ABSTRACT

VU, SCOTT KHOI. A New Biophysical Model for Translation Elongation to Predict and Optimize Heterologous Protein Production in *Escherichia coli*. (Under the direction of Dr. D. L. Bitzer and Dr. D.S. Lalush.)

Translation is a biological process in which an organism produces specific polypeptides that fold into functional proteins. Heterologous protein production uses translation to synthesize proteins that are not normally expressed in production organisms. These proteins are used across a broad range of industrial applications, ranging from biofuel production to agriculture and biopharmaceuticals. Unfortunately, attempts to translate unmodified heterologous genes in production organisms such as *Escherichia coli* often result in little or no synthesis of the desired protein. Some of the issues identified are poor translation, non-optimal ribosome binding sites (RBS), RBS and start codon spacing, frameshifting, premature termination, and inclusion body formation. Although maximizing protein yield has been studied in detail for some time, the underlying processes and effects of translation elongation on protein yield have not been resolved. For many years, the primary cause of low protein yield was thought to simply be codon bias and rare codons coding for rare aminoacyl-tRNA. Unfortunately, gene optimizations using codon bias sometimes work but often do not.

A new computational-based biophysical model for translation elongation has been developed that provides: a) new insights into the mechanisms behind translation, e.g. elucidating programmed frameshifts *prfB* and *dnaX*, and b) a new approach to gene optimization for heterologous protein production. The model introduces two important concepts: i) ribosome displacement and ii) ribosome interactions as a “spring.” Ribosome displacement is the misalignment between the A-site and the normal “zero” reading frame as
defined by the start codon. Misalignments are caused by a force exerted on the ribosome from an energetic “spring,” which is derived from the binding between the 16S rRNA 3’ terminal end (anti-Shine-Dalgarno region) and the mRNA. The amount of ribosome displacement is based on three factors: 1) how much the ribosome has already been displaced, 2) the amount of force exerted on the ribosome, and 3) how long the ribosome has to wait for the delivery of the next aminoacyl-tRNA, which is related to aminoacyl-tRNA abundance. These three factors are convolved to describe the ribosome “spring” model which produces new ribosome wait-time parameters. From these wait-time parameters, translation bottlenecks are identified and corrected to optimize genes for heterologous protein production.

The model’s predictive and optimization accuracy were verified by optimizing and expressing five model genes: \textit{gst} (glutathione-S-transferase), \textit{pf0132} (alpha-glucosidase), \textit{clju_c11880} (alcohol dehydrogenase), \textit{bif1} (beta-galactosidase), and \textit{rt8_0542} (endoglucanase and exoglucanase) in \textit{Escherichia coli}. The model’s predictive and optimization power was also compared to that of codon bias manipulation, which is considered the standard in gene optimization. Model-optimized genes showed higher expression levels than codon bias-optimized and wildtype genes. In addition to increased yield, model-optimized \textit{clju_c11880} (alcohol dehydrogenase) showed reduction in inclusion body formation as compared to wildtype. Multiple mRNA variants of GST were also designed to verify the model’s predictive capabilities compared to codon bias prediction. Expression levels from the nine \textit{gst} variants showed higher correlation to the model’s predictions compared to codon bias’s predictions. Wildtype and codon bias-optimized \textit{rt8_0542} from \textit{Caldicellulosiruptor} were also analyzed using the model to determine key
differences, and model optimization of r8_0542 was included as a reference for in silico comparison to codon bias optimization and wildtype variants.

Based on the results presented in this dissertation and the literature published, the author contends that the determinants of protein yield from translation elongation are not just codon bias, but, rather, the convolution between aminoacyl-tRNA usage and additional effects involving the interaction between the 16S rRNA 3’ terminal end and the mRNA, which may lead to a displaced ribosome and, therefore, increased ribosome wait times, bottlenecks, or frameshifts. The model optimizes heterologous translation by minimizing ribosome wait times and bottlenecks. The biophysical model, its computational and mathematical implementations, and case studies are described in this dissertation.
A New Biophysical Model for Translation Elongation to Predict and Optimize Heterologous Protein Production in *Escherichia coli*

by

Scott Khoi Vu

A dissertation submitted to the Graduate Faculty of North Carolina State University in partial fulfillment of the requirements for the Degree of Doctor of Philosophy

Biomedical Engineering

Raleigh, North Carolina

2015

APPROVED BY:

________________________________  ________________________________
Dr. Donald L. Bitzer     Dr. David S. Lalush
Co-Chair of Advisory Committee   Co-Chair of Advisory Committee

________________________________
Dr. Robert M. Kelly     Dr. Eric S. Miller

________________________________
Dr. Mladen A. Vouk
DEDICATION

This dissertation is dedicated to my Family and Dr. Donald L. Bitzer.
My family and I are originally from Saigon, Vietnam, which is, to date, still a communist country. My family desired freedom: the freedom to pursue a formal education and the freedom to pursue our own dreams. However, because we were anti-communist during the Vietnam War, staying in Vietnam offered us little opportunities for the future. In 1988, when I was only four years old, my family and I escaped from Vietnam by boat to build a better future for ourselves in another country. Many boat escapees never made it through the voyage; luckily, my mother, my aunt, and I were some of the very few that survived. We chose to immigrate to the United States because of the American commitment to freedom and liberty and its support for the advancement of science and education. Fortunately, the United States’ delegation of immigration accepted us, and we arrived in the United States in 1990 when I was six years old.

After persevering through many hardships, among them homelessness, we came to settle in Chapel Hill, North Carolina. Believing that education would better our future, both my mother and my aunt enrolled as a full-time college students while working full-time to support the family through school. At the age of fourteen, I enrolled as a part-time student at North Carolina State University while studying full-time in high school to further enrich my education. At the age of fifteen, I applied and was accepted as a full-time student at the university. I graduated with honors at age nineteen with triple majors in Computer Science, Electrical Engineering, and Computer Engineering. Because I wanted to contribute to society by engaging in research, teaching, and innovation, I continued my education, pursuing a
Ph.D. in Biomedical Engineering with a minor in Biotechnology and a certificate in Bioinformatics and Computational Biology. I will be the first person in my family to receive a Ph.D. degree.

During my years studying at North Carolina State University, I was fortunate to have many opportunities to teach, tutor, lead, and engage in research. When I was just fifteen, I taught college-level computer programming laboratories courses (Java and C++) and also worked as a course teaching assistant for a C++ class. Outside of formal teaching, I tutored individual students in Calculus for the Math Department and Digital Logic Design for the College of Engineering. I also engaged in leadership by being the President of Pi Kappa Pi, a Computer Science Honor Society at NC State University, for the 2003-2004 school year.

I was also very fortunate to be able to work with Dr. Donald L. Bitzer (inventor of the plasma television) as a research assistant at the age of 15, in fact, on the same project described in this dissertation. I have seen the model grow from its infancy stage of synchronous free energy signals to polar plot analysis to primitive displacement plot analysis and eