to 3,656)

After screening, sequencing, and matching the RBSs that were expressed in each colony to the library of RBSs that was designed, it was possible to search for maximum translation rate capacity plateaus by plotting the expression of each colony vs the predicted TIR of whichever RBS had been taken up by that colony. These results for Common, Fast, Rare, and SIT optimized sGFPs are shown in the following figures.

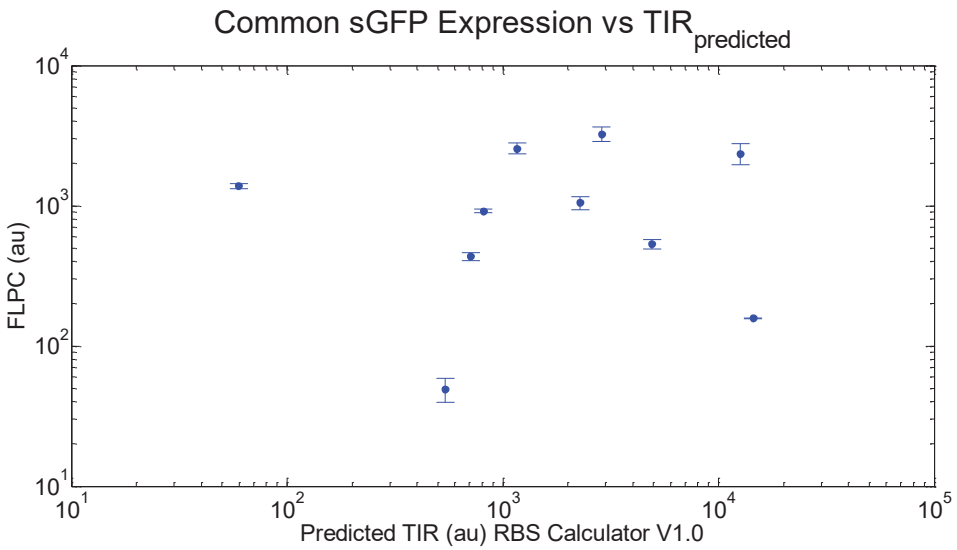


Figure 30: Maximum translation rate capacity analysis for Common optimized sGFP was conducted by plotting the average FLPC for each colony vs the predicted TIR of the specific RBS sequence that was incorporated by that colony.

Although insufficient data were collected to definitively show a plateau, there may be a leveling of expression at high TIR. It was predicted that by choosing only common codons in this optimization scheme the expression would increase to very high levels without plateauing at high TIR. This hypothesis cannot be discounted due to insufficient data.

Similar graphical representation of results are shown for Fast-optimized sGFP in Figure 31.

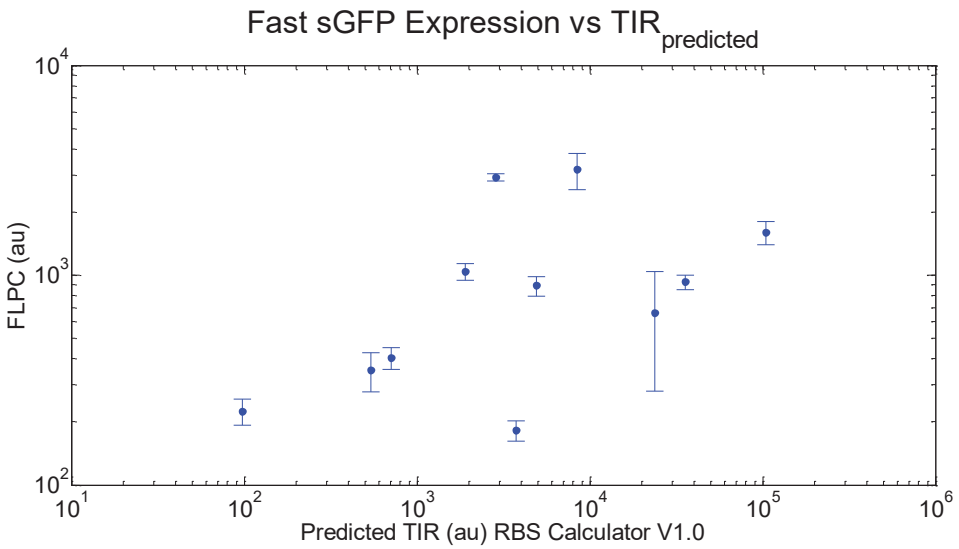


Figure 31: Maximum translation rate capacity analysis for Fast optimized sGFP was conducted by plotting the average FLPC for each colony vs the predicted TIR of the specific RBS sequence that was incorporated by that colony.

The data for Fast optimized sGPF show somewhat of a positive trend of increasing FLPC with predicted TIR. This is expected, as TIR is generally the rate limiting step in protein synthesis, but unfortunately insufficient data were collected to be able to discern the approximate value of TIR at which elongation becomes the rate limiting step (ie the maximum translation rate capacity). It was hypothesized that Fast optimized sGFP would result in a protein that increased in expression even at very high TIR, due to the fact that the codons in the coding sequence are predicted to be the fastest to be translated.

Similar graphical representation of results are shown for Rare-optimized sGFP in Figure 31.

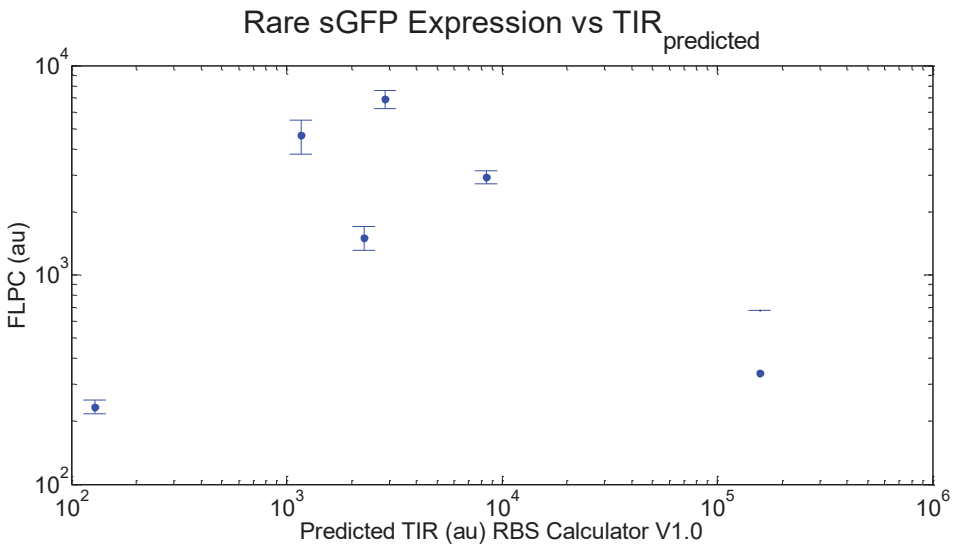


Figure 32: Maximum translation rate capacity analysis for Rare-optimized sGFP was conducted by plotting the average FLPC for each colony vs the predicted TIR of the specific RBS sequence that was incorporated by that colony.

Result for Rare-optimized sGFP suggest that intermediate values of TIR may result in higher expression than very high values of TIR, although due to the small amount of data that were collected, this cannot be stated with certainty. It was hypothesized that by using only rare codons, a very inefficient and slowly translated coding sequence would be developed, and that this would lead to a maximum translation rate capacity plateau, perhaps at a relatively low value of TIR.

The same graphical approach for SIT-optimized sGFP is presented in Figure 33.

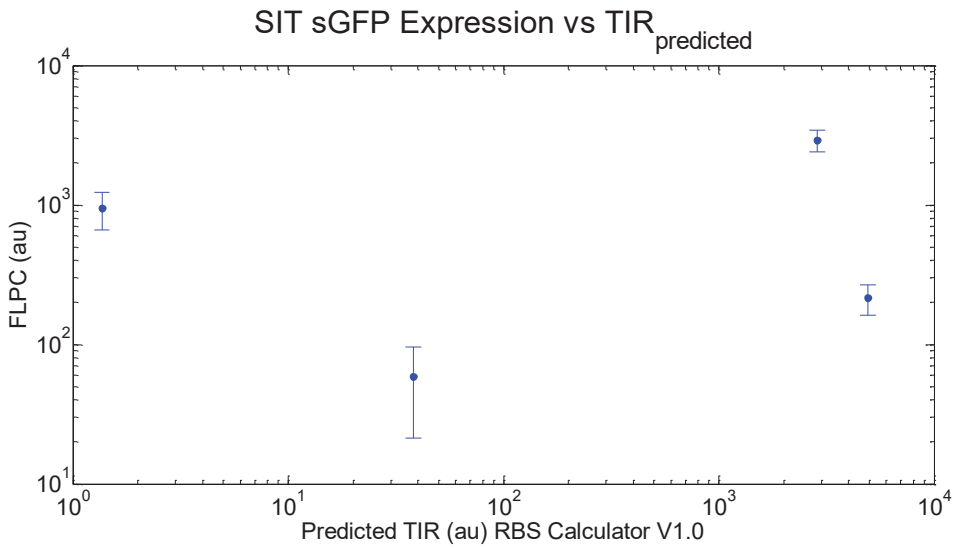


Figure 33 Maximum translation rate capacity analysis for SIT-optimized sGFPwas conducted by plotting the average FLPC for each colony vs the predicted TIR of the specific RBS sequence that was incorporated by that colony.

It was hypothesized that the slow insertion time sGFP would also display a maximum translation rate capacity plateau, and that this plateau would occur relatively early due to these codons being translated slowly. During cloning, it was very difficult to produce plates with sufficient number of colonies to proceed to TECAN data collection, and during sequencing it was discovered that the majority of the colonies were using the same RBS sequence, which greatly reduced the amount of results that are available for the SIT coding sequence.

Is it possible to prove that any optimization scheme is better than the others?

To answer this question, the data for expression must be analyzed across coding sequences while avoiding the confounding variable of ribosome binding site strength. This is done by comparing the expression of colonies with different variant sGFPs, but that had incorporated the same RBS. An analysis of variance is used to determine the presence of any statistically significant differences between the expression means. There was only one instance where it was possible to compare between all four sGFPs, which was for an RBS with predicted TIR = 2467 au. These results are shown in Figure 34.

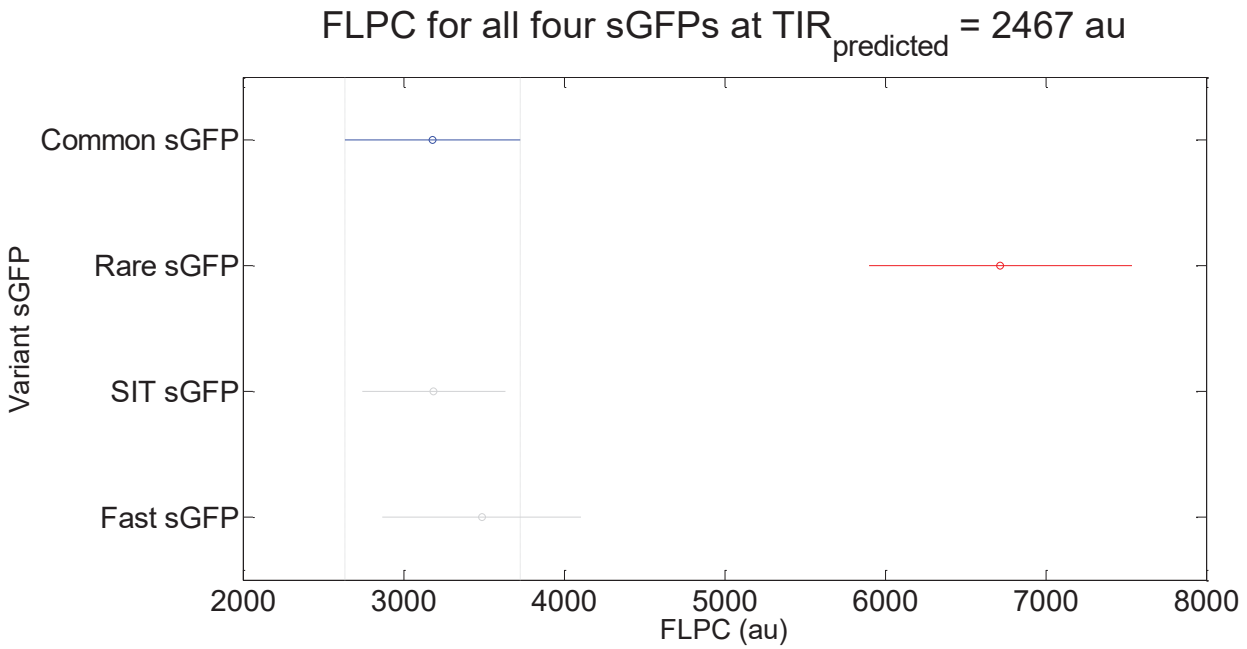


Figure 34: Analysis of variance of expression of all four variant sGFPs at TIR = 2467 au. Expression is shown on the x axis, with the mean of each group shown by a circle. Length of the whisker is set at the 90% confidence level for the mean.

The analysis shows that the only difference which was statistically significant was between rare optimized sGFP and the other three sGFPs. Statistical significance was reported at the p=0.9

certainty level, and the critical value was determine using the Tukey-Kramer method. Unexpectedly, rare optimized sGFP had the highest expression, although this is based on too few colonies to definitively say that this coding sequence is superior. The same analysis was also conducted for the ribosome binding site with predicted TIR = 707 au (available for three sGFPs) and is shown in Figure 35.

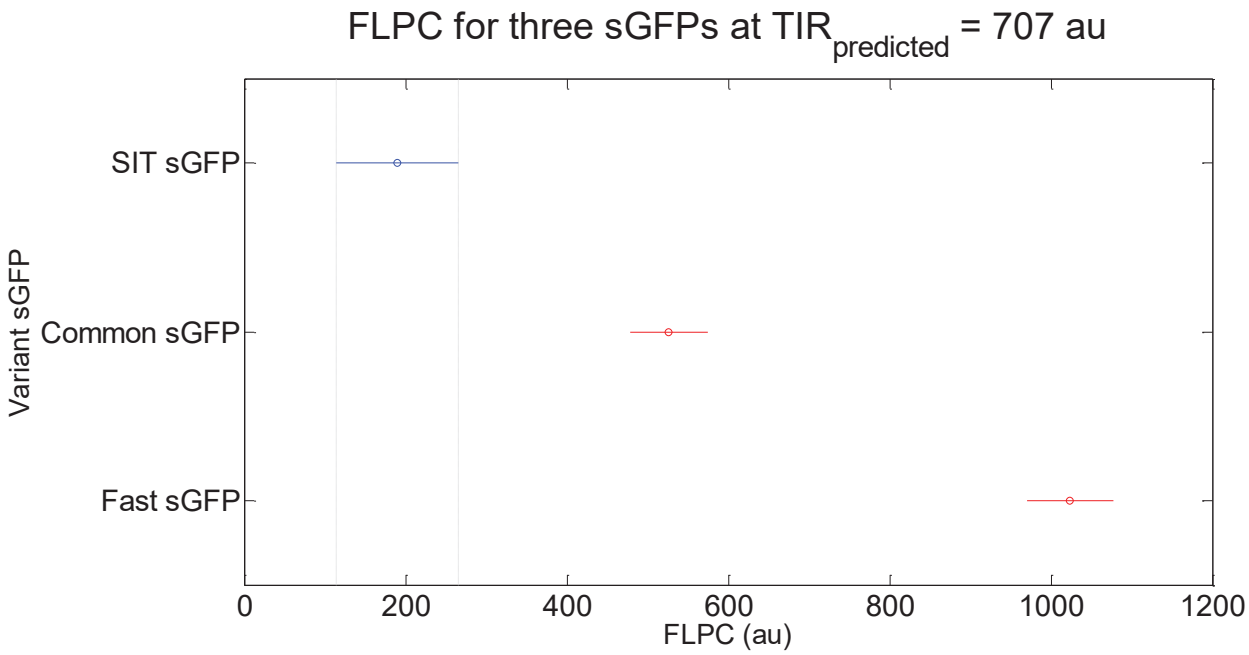


Figure 35: Analysis of variance of expression of three variant sGFPs at TIR = 707 au. Differences between all means were statistically significant.

From Figure 35 it is suggested that the fast sGFP is outperforming the common optimized sGFP and the slow insertion time optimized sGFP, but again there is too little data to definitively show that this will always be the case.

Can the RBS calculator be used to determine which constructs experienced a maximum translation rate capacity plateau?

Using the results from this experiment, the constants used by the RBS calculator model were back calculated. This allowed TIR for each RBS in the library to be calculated again, and the results showed an expected pattern, that expression tends to rise as TIR is increased but that this

increase declines as TIR is raised; that is to say that there are diminishing returns to increasing TIR. This suggests that these expression systems may have experienced a maximum translation rate capacity. An understanding of the RBS calculator is central to these ideas.

The RBS calculator uses statistical thermodynamic calculations to relate the rate of ribosome – mRNA association, which is called translation initiation rate (TIR) to the overall change in Gibbs free energy for the association reaction (ΔG_{total}). The value of ΔG_{total} is calculated based on several sub ΔG terms, which are presented in Equation 4.

$$\Delta G_{total} = \Delta G_{final} - \Delta G_{initial}$$

Equation 4: Several sub terms are used calculated to determine the overall ΔG.

$$\Delta G_{total} = \left(\Delta G_{mRNA-rRNA} + \Delta G_{start} + \Delta G_{spacing} - \Delta G_{standby}\right) - \Delta G_{mRNA}$$

This is then related to the overall Translation Initiation rate using Equation 2.

$$r\,(au) = Ke^{(-\beta \cdot \Delta G_{total})}$$

Equation 5: Translation initiation rate is a function of total free energy change

Where:

r	=	Translation initiation rate (au)
K	=	Proportionality constant
β	=	Boltzmann factor
ΔG_{total}	=	Total Gibbs free energy change (kcal/mol)

Thus, a more negative free energy change results in higher rate of mRNA—ribosome association. This relationship relies on two parameters, β and K, which are part of the exponential relationship between ΔG_{total} and TIR, as specified by statistical thermodynamics. Although precise calculations can be made by estimating these values, in a very thorough approach they can be

calculated from the expression data and then be used to re-calculate the predicted TIR for each ribosome binding site sequence in the library.

This was done by plotting the FLPC of all colonies that were measured vs the calculated change in free energy for the association of the ribosome and mRNA during translation initiation (ΔG_{total}), which is calculated by the RBS calculator. This relationship is shown in Figure 36.

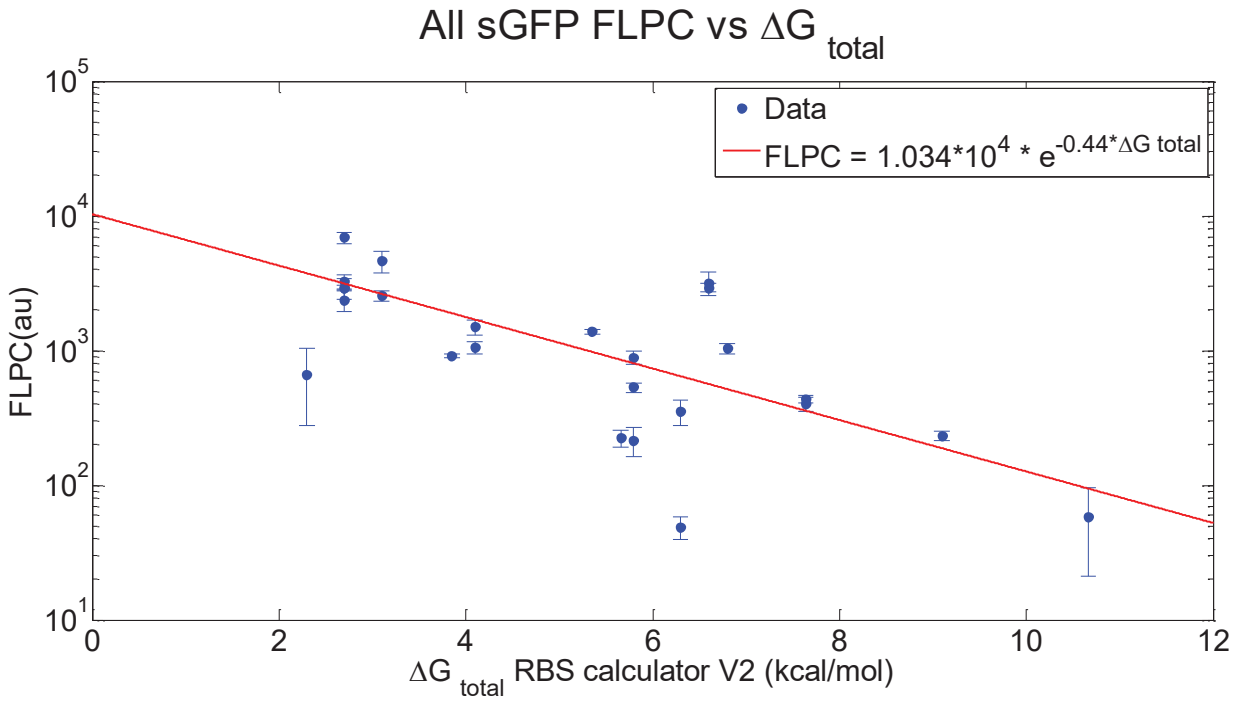


Figure 36: Relationship between FLPC and ΔG_{total} shows that expression decreases as the thermodynamics of ribosome—mRNA association become more unfavorable.

From Figure 36 an exponential model is fit, which is then used to calculate the parameters β and K for this system. This analysis is conducted using the results from all coding sequences together, with removal of outliers. Note that the same relationship (FLPC vs ΔGtotal) was plotted for each coding sequence individually, using the calculations of ΔGtotal from both the RBS calculator version 1.0 and 2.0 (and no removal of outliers), and these plots are presented in Appendix D: Supplemental Information.

$$r\ (au) = 1.034 * 10^4 * e^{(-0.44 \cdot \Delta G_{tot})}$$

Equation 6: Exponential fit allows parameters β and K to be determined

Where:

ΔG_{total}	=	Total Gibbs free energy change
β	=	Apparent Boltzman constant for the system
K	=	Proportionality constant

Thus, it is found that β = 0.44 and K = 10,034. These values fall roughly into the expected ranges, as β values are typically in the range of 0.4—0.5, and K is known to range from roughly 300—2500 [12]. Using the calculated values, TIR is recalculated for each ribosome binding site sequence in the library using Equation 6. Finally, the FLPC for each colony in the data set is plotted vs the recalculated TIR. This is shown in Figure 37.

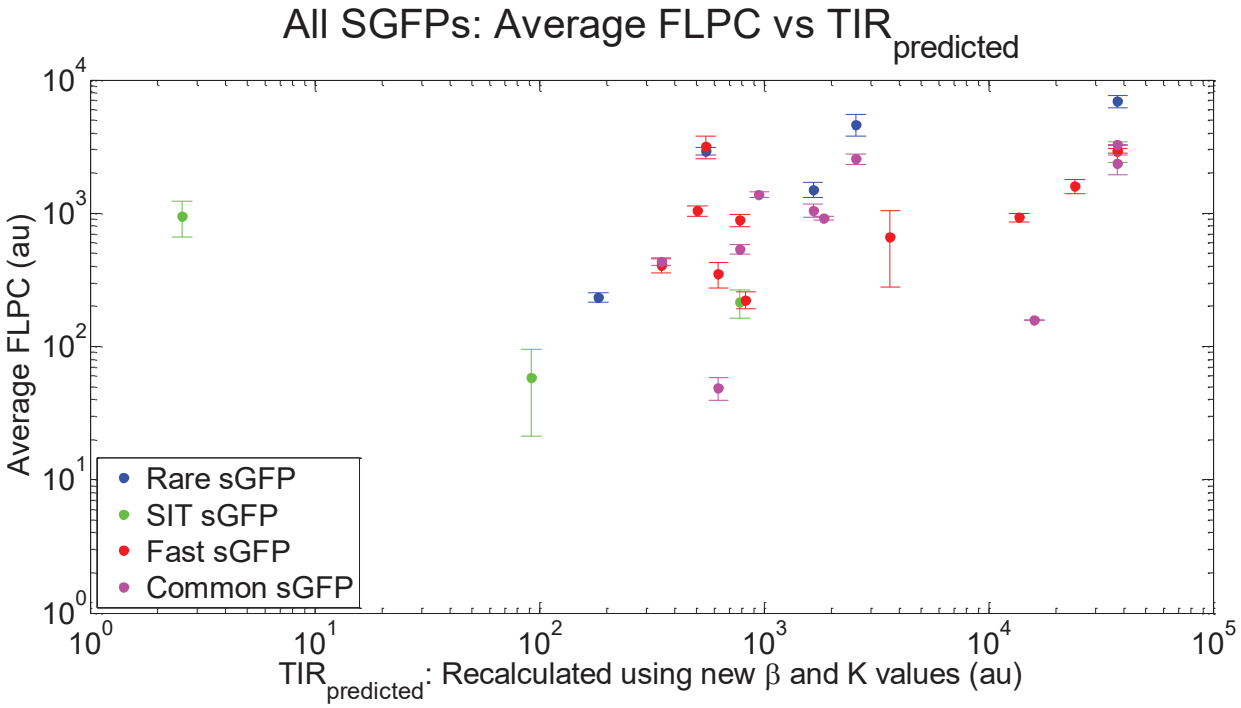


Figure 37: FLPC for all colonies of all coding sequences plotted vs TIR, after TIR has been recalculated based on the modeled β and K values

The data suggest that expression tends to increase as TIR increases, however, insufficient results were collected to show a definite trend. It remains possible that some optimized coding

sequences are stronger and do not demonstrate a maximum translation rate capacity plateau, where weaker coding sequences do plateau, or perhaps that weaker coding sequences demonstrate a plateau at a lower TIR whereas stronger ones do not plateau until higher values of TIR. A coding sequence that benefited from very fast translation elongation would be identified by expression that increased even as TIR was increased to very high levels.

To compare between variant coding sequences, it was desired to see how the expression of the different sGFPs varied based on predicted TIR. Thus, a "relative error" for each colony with a known RBS was plotted versus the predicted TIR of that RBS sequence (Figure 38).

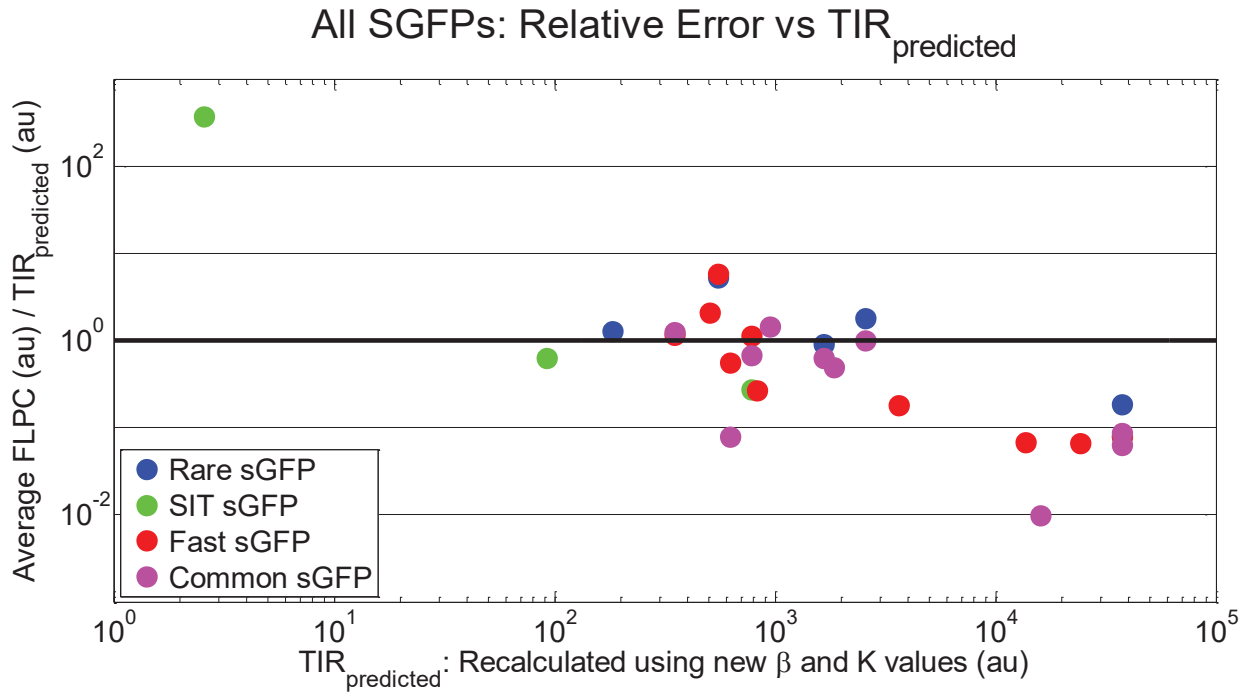


Figure 38: Relative error analysis of all four coding sequences is conducted by plotting (FLPC/Predicted TIR) vs Predicted TIR. The thick black line shows Relative Error =1, which is the line of perfect correspondence between the model and expression.

The value of relative error can be interpreted using Table 12.

Table 12: Relative Error $= (FLPC/TIR_{pred})/TIR_{pred}$

Relative error	Interpretation

$R < 1$	Expression is over-predicted for the system
$R > 1$	Expression is under-predicted for the system

In the case of a protein that was being expressed with translation initiation as the rate limiting step, the value of FLPC/TIR vs predicted TIR would be relatively constant. This is because any increases in TIR would be reflected by proportional increases in expression. In the case of translation elongation becoming rate limiting, after a certain TIR was reached there would be declining rewards to increasing it further. Thus, the value of FLPC/TIR would decrease at high TIR. This is found experimentally by other researchers (see Figure 5) as well as predicted by modeling the translation process [21]. This decrease in relative error is reflected in Figure 38, but there is still insufficient data to draw any hard conclusion about the value of TIR at which this affect begins, and at what level of expression the plateau occurs.

Why might a coding sequence predicted to be less efficient have higher expression?

Since each coding sequence had a different nucleotide profile, the mRNA transcripts were different and thus might have had different secondary structures. This could have impacted results because it is known that if mRNA has regions longer than roughly 10 base pairs where there is no secondary structure it can be more easily degraded by RNAase enzymes. It is plausible that even a very efficient coding sequence which had more vulnerability to enzymatic degradation could lead to less expression than a more inefficient coding sequence that was less prone to degradation. To determine if this could have been the case in this project, the Vienna RNA folding program (available at http://rna.tbi.univie.ac.at/), [32,33] was ran on the transcripts to determine if there were any obvious differences in their predicted secondary structures, specifically with regards to long stretches with no Watson-Crick base pairing. It appears that there is no section in any predicted

minimum free energy structure that indicates it would be especially prone to RNAase activity. These figures are provided in RNA Folding Figures.

Could the addition of a translated leader sequence have influenced the results?

In order to design a robust Ribosome Binding Site (RBS) library, it was necessary to homogenize the initial region of the coding sequence that followed the dRBS sequence. This is because RBS strength has been shown to be dependent on the DNA sequence up to roughly 35 base pairs downstream of the RBS site [14]. Although each coding sequence had the same amino acid profile each had a different DNA sequence due to codon optimization. This necessitated addition of a homogeneous leader sequence. However, one potential downside of using a translated leader sequence that is that the protein of interest is tagged with a short addition on its N-terminus. This could potentially impact the functionality of the protein, however, this was not predicted to be the case. This is demonstrated by modeling the folded structure of the protein, both with and without the leader sequence.

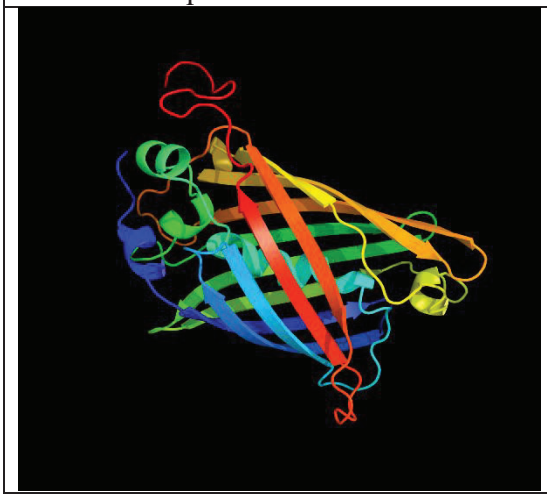
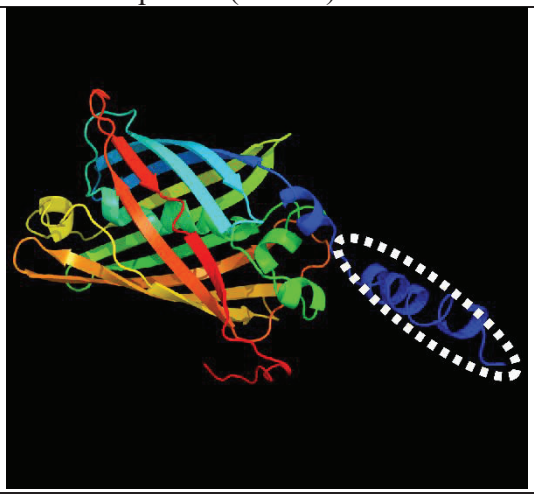
Comparison of predicted protein structure	
No leader sequence	Leader sequence (circled)
	

Figure 39: Predicted folded structure of sGFP shows that addition of leader sequence does not significantly alter the functional core of the protein. The leader sequence is circled. Structure predicted by the Protein Homology/analogY Recognition Engine (PHYRE^2), [40].

The main structure of the protein is a β barrel comprised of several anti-parallel β sheets, stabilized by hydrogen bonding. The structure is unchanged by the addition of the short leader sequence (21 amino acids) to the N-terminus, which is predicted to form an alpha helix (show in in blue).

In addition to the prediction of protein folded structure, the short translated leader sequence was searched for in the National Center for Biotechnology Information (NCBI) database of known protein domains (http://blast.ncbi.nlm.nih.gov/). This search looks for similarity between the amino acid profile of the leader and any proteins that have been characterized with taxpayer money. The highest result was 47% similarity, and this is likely not significant because in large databases short sequences can be found solely due to random effects up to this level of similarity (Expectation = 66). Expectation values close to zero indicate alignment that is not due to chance, and so this gives increased confidence that the addition of the leader did not impact protein expression [41].

The fact that the protein had retained functionality even after complete codon optimization of the CDS was confirmed by the display of fluorescence from all optimized genes Bacteria expressing common optimized sGFP are shown in Figure 40.

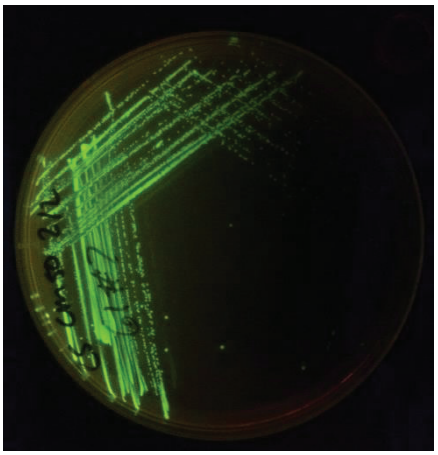


Figure 40: Optimized sGFPs displayed visual fluorescence

Thus, it is shown that a gene which whose codon profile has been artificially changed still can result in expression of a functional recombinant protein. A logical question is:

What are the driving forces of natural codon preference in organisms, and what does this indicate about the relationship between codon bias and translation elongation rates?

Organisms have evolved over long periods of time to a state of high efficiency. Furthermore, there is genome wide preference for certain degenerate codons [42], and such a preference has been observed in all organisms [16]. Preferences are not conserved from organism to organism, however, leading to debate regarding the origin and function of this phenomena [43].

Constitutively, the most statistically favored codons in the genome also correlate to the nucleotide composition of the organism's overall genome, meaning that organisms with high GC content genomes tend to prefer GC rich codons [43,44]. High AT genomes are similarly comprised of mostly AT rich codons [43]. It is also hypothesized that environmental effects may have a large impact on codon bias. For instance, it has been shown that the ability of organisms to grow at very high temperature affects the bias for specific degenerate codons [44,45]. However, the principal selective cause for codon bias is that certain codons are translated more accurately as well as more efficiently [43], yet the specific mechanism to explain this is not yet fully understood [21,43]. Selection

bias is also not constant within a genome. Specifically, codon usage bias differs between the beginning region of genes and the bulk of the mRNA transcript [46].

It has been discovered that codons corresponding to rare tRNAs are often present in the beginning of mRNA transcripts [17,23,47]. It was hypothesized that these codons were present to form an elongation "ramp," which would allow translation to start slowly and progress to higher rates, possibly preventing ribosomal "traffic jams" and facilitating an even and efficient spacing of ribosomes on the mRNA [23], however, this hypothesis was contradicted by recent results showing that rare codons are present to reduce secondary structure in the mRNA [17,47]. The ribosome can only initiate translation with linearized mRNA, and thus the presence of secondary structure in the mRNA at or near the "standby site" where the Ribosome 30s subunit first associates with the transcript reduces TIR [14,34,48]. Reductions in TIR then result in an overall lower protein expression, as predicted by the model in Equation 1.

This makes sense in the context of the equilibrium thermodynamic model of translation initiation, as the Gibbs free energy required to unfold mRNA secondary structures contributes to a less negative free energy change for the overall association reaction [14] and thus decreases TIR, as shown in Equation 5. It has been discovered that in bacteria selection favors codons that reduce mRNA folding in the region where translation begins, regardless of whether these codons are frequent or rare [17,47], which indicates that the presence of rare codons in these regions is to raise TIR rather than to create a "ramp" for translation elongation rate.

Even though there are 64 total codons, there are not an equal number of tRNAs with complimentary anticodons. The specific number ranges from 39 specific tRNAs in *E. coli*, to 45 in *Homo sapiens*, to only 28 in the obligate intracellular parasite *Mycoplasma* [16]. tRNAs which pair to codons that do not exactly match their anticodon are called tRNA isoacceptors, and they usually differ from the exact compliment to the paired codon in the third base pair [7]. This impacts

codon usage preference because it has been determined that in highly expressed genes, the most abundant tRNA isoacceptors in the cell coincide with the most frequently used codon within a degenerate set [49]. One method by which a gene's specificity for a particular organism can be quantified is its tRNA adaptation index (tAI), which is defined as the mean adaptation of the tRNAs necessary to translate that gene to the tRNA pool in the cell [21].

However, since charged tRNA must be available at the acceptor site of the ribosome for translation to occur, depletion of the pool of a certain charged tRNA near the site of translation will slow the elongation rate. When cognate tRNAs are limited, competition between cognate, near cognate (isoacceptor), and non-cognate tRNAs occurs with greater frequency [50], and becomes rate limiting in translation elongation [30]. Faster recognition of cognate tRNA accelerates translation elongation, whereas incorporation of near cognate tRNAs causes delay, as these can be either accepted (no change in elongation rate relative to cognate tRNA) or rejected during a proofreading step (causing delay) [51,52]. Thus, it would be intuitive that codon usage preferences would correlate with the specific population bias of tRNA isoacceptors for a particular organism, and this is found to be true [53].

The hypothesis that codon bias is due to a need for efficiency in protein synthesis is supported by results which show that codon bias is strongest in genes that are highly expressed [54] such as ribosomal genes, translation elongation factors, and membrane proteins [55]. More highly expressed genes also demonstrate a higher maximum translation rate capacity than lowly expressed genes (see Figure 6), and this indicates that the extent of codon bias can influence maximum translation rate capacity [21]. Furthermore, the strength of codon usage bias has been shown to be highly correlated with bacterial growth rates, indicating that natural selection favors translational efficiency [54].

Since the total expression capacity of a cell is limited (due to finite cellular resources), the goal of engineering expression of a desired product is not necessarily to globally upregulate protein synthesis, but to increase the translation of individual mRNAs differentially and thus ensure that the resources used in protein synthesis are allocated to the desired mRNA transcripts [12]. Thus, codon optimization is an attractive option for raising expression of a desired product, as it can increase efficiency, freeing cellular resources (slight global increase), as well as differentially increase translation of only desired products.

Codon optimization is a particularly attractive option for raising protein expression because it complements the ability of the RBS calculator to design functional DNA. This is because the RBS calculator can be used to accurately raise TIR, but if plateaus in expression occur at high TIR, there is a need for an additional tool.

Why is there a need for a biophysical model of translation elongation?

First, there are several ways that DNA is designed *without* a robust biophysical model. For instance, codon optimization by including only common codons does not take into account the physical interactions of translation. Similarly, transcriptional promoters are designed without a true biophysical model, but can still be used to raise expression. However, these methods will not be sufficient to fully conduct codon optimization. In summary, this is because phenomenological approaches to design require characterization of large amounts of sequences to get enough data to develop relationships between DNA sequence and expression level, and the mathematics that define the variable space for codon optimization preclude a complete set of experiments from being possible. This is because maximum translation rate capacity is due to more than the combination of codons of an mRNA transcript (order independent), and in fact varies by the permutations of codons (order dependent) [21], meaning that even if only common codons are to be

used, optimization of the first half a gene may not lead to the same expression as optimization of the second half of the same gene.

In fact, the effect of codon order on translation rates of mRNA transcripts with identical combinations of codons has been shown to affect protein expression by more than 20% [21]. This demonstrates that there is the possibility of changing only some codons to a desired codon within the degenerate set, based on the predicted effects of codon order on translation rate capacity. The recognition that codon order impacts expression enables a broader discussion of the space of variables for codon optimization.

One of the fundamental goals of optimization is to determine how large the space is that would be investigated if every feasible combination of variables were explored, and then determine if this space can be narrowed by making realistic assumptions about the system. In this case, the optimization goal is to raise protein expression, and the variable that is considered is the sequence of DNA expressed by the organism. With no constraints, the total number of possible DNA strings of length L can be found using Equation 7.

$$N_{string} = 4^{L}$$

Equation 7: Number possible coding sequences for string of length L

$$N_{string} = 4^{(3A)}$$

Where:

L = Number of base pairs in the string
A = Number of amino acids specified by the string

This result is a direct consequence of there being only four possible bases in DNA, and that the order of those bases in a string of length L is important, that is, the total number of possible strings is the number of permutations of 4 nucleotides in a string of length L.

In the context of codon optimization, however, it is necessary to preserve the primary structure of the protein, so the permutations are limited to those that result in the same protein being expressed. Thus, the number of possible coding sequences for a protein of length *L* amino acids is calculated by Equation 8.

$$N_{gene} = \prod_{i=1}^{L} D_i$$

Equation 8: Possible degenerate coding sequences for gene of length *L* amino acids

Where:

D_i	=	Number of degenerate codons for amino acid *i*
N_{gene}	=	Number of permutations of the gene resulting in expression of the same amino acid

Using these equations, it is possible to calculate the optimization space, and also determine the point at which the exploration of all possible combinations of sequences is no longer feasible. This is important because sometimes it may not be desired to optimize an entire coding sequence, as was done in this project, and instead only half of a coding sequence, or one fourth, or even just a few codons. This has been attempted in a previous project in which the codon composition of a short leader sequence was altered [56] and it was identified that topic for future work was the determination of the feasibility of optimizing longer sequences.

It can be seen from Equation 8 that the number of permutations of a gene grows exponentially with the gene's length, but this relationship can be quantified more precisely by simulation. In this approach, random genes of length A amino acids were generated according to the codon usage of E. coli (codon fraction per 100 codons used to weight the stochastic generation of random genes) and then the number of permutations of each gene were calculated. Averaging this number over the number of generated genes gives a reasonable expectation for the optimization space that is specific to E. coli. The results of this simulation are shown in Figure 41

and the code for the simulator is available in: Script for Determining Combinatorial Space of Codon Optimization in *E. coli*.

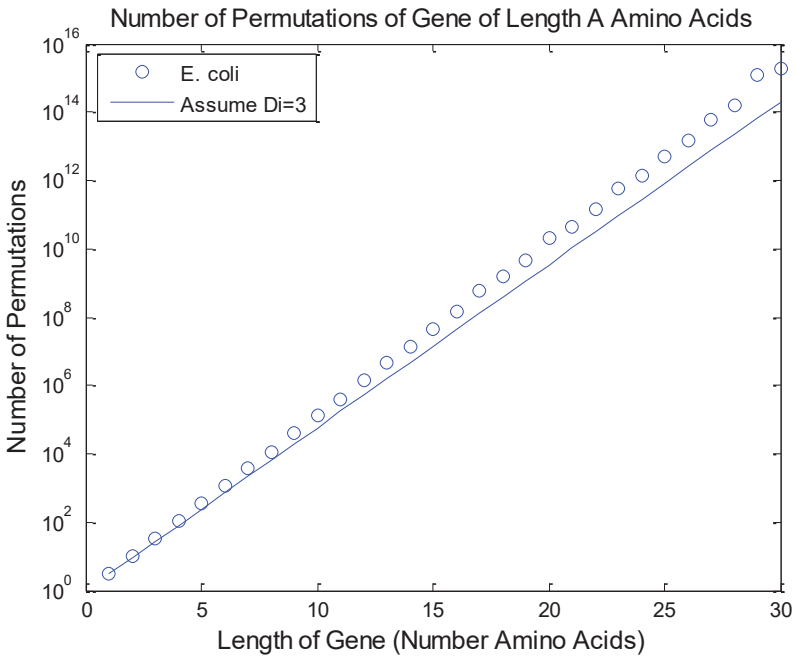


Figure 41: Number of permutations for codon optimized E. coli genes grows exponentially.

It can be seen that complete characterization of all permutations of a codon optimized gene is impossible due to combinatorial explosion, and that a guided approach is needed in design, which comes from mechanistic modeling of translation elongation.

This is somewhat similar to the problem faced by protein folding calculators, and also the RBS calculator, as the number of permutations of a ribosome binding site of length L quickly exceeds the realm of feasibly calculating the thermodynamic properties of each one. Because of this, a guided approach is taken rather than simply "brute force" calculation of all possible outcomes. Since an RBS is not translated, the number of permutations is calculated by Equation 7.

What are some alternatives for raising translation elongation, and are there any foreseeable drawbacks to codon optimization?

It has been shown that codon optimization is a feasible approach to increasing expression limits imposed by maximum translation rate capacity [24–26], but other methods may be viable as well. It has been theorized that expression of recombinant proteins in hosts outside the original organism could be lifted without codon optimization by upregulating the expression of tRNAs that are rare in the expression organism but common in the original organism [57]. The advantage of this approach is that it could potentially adapt an industrial workhorse, such as *E. coli*, to expression of numerous exogenous proteins originally found in a different organism, such as *Homo sapiens* without having to optimize and re-synthesize individual genes. However, this hypothesis was tested by expressing 94 human proteins in *E. coli*, and the genes that were codon optimized for specificity to *E. coli* showed higher expression than the tRNA adapted trials in all cases [57].

Even though there are numerous examples of codon optimization being used to increase protein expression, there are potential drawbacks to codon optimization of a gene for translational efficiency which must also be considered. This is because translation does not occur independently of other processes implicated in protein production, in particular protein folding, which for long polypeptide chains occurs simultaneously to elongation [58–60]. Because of this, modifying the rate of translation elongation may have negative repercussions for protein folding [58]. In an experiment where 16 rare codons were replaced with common degenerate codons in a gene coding for chloramphenicol acetyltransferase (confers Chloramphenicol resistance), the resultant protein was shown to have 20% lower specific activity, even though it was composed of the same amino acid sequence [61]. This was attributed to mis-folding of the protein, and suggests that the kinetics of translation elongation can impact the ability of proteins to correctly fold in vivo [61,62].

Even the change of a single degenerate codon in the coding sequence (from a common codon to a rare codon) has been shown to negatively affect the protein's enzymatic activity, which was attributed to mis-folding due to ribosomal pauses allowing the protein to pursue alternate

folding pathways and become “kinetically trapped” at local minima on the free energy folding surface instead of reaching the native conformation [60]. This suggests that slowly translated codons may not always result in better folding of a protein, and in fact, that fast translating codons can improve cotranslational folding by preventing the protein from winding up in misfolded intermediate states [58].

Furthermore, the proportion of slow translating codons is not constant across all coding sequences, which indicates that the purpose of these codons is not simply to slow down translation uniformly across all regions of the genome. In fact, research has shown that slow-translating codons are more prevalent at the boundaries between protein domains, which may indicate that there is a relationship between translation rate and co-translational protein folding [61].

Thus, codon level optimization must be conducted with the understanding that there are multiple events necessary for proper protein production, and that increasing translation rate may have unintended consequences.

What are the opportunities for future research and engineering?

There is a lot of potential for future research in this topic due to the need in industry to ensure that recombinant protein production can always be maximized. From an experimental standpoint, an optimization protocol similar to the ones used in this project could be used, and simply collecting more data would be very useful. Along with this, collecting data across a variety of growth media, different temperatures, levels of agitation, and using a method like flow cytometry for more precise results would allow for more specific conclusions to be made regarding the effectiveness of different methods of codon optimization.

Second, future research should be designed to approach the problem of raising protein expression with the goal of developing a quantitative, mechanistic model. A successful model of translation elongation would be a development of similar magnitude to that of the RBS calculator.

By including all of the known biophysical interactions that occur in translation, a coding sequence with tunable translation elongation rates could be designed. Such a model could:

A. Consider the tRNA adaptation index of the coding sequence to the specific organism where it is expressed.
B. Consider the amino acid insertion time with the goal of minimizing total elongation time
C. Reduce secondary structure in the mRNA, with decreasing weight to this objective as distance from start codon increases, as this is most important during translation initiation
D. Consider the potential for "ribosome drafting" which occurs in situations where slowly folding mRNA hairpins allow additional ribosomes to participate in translation of a particular transcript without incurring the full free energy penalty of unfolding the mRNA
E. Calculate the effect of decreasing charged tRNA concentration as the transcript is read, which may result in penalties for genes that use only one codon to specify an amino acid every time it is called

Another advantage of a model of translation elongation is the acceleration of research and development. Current therapeutics such as recombinant artemisinin have taken a very long time to develop and bring to industrial production, and the use of codon optimization to boost product production (as well as expression of intermediate enzymes in metabolic pathways) could greatly accelerate the process. Codon optimization may also prove to be an effective method for differentially regulating the expression of individual genes in synthetic operons, as these systems

are currently engineered using biophysical approaches that accurately modulate translation initiation at intermediate sites [63] but may benefit from the additional control over elongation rates.

In addition to production purposes, development of next generation "bio-computers" based on sophisticated algorithms encoded by genetic circuits would benefit from being able to ensure that translation elongation was not rate limiting.

Conclusions

The data and analysis from this study can be used to draw several important conclusions. The challenge in doing so is recognizing situations when insufficient data were collected to avoid coming to conclusions that are incorrect, while simultaneously making the best possible use of the work that was successfully completed. The main objective was to determine whether the novel criteria for codon optimization could be used to lift maximum translation rate capacity plateaus that were noticed in reporter gene expression at very high translation initiation rates (TIR) by previous researchers. Unfortunately, not enough data were collected to provide a definitive answer to this question. In order to do so, more constructs bearing very high TIR ribosome binding sites would need to be characterized. This is because in order to demonstrate the presence of a plateau, and quantify the TIR at which the plateau occurred, a large amount of colonies would need to be successfully sequenced. During sequencing, a large amount of "duplicate" RBS sequences in the library were found, that is, many constructs that were sequenced had the same RBS sequence. This prevented data from being collected across a sufficiently wide dynamic range of TIR.

However, the analysis of the data that were collected allowed the RBS calculator model to be parameterized for the specific expression system used in this project. From the plot of expression verses total ΔG, the values of parameters β and K were found for the system, and they were shown to fall into the approximate range of expected values. These new parameter values were then used to recalculate TIR for each of the sequences in the library based on the relationship between TIR and the overall ΔG through statistical thermodynamics. Then, the new data of expression vs TIR were plotted, roughly showing the expected trend of increasing expression at increasing TIR. Next, the relative error was plotted, which shows the "reward" of increasing TIR on expression. It was found that the returns diminished at high TIR, which suggests an overall over

prediction of expression in the system and suggests that expression may still plateau, even for the favorably optimized genes.

The variant genes were also compared head to head by conducting an analysis of variance of expression when the same RBS sequence was used. The results were inconclusive: when all four were compared head to head, the Rare optimized sGFP unexpectedly had the highest average expression. When three variants were compared at the same TIR, the Fast optimized sGFP had the highest expression, which was consistent with the hypothesis.

Finally, the combinatorial space for codon optimization of coding sequences in E. coli was analyzed by determining the number of possible permutations of simulated genes generated using the statistical preference of all codons in E. coli. This gives future researchers a starting point to determine what size of sequences could be optimized and then exhaustively characterized, and also shows that eventually a biophysical model of translation will be necessary to fully understand the effect of codon optimization on expression, and whether or not it can be used to raise maximum translation rate capacity for commercially significant products.

Script for Optimizing Genes

In the design phase of this project, genes were taken apart into codons, and optimization was conducted by replacing all codons that specified a single amino acid with the degenerate codon specified for that amino acid in each optimization scheme. To do this efficiently, a script was composed in MATLAB were the gene was input in a string and then optimized according to all of the schemes. The preference for which codon should be used for each amino acid in each scheme was contained in a cell array called "codontable," which can also be found in (Table 4).

```
%GFP_optimize_complete
%Clay Swackhamer
home
clear
clc

gene='ATGCGTAAAGGCGAAGAACTGTTTACCGGTGTGGTTCCGATTCTGGTGGAACTGGATGGTGATGTTAATGGTCATAAATTCAGCGTTCGTG
GTGAAGGCGAAGGTGATGCCACGAATGGTAAACTGACCCTGAAATTTATCTGCACCACAGGTAAACTGCCGGTTCCGTGGCCGACCCTGGTTACCAC
CCTGACCTATGGTGTTCAGTGTTTCGCACGTTATCCGGATCATATGAAACAGCACGATTTCTTTAAAAGCGCCATGCCGGAAGGTTATGTTCAGGAA
CGTACCATTAGCTTTAAAGATGACGGCACCTATAAAACCCGTGCCGAAGTTAAATTCGAAGGCGATACCCTGGTGAATCGTATCGAACTGAAAGGCA
TCGATTTTAAAGAGGATGGTAATATCCTGGGCCATAAACTGGAATATAATTTTAACAGCCATAACGTGTATATCACCGCAGATAAACAGAAAAACGG
CATTAAAGCGAACTTTAAAATCCGCCATAATGTGGAAGATGGTAGCGTTCAGCTGGCAGATCATTATCAGCAGAATACGCCGATCGGTGATGGTCCG
GTTCTGCTGCCGGATAATCATTATCTGAGCACCCAGAGCGTTCTGAGTAAAGATCCGAATGAAAAACGTGATCACATGGTGCTGTTAGAGTTCGTTA
CCGCAGCAGGTATTACACATGGTATGGATGAACTGTATAAA'
codons=cellstr(reshape(gene,3,[])'); %divide inputgene into codons. Make sure it is divisible by
three. Omit any stop codons. Since they don't code for an AA they interrupt the nt2aa function

aasequence=nt2aa(codons,'alternativestartcodons','false'); %convert genecodons to AA's
length(aasequence); %gives the number of AA's in the sequence

header = {'Amino Acid','Rare','Common','Fast','Slow','SIT'} %Specify which degenerate codon will
be used for each amino acid in each optimization scheme
codon_table =
{'M','ATG','ATG','ATG','ATG','ATG';'W','TGG','TGG','TGG','TGG','TGG';'F','TTC','TTT','TTC','TTT',
'TTC';'T','ACT','ACC','ACT','ACA','ACA';'I','ATA','ATT','ATC','ATA','ATA';'L','CTA','CTG','CTG','
TTG','CTT';'V','GTA','GTG','GTT','GTG','GTC';'S','TCA','AGC','TCT','TCG','TCC';'P','CCC','CCG','C
CG','CCC','CCC';'A','GCT','GCG','GCT','GCC','GCC';'Y','TAC','TAT','TAC','TAT','TAC';'H','CAC','CA
T','CAC','CAT','CAT';'Q','CAA','CAG','CAG','CAA','CAG';'N','AAT','AAC','AAC','AAT','AAC';'K','AAG
','AAA','AAA','AAG','AAG';'D','GAC','GAT','GAC','GAT','GAC';'E','GAG','GAA','GAA','GAG','GAA';'C'
,'TGT','TGC','TGC','TGT','TGC';'R','AGG','CGT','CGT','CGA','AGG';'G','GGA','GGC','GGT','GGG','GGA
'}
```

```
%convert AA's back to codons, using the desired scheme for optimization
```

RARE

```
for i = 1:length(aasequence);
    indexVector = strfind(codon_table(:,1),aasequence{i});
    index(i) = find(not(cellfun('isempty',indexVector)));
    raregeneseq(i) = codon_table(index(i),2);
end
raregeneseq;
rare = strjoin(horzcat(raregeneseq),'');
```

COMMON

```
for i = 1:length(aasequence);
    indexVector = strfind(codon_table(:,1),aasequence{i});
    index(i) = find(not(cellfun('isempty',indexVector)));
    commongeneseq(i) = codon_table(index(i),3);
end
commongeneseq;
common = strjoin(horzcat(commongeneseq),'');
```

FAST

```
for i = 1:length(aasequence);
    indexVector = strfind(codon_table(:,1),aasequence{i});
    index(i) = find(not(cellfun('isempty',indexVector)));
    fastgeneseq(i) = codon_table(index(i),4);
end
fastgeneseq;
fast = strjoin(horzcat(fastgeneseq),'');
```

SLOW

```
for i = 1:length(aasequence);
    indexVector = strfind(codon_table(:,1),aasequence{i});
    index(i) = find(not(cellfun('isempty',indexVector)));
    slowgeneseq(i) = codon_table(index(i),5);
end
slowgeneseq;
slow = strjoin(horzcat(slowgeneseq),'');
```

SLOW INSERTION TIME (IMAN DATA)

```
for i = 1:length(aasequence);
    indexVector = strfind(codon_table(:,1),aasequence{i});
    index(i) = find(not(cellfun('isempty',indexVector)));
```

```
    sitseq(i) = codon_table(index(i),6);
end
sitseq;
SIT = strjoin(horzcat(sitseq),'');
```

AGGREGATE RESULTS

```
optimized_genes = {'Rare',rare;'Common',common;'Fast',fast;'Slow',slow;'SIT',SIT};

lenrare = length(rare);%Check to make sure that they are all the same length
lencom = length(common);
lenfast = length(fast);
lenslow = length(slow);
lensit = length(SIT);
```

Published with MATLAB® R2014a

Script for Totaling Codon Insertion Time of each GFP

One of the variant GFPs was constructed entirely of the codon for each amino acid with the slowest insertion time of all degenerate codons. In order to see the difference in total insertion time between the variants, a script was composed to look up the insertion time for each codon and then sum them for each variant. The table of codon insertion times can be found in Table 2.

```
%insertion_time_calculator_complete
%Clay Swackhamer
clc

%define an insertion time for each codon
all_codons =
{'TTT';'TTC';'TTG';'TTA';'TCT';'TCC';'TCG';'TCA';'TGT';'TGC';'TGG';'TGA';'TAT';'TAC';'TAG';'TAA';
'CTT';'CTC';'CTG';'CTA';'CCT';'CCC';'CCG';'CCA';'CGT';'CGC';'CGG';'CGA';'CAT';'CAC';'CAG';'CAA';'
GTT';'GTC';'GTG';'GTA';'GCT';'GCC';'GCG';'GCA';'GGT';'GGC';'GGG';'GGA';'GAT';'GAC';'GAG';'GAA';'A
TT';'ATC';'ATG';'ATA';'ACT';'ACC';'ACG';'ACA';'AGT';'AGC';'AGG';'AGA';'AAT';'AAC';'AAG';'AAA'};
codontimes =
[136;195;50;157;55;246;96;106;75;109;168;12;53;77;19;11;260;204;35;186;143;197;134;237;28;35;397;
34;296;222;231;179;26;208;42;73;39;415;44;83;35;49;81;324;77;116;36;57;97;128;266;128;55;153;129;
178;85;127;461;190;109;161;102;76];

rare_codons=cellstr(reshape(rare,3,[])'); %divide each input gene into codons. Requires output of
'GFP_optimize_complete'
common_codons=cellstr(reshape(common,3,[])');
```

```
fast_codons=cellstr(reshape(fast,3,[])');
slow_codons=cellstr(reshape(slow,3,[])');
SIT_codons=cellstr(reshape(SIT,3,[])');

numrare=length(rare_codons); %make sure that each of the variants still has the same number of
nucleotides
numcommon=length(common_codons);
numfast=length(fast_codons);
numslow=length(slow_codons);
numSIT=length(SIT_codons);
```

RARE

```
for i = 1:numrare;
    indextimevector = strfind(all_codons(:,1),rare_codons{i}); %looks through codontimes cell
array to find the codon that is the same as the ith codon in the gene
    index(i) = find(not(cellfun('isempty',indextimevector))); %reports the number of this codon
in the table (if it is the third from the top then i=3)
    times(i) = codontimes(index(i),1); %looks up an insertion time to correspond to this in the
matrix 'codontimes', thus [codontimes] must be constructed with each codon in the exact same
position as it occupies in {codons}
end
times;
rare_insertion_time_ms=sum(times);
rare_insertion_time_s=rare_insertion_time_ms/1000
```

```
rare_insertion_time_s =
   35.8710
```

COMMON

```
for i = 1:numcommon;
    indextimevector = strfind(all_codons(:,1),common_codons{i});
    index(i) = find(not(cellfun('isempty',indextimevector)));
    times(i) = codontimes(index(i),1);
end
times;
common_insertion_time_ms=sum(times);
common_insertion_time_s=common_insertion_time_ms/1000
```

```
common_insertion_time_s =
   23.8750
```

FAST

```
for i = 1:numfast;
    indextimevector = strfind(all_codons(:,1),fast_codons{i});
    index(i) = find(not(cellfun('isempty',indextimevector)));
    times(i) = codontimes(index(i),1);
end
times;
```

```
fast_insertion_time_ms=sum(times);
fast_insertion_time_s=fast_insertion_time_ms/1000
```

```
fast_insertion_time_s =
   22.1210
```

SLOW

```
for i = 1:numslow;
    indextimevector = strfind(all_codons(:,1),slow_codons{i});
    index(i) = find(not(cellfun('isempty',indextimevector)));
    times(i) = codontimes(index(i),1);
end
times;
slow_insertion_time_ms=sum(times);
slow_insertion_time_s=slow_insertion_time_ms/1000
```

```
slow_insertion_time_s =
   28.4630
```

SLOW INSERTION TIME (SIT)

```
for i = 1:numSIT;
    indextimevector = strfind(all_codons(:,1),SIT_codons{i});
    index(i) = find(not(cellfun('isempty',indextimevector)));
    times(i) = codontimes(index(i),1);
end
times;
sit_insertion_time_ms=sum(times);
sit_insertion_time_s=sit_insertion_time_ms/1000
```

```
sit_insertion_time_s =
   48.7350
```

AGGREGATE RESULTS

```
insertion_times =
{'Rare',rare_insertion_time_s;'Common',common_insertion_time_s;'Fast',fast_insertion_time_s;'Slow
',slow_insertion_time_s;'SIT',sit_insertion_time_s}
```

```
insertion_times =
    'Rare'      [35.8710]
    'Common'    [23.8750]
    'Fast'      [22.1210]
    'Slow'      [28.4630]
    'SIT'       [48.7350]
```

Published with MATLAB® R2014a

Script for Determining Combinatorial Space of Codon Optimization in *E. coli*

This script allows the combinatorial space of codon optimization to be identified. Since both codon order and codon identity are important, the total number of permutations of a gene grows exponentially with the length of the gene. This script generates random genes of incrementally increasing length, with the constraint that all genes in a set must have the same amino acid profile. This profile is set by assigning a statistical weight to each possible codon based on its experimentally determined frequency in the entire complement of *E. coli* coding sequences [28]. Then, the total number of permutations of those genes are calculated, averaged, and the results are plotted. Results show combinatorial explosion even more quickly than if each amino acid simply had three possible degenerate codons. This is because of the statistical abundance of amino acids with either four, five, or six possible degenerate codons.

```
%e_coli_optimspace
%Clay Swackhamer
clear
tic
load codons
clc
%Objective: Find out how big is the optimization space for a gene of length
%A amino acids in e coli.

%N_gene = product(Di), where Di is the number of degenerate codons in the
%position of codon i
num_aas = 30; %number amino acids in gene
A = 1; %index of amino acids

for j=1:1:num_aas

    iter = 1000;
    for i=1:1:iter;

        probs = cell2mat(codons(2:65,3)); %probability of this codon appearing out of every 100
codons in e coli genome
        num_codons = length(cell2mat(codons(2:65,1))); %64 codons are possible

        R = randsample(num_codons,A,true,probs); %draw a random sample of A integers from 1-64,
with replacement, with the probs weighted based on the prob of that codon occuring
        Di = cell2mat(codons(2:65,4)); %number of degenerate codons for codon i
        perms = Di(R); %take the sample that was drawn and return the amount of possible
degenerate codons for each of them
        N_gene = prod(perms); %take the product of all those, based on the formula up top. This
is the number of permutations of this gene that still express the same AA
```

```
        individual_trials(i) = N_gene; %the optimization space on each random trial

    end
        average_space(j) = mean(individual_trials); %the average of all the numbers of spaces
calculated by the loop from i:1:iter
         A = A +1;
end
average_space

x_vals = 1:1:num_aas; %number of amino acids to be plotted on x axis
figure
semilogy(x_vals, average_space,':')
hold on
semilogy(x_vals, 3.^x_vals) %the amount of permutations if you assume that there are three
possible degenerate codons for each AA
title('Number of Permutations of Gene of Length A Amino Acids','Fontsize',12)
ylabel('Number of Permutations','Fontsize',12)
xlabel('Length of Gene (Number Amino Acids)','Fontsize',12)
legend('E. coli','Assume Di=3','Location','northwest')
%Codon Frequency Weighted for Statistical Presence in E. coli Genome
toc
%}
```

Published with MATLAB® R2014a

Script for Calculating percent similarity between sGFPs

This script is used to determine the positional similarity between sGFP coding sequences. It functions by breaking each optimized gene (direct output from the section: Script for Optimizing Genes) into individual codons. Then, one gene is compared to another using a logical matrix that records a 1 if the codon of gene A in position *i* is the same as the codon of gene B in position *i*. Next, the total percent similarity is calculated. The total number of comparisons of two genes in a pool of five is ten (five choose two).

```
% Compare gene similarity by position

clear
load optimized_genes
%Divide all genes into row vectors with one codon in each cell

rare = cell2mat(optimized_genes(1,2));
rare_codons=cellstr(reshape(rare,3,[])');
ind = length(rare_codons);
for i=1:1:ind
```

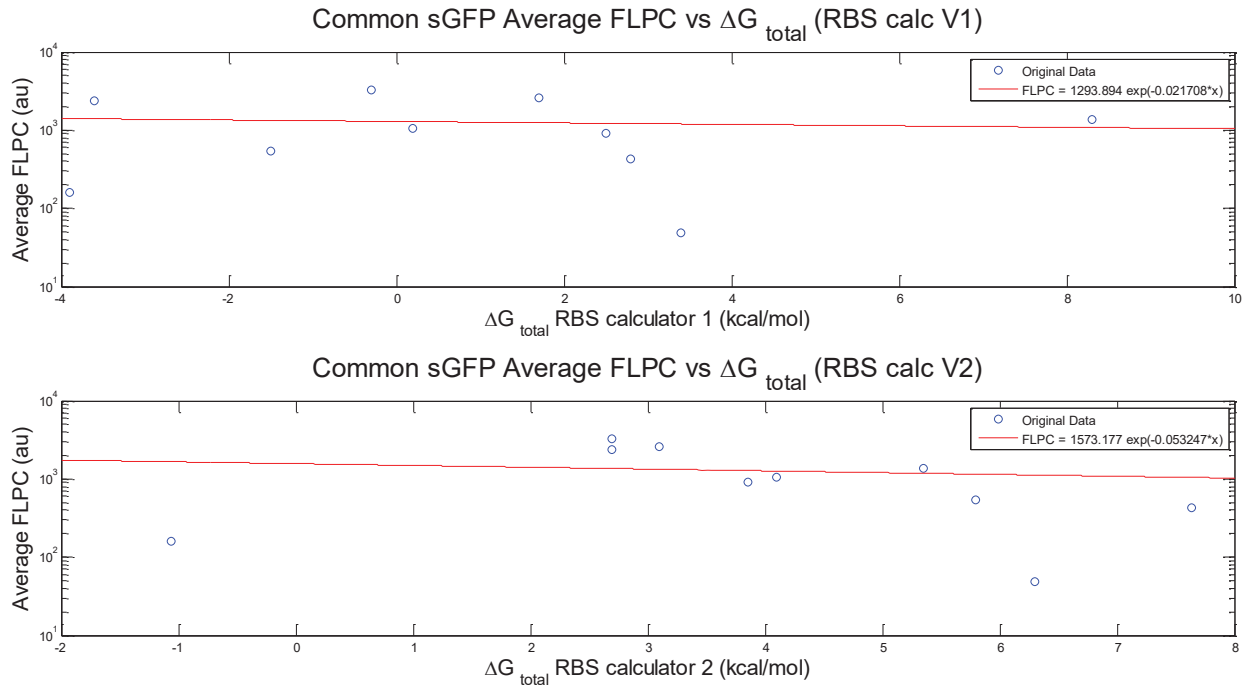
```
    rare_mat(i) = rare_codons(i);
end

common = cell2mat(optimized_genes(2,2));
common_codons=cellstr(reshape(common,3,[])');
for i=1:1:ind
    common_mat(i) = common_codons(i);
end

fast = cell2mat(optimized_genes(3,2));
fast_codons=cellstr(reshape(fast,3,[])');
for i=1:1:ind
    fast_mat(i) = fast_codons(i);
end

slow = cell2mat(optimized_genes(4,2));
slow_codons=cellstr(reshape(slow,3,[])');
for i=1:1:ind
    slow_mat(i) = slow_codons(i);
end

sit = cell2mat(optimized_genes(5,2));
sit_codons=cellstr(reshape(sit,3,[])');
for i=1:1:ind
    sit_mat(i) = sit_codons(i);
end
%Make comparisons between them
percentsim_rare_common = sum(strcmp(rare_mat,common_mat))/ind*100
percentsim_rare_fast = sum(strcmp(rare_mat,fast_mat))/ind*100
percentsim_rare_slow = sum(strcmp(rare_mat,slow_mat))/ind*100
percentsim_rare_sit = sum(strcmp(rare_mat,sit_mat))/ind*100

percentsim_common_fast = sum(strcmp(common_mat,fast_mat))/ind*100
percentsim_common_slow = sum(strcmp(common_mat,slow_mat))/ind*100
percentsim_common_sit = sum(strcmp(common_mat,sit_mat))/ind*100

percentsim_fast_slow = sum(strcmp(fast_mat,slow_mat))/ind*100
percentsim_fast_sit = sum(strcmp(fast_mat,sit_mat))/ind*100

percentsim_slow_sit = sum(strcmp(slow_mat,sit_mat))/ind*100

Make figure to show positional similarity (has rotated x axis labels)

figure
% Percent Similarity based on position
genes = {'Rare/Common';'Rare/Fast';'Rare/Slow';
'Rare/SIT';'Common/Fast';'Common/Slow';'Common/SIT';'Fast/Slow';'Fast/SIT';'Slow/SIT'}
percent_similar = [percentsim_rare_common; percentsim_rare_fast; percentsim_rare_slow;
percentsim_rare_sit; percentsim_common_fast; percentsim_common_slow; percentsim_common_sit;
percentsim_fast_slow; percentsim_fast_sit; percentsim_slow_sit]%codon adaptation index. see
document "codon usage table.xslx"
bar(percent_similar)
set(gca,'XTickLabel',{'Rare/Common';'Rare/Fast';'Rare/Slow';'Fare/SIT';'Common/Fast';'Common/Slow
';'Common/SIT';'Fast/Slow';'Fast/SIT';'Slow/SIT'},'Fontsize',14)
```

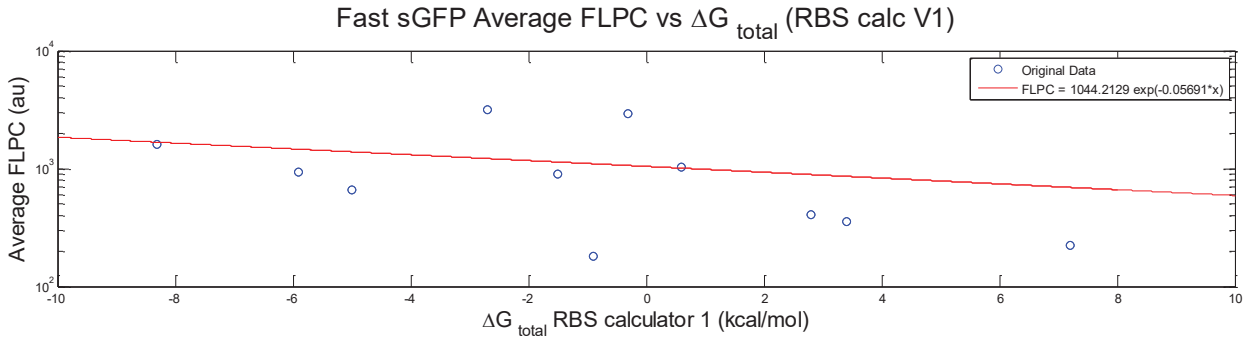
```
rotateXLabels( gca(), 45 )
title('sGFP Comparison: Positional Similarity','Fontsize',20)
ylabel('Percent Similarity (% total gene)','Fontsize',16)
%}
```

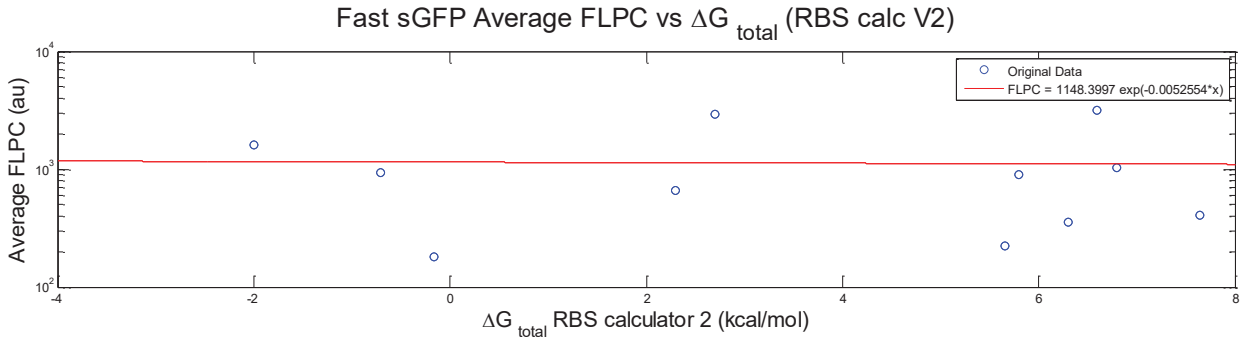
Published with MATLAB® R2014a

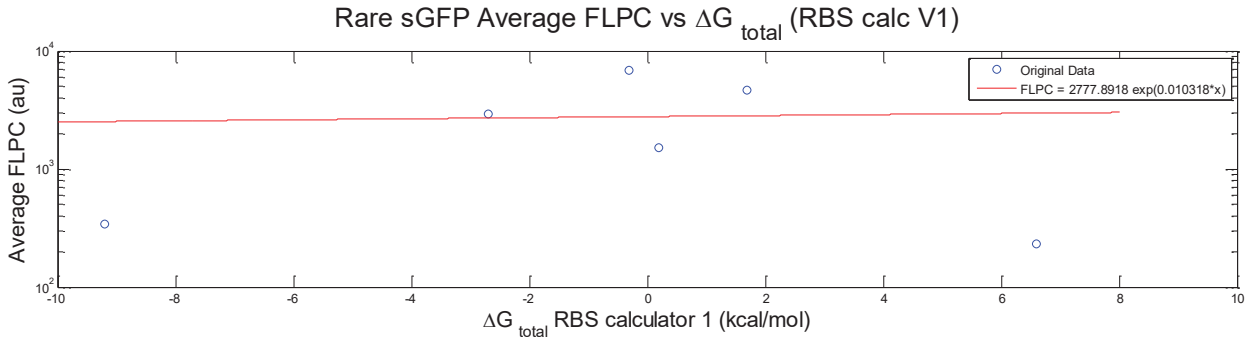
Expression of Individual Variants Genes vs ΔG_{total}

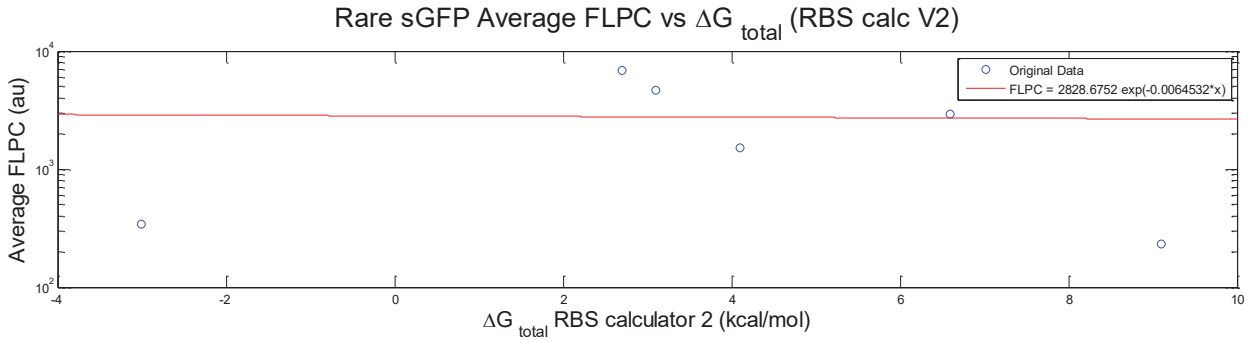
In the following figures, the relationship between FLPC and ΔG_{total} is plotted for each coding sequence individually, using the calculations of ΔG_{total} from both the RBS calculator version 1.0 and 2.0. No outliers are removed, and the fit with the exponential model is poor in all cases. When only the data from RBS calculator version 2 was used, there were far less sequences with $\Delta G_{total} < 0$, and these were removed as outliers, then the data from all coding sequences were graphed simultaneously, which is shown in Figure 36. These figures may be useful for future researchers who would be interested in the likely range of parameters β and K for this system, or would like to add future data to data for one or more specific coding sequences that were expressed in this research.

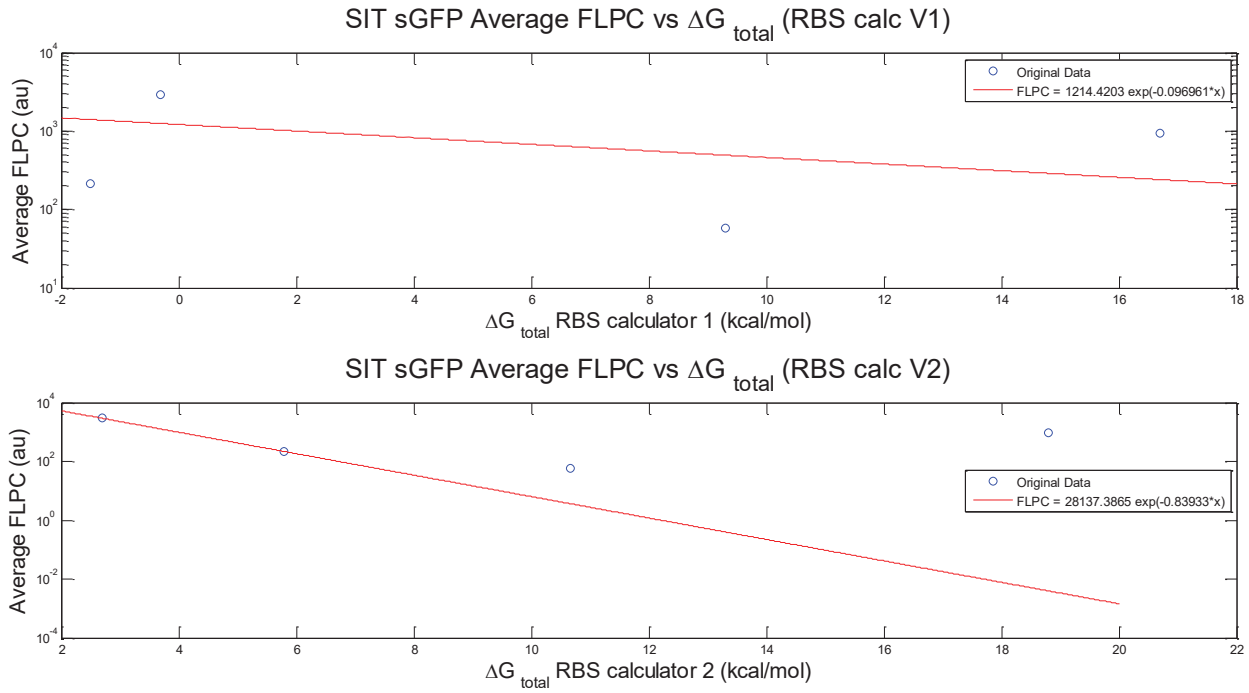



Fast sGFP Average FLPC vs ΔG total (RBS calc V1)
Original Data
FLPC = 1044.2129 exp(-0.05691*x)
Average FLPC (au)
ΔG total RBS calculator 1 (kcal/mol)


Fast sGFP Average FLPC vs ΔG total (RBS calc V2)
Original Data
FLPC = 1148.3997 exp(-0.0052554*x)
Average FLPC (au)
ΔG total RBS calculator 2 (kcal/mol)


Rare sGFP Average FLPC vs ΔG total (RBS calc V1)
Original Data
FLPC = 2777.8918 exp(0.010318*x)
Average FLPC (au)
ΔG total RBS calculator 1 (kcal/mol)


Rare sGFP Average FLPC vs ΔG total (RBS calc V2)
Original Data
FLPC = 2828.6752 exp(-0.0064532*x)
Average FLPC (au)
ΔG total RBS calculator 2 (kcal/mol)



RNA Folding Figures

This section contains the folded mRNA structure, from the promoter through the transcriptional terminator, as predicted by the ViennaRNA fold webserver [32,33].

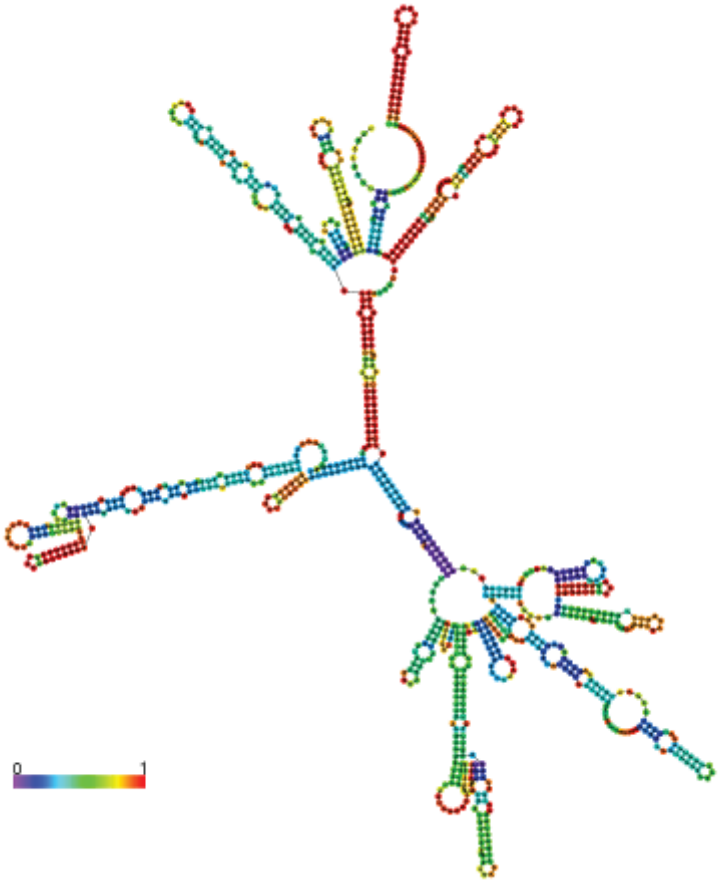


Figure 42: Predicted mRNA fold for Common sGFP

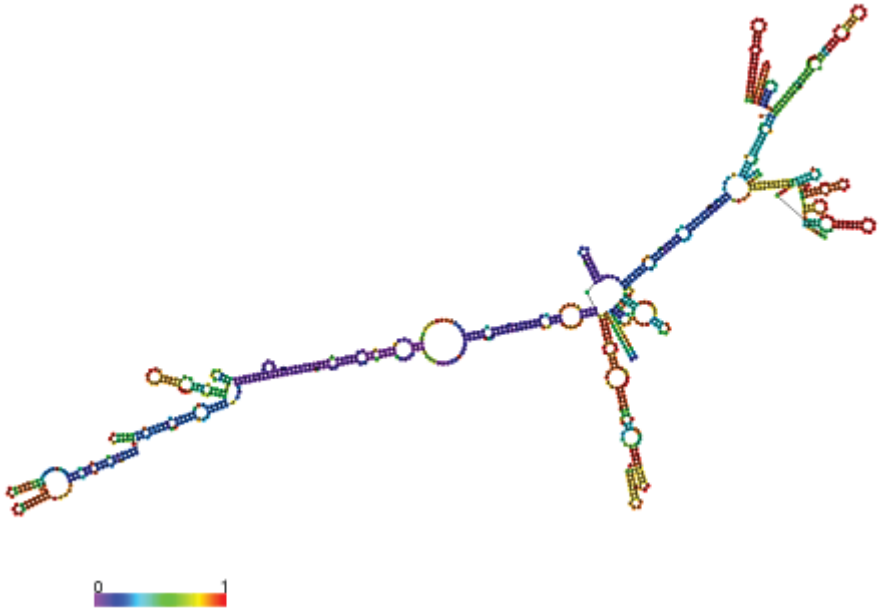


Figure 43: Predicted mRNA fold for Fast sGFP

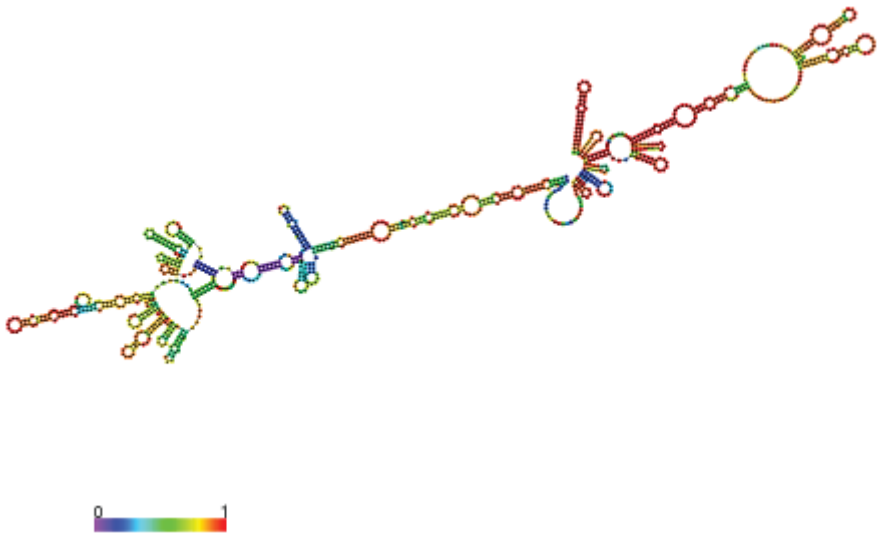


Figure 44: Predicted mRNA fold for Rare sGFP

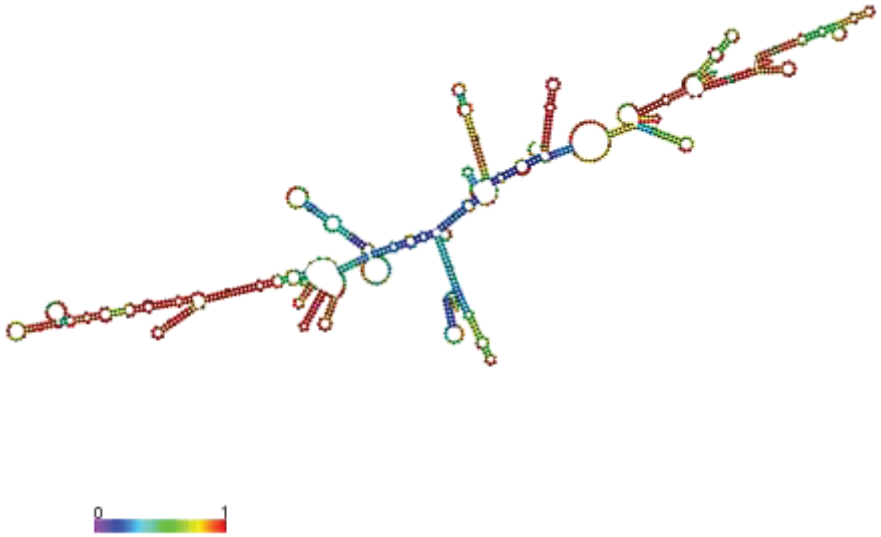


Figure 45: Predicted mRNA fold for Slow sGFP

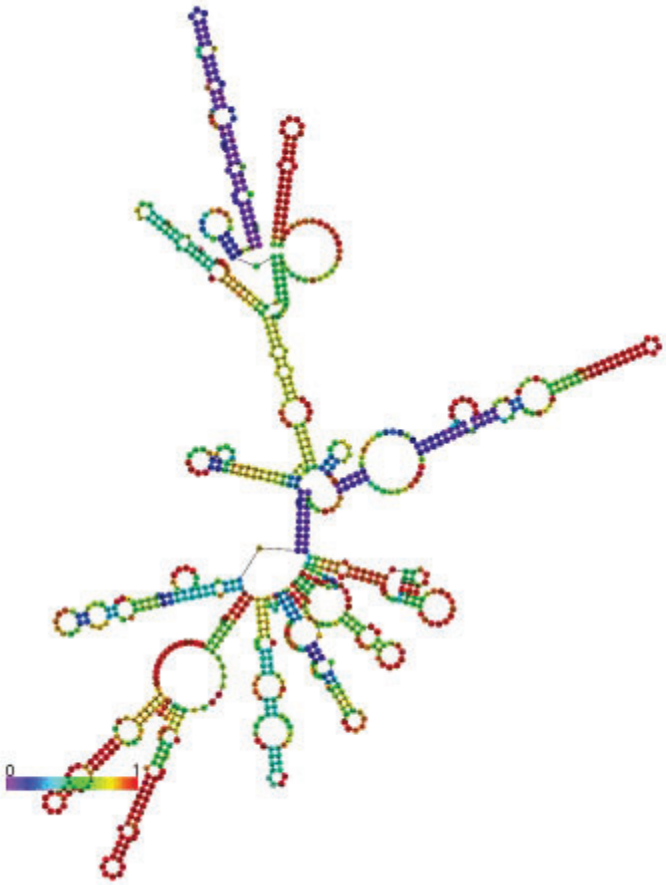


Figure 46: Predicted mRNA fold for SIT sGFP

BIBLIOGRAPHY

1. *The U.S. Biopharmaceutical industry: perspectives on future growth and the factors that will drive it*. (2014).

2. Otto, R., Santagostino, A. & Schrader, U. Rapid growth in biopharma: Challenges and opportunities. *From Sci. to Oper. Quest. Choices Strateg. Success Biopharma* 1–21 (2014). at <www.mckinsey.com/14.6.2015>

3. Glick, B., Pasternack, J. & Patten, C. *Molecular biotechnology: principles and applications of recombinant DNA*. (ASM Publishing, 2010).

4. Crick, F. Central dogma of molecular biology. *Nature* **227,** 561–563 (1970).

5. Vischer, E. & Chargaff, E. The separation and quantitative estimation of purines and pyrimadines in minute amounts. *J. Biol. Chem.* **176,** 703–714 (1948).

6. Bevilacqua, P. C. & Blose, J. M. Structures, kinetics, thermodynamics, and biological functions of RNA hairpins. *Annu. Rev. Phys. Chem.* **59,** 79–103 (2008).

7. Alberts, B. *et al. Molecular biology of the cell*. (Garland Science, 2012).

8. Brewster, R. C., Jones, D. L. & Phillips, R. Tuning promoter strength through RNA polymerase binding site design in Escherichia coli. *PLoS Comput. Biol.* **8,** e1002811 (2012).

9. Chen, T., He, H. L. & Church, G. M. Modeling gene expression with differential equations. in *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing* 29–40 (1999).

10. Pieter, L., Zupancic, M. L., Record, M. T. & Haseth, D. E. RNA polymerase-promoter interactions : the comings and goings of RNA polymerase. *J. Bacteriol.* **180,** 3019–3025 (1998).

11. Reece, J. *et al. Campbell Biology*. (Pearson, 2012).

12. Salis, H. M. The ribosome binding site calculator. *Methods Enzymol.* **498,** 19–42 (2011).

13. Shine, J. & Dalgarno, L. Determinant of cistron specificity in bacterial ribosomes. *Nature* **254,** (1975).

14. Salis, H. M., Mirsky, E. a & Voigt, C. a. Automated design of synthetic ribosome binding sites to precisely control protein expression. *Nat. Biotechnol.* **27,** 946–950 (2009).

15. Garcia, H. G. & Phillips, R. Quantitative dissection of the simple repression input-output function. *Proc. Natl. Acad. Sci.* **108,** 12173–12178 (2011).

16. Quax, T. E. F., Claassens, N. J., Söll, D. & van der Oost, J. Codon bias as a means to fine-tune gene expression. *Mol. Cell* **59,** 149–161 (2015).

17. Bentele, K., Saffert, P., Rauscher, R., Ignatova, Z. & Blüthgen, N. Efficient translation initiation dictates codon usage at gene start. *Mol. Syst. Biol.* **9,** 675 (2013).

18. Liang, S., Xu, Y., Dennis, P. & Bremer, H. mRNA composition and control of bacterial gene expression. *Am. Soc. Microbiol.* **182,** 3037–3044 (2000).

19. Sievers, A., Beringer, M., Rodnina, M. V. & Wolfenden, R. The ribosome as an entropy trap. *Proc. Natl. Acad. Sci.* **101,** 7897–7901 (2004).

20. Farasat, I. *Working results: GFP expression plateaus reaches maximum translation rate capacity at very high translation initiation rates*. (2014).

21. Reuveni, S., Meilijson, I., Kupiec, M., Ruppin, E. & Tuller, T. Genome-scale analysis of translation elongation with a ribosome flow model. *PLoS Comput. Biol.* **7,** e1002127 (2011).

22. Korana, H. G. Chemical and enzymatic synthesis of polynucleotides. *The nucleic acids* **3,** 105–136 (1965).

23. Cannarrozzi, G. *et al.* A role for codon order in translation dynamics. *Cell* **141,** 355–367 (2010).

24. Wang, J. *et al.* Codon optimization significantly improves the expression of an Amylase gene from Bacillus lichenformis in Pichia pastoris. *Biomedial Res. Int.* **2015,** 1–9 (2014).

25. Yadava, A. & Ockenhouse, C. F. Effect of codon optimization on expression levels of a functionally folded malaria vaccine candidate in prokaryotic and eukaryotic expression

systems. *Infect. Immun.* **71,** 4961–4969 (2003).

26. Zhou, Z., Schnake, P., Xiao, L. & Lal, A. a. Enhanced expression of a recombinant malaria candidate vaccine in Escherichia coli by codon optimization. *Protein Expr. Purif.* **34,** 87–94 (2004).

27. Pédelacq, J.-D., Cabantous, S., Tran, T., Terwilliger, T. C. & Waldo, G. S. Engineering and characterization of a superfolder green fluorescent protein. *Nat. Biotechnol.* **24,** 79–88 (2006).

28. Maloy, S., Stewart, V. & Taylor, R. *Genetic analysis of pathogenic bacteria*. (Cold Spring Harbor Laboratory Press, 1996).

29. Ng, C. Y., Farasat, I., Zomorrodi, A. R., Maranas, C. D. & Salis, H. M. Model-guided construction and optimization of synthetic metabolism for chemical product synthesis. in 10 (Synthetic Biology Engineering Research Center, 2013).

30. Fluitt, A., Pienaar, E. & Viljoen, H. Ribosome kinetics and aa-tRNA competition determine rate and fidelity of peptide synthesis. *Comput. Biol. Chem.* **31,** 335–46 (2007).

31. Puigbò, P., Bravo, I. G. & Garcia-Vallvé, S. E-CAI: a novel server to estimate an expected value of codon adaptation index (eCAI). *BMC Bioinformatics* **9,** 65 (2008).

32. Hofacker, I. L. Vienna RNA secondary structure server. *Nucleic Acids Res.* **31,** 3429–31 (2003).

33. Gruber, A. R., Lorenz, R., Bernhart, S. H., Neubock, R. & Hofacker, I. L. The Vienna RNA Websuite. *Nucleic Acids Res.* **36,** W70–W74 (2008).

34. Espah Borujeni, A., Channarasappa, a. S. & Salis, H. M. Translation rate is controlled by coupled trade-offs between site accessibility, selective RNA unfolding and sliding at upstream standby sites. *Nucleic Acids Res.* **42,** 2646–2659 (2014).

35. *E . Z . N . A ® Plasmid Mini Kit*. (Omega Bio-Tek, 2013). at <http://omegabiotek.com/store/wp-content/uploads/2013/05/D6942.D6943.D6945-Plasmid-DNA-Mini-Kits-I-and-II-Combo-032813-Online.pdf>

36. Brown, T. A. *Gene Cloning and Dna*. (Wiley-Blackwell, 2010).

37. Saiki, R. K. *et al.* Primer-directed enzymatic amplification of DNA with a thermostable DNA polymerase. *Science (80-.).* **239,** 487–491 (1988).

38. Madigan, M., Martinko, J., Bender, K., Buckley, D. & Stahl, D. *Brock Biology of Microorganisms*. **53,** (Pearson, 2013).

39. Demirci, A. *Engineering microbiology course manual*. (2015).

40. Kelley, L. A., Mezulis, S., Yates, C. M., Wass, M. N. & Sternberg, M. J. E. The Phyre2 web portal for protein modeling, prediction and analysis. *Nat. Protoc.* **10,** 845–858 (2015).

41. Baxevanis, A. D. & Ouellette, F. *Bioinformatics: A practical guide to the analysis of genes and proteins*. (Wiley-Interscience, 2006). doi:10.1007/s10439-006-9105-9

42. Grantham, R., Gautier, C., Mercier, R. & Pave, A. Codon catalog usage and the genome hypothesis. *Nucleic Acids Res.* **8,** 2639–2653 (1980).

43. Hershberg, R. & Petrov, D. a. Selection on Codon Bias. *Annu. Rev. Genet.* **42,** 287–299 (2008).

44. Lynn, D. J., Singer, G. a C. & Hickey, D. a. Synonymous codon usage is subject to selection in thermophilic bacteria. *Nucleic Acids Res.* **30,** 4272–4277 (2002).

45. Singer, G. a C. & Hickey, D. a. Thermophilic prokaryotes have characteristic patterns of codon usage, amino acid composition and nucleotide content. *Gene* **317,** 39–47 (2003).

46. Tuller, T. *et al.* An evolutionarily conserved mechanism for controlling the efficiency of protein translation. *Cell* **141,** 344–354 (2010).

47. Church, G., Goodman, D. & Kosuri, S. Causes and effects of N-terminal codon bias in bacterial genes. *Science (80-.).* **342,** 475–479 (2013).

48. Studer, S. M. & Joseph, S. Unfolding of mRNA secondary structure by the bacterial translation initiation complex. *Mol. Cell* **22,** 105–115 (2006).

49. Elf, J., Nilsson, D., Tenson, T. & Ehrenberg, M. Selective charging of tRNA isoacceptors explains patterns of codon usage. *Science* **300,** 1718–1722 (2003).

50. Guoy, M. & Gautier, C. Codon usage in bacteria: correlation with gene expressivity. *Nucleic Acids Res.* **10,** (1982).

51. Marshall, R. A., Aitken, C. E., Dorywalska, M. & Puglisi, J. D. Translation at the single-molecule level. *Annu. Rev. Biochem.* **77,** 177–203 (2008).

52. Agirrezabala, X. & Frank, J. Elongation in translation as a dynamic interaction among the ribosome, tRNA, and elongation factors EF-G and EF-Tu. *Q. Rev. Biophys.* **42,** 159–200 (2009).

53. Ikemura, T. Codon usage and tRNA content in unicellular and multicellular organisms. *Mol. Biol. Evol.* **2,** 13–34 (1985).

54. Sharp, P. M., Emery, L. R. & Zeng, K. Forces that influence the evolution of codon bias. *Philos. Trans. R. Soc. B Biol. Sci.* **365,** 1203–1212 (2010).

55. Chen, G. F. & Inouye, M. Suppression of the negative effect of minor arginine codons on gene expression; preferential usage of minor codons within the first 25 codons of the Escherichia coli genes. *Nucleic Acids Res.* **18,** 1465–1473 (1990).

56. Heasley, V. Effect of synonomous, consecutive, repetitive codons on bacterial translation elongation rates. (The Pennsylvania State University, 2014).

57. Maertens, B. *et al.* Gene optimization mechanisms: A multi-gene study reveals a high success rate of full-length human proteins expressed in Escherichia coli. *Protein Sci.* **19,** 1312–1326 (2010).

58. O'Brien, E. P., Ciryam, P., Vendruscolo, M. & Dobson, C. M. Understanding the influence of codon translation rates on cotranslational protein folding. *Acc. Chem. Res.* **47,** 1536–1544 (2014).

59. Elcock, A. H. Molecular simulations of cotranslational protein folding: Fragment stabilities, folding cooperativity, and trapping in the ribosome. *PLoS Comput. Biol.* **2,** 0824–0841 (2006).

60. Tsai, C. J. *et al.* Synonymous mutations and ribosome stalling can lead to altered folding pathways and distinct minima. *J. Mol. Biol.* **383,** 281–291 (2008).

61. Komar, A., Lesnik, T. & Reiss, C. Synonymous codon substitutions affect ribosome

traffic and protein folding during in vitro translation. *FEBS Lett.* **462,** 387–91 (1999).

62. Quax, T. E. F. *et al.* Differential translation tunes uneven production of operon-encoded proteins. *Cell Rep.* **4,** 938–44 (2013).

63. Tian, T. & Salis, H. M. A predictive biophysical model of translational coupling to coordinate and control protein expression in bacterial operons. *Nucleic Acids Res.* 1–15 (2015). doi:10.1093/nar/gkv635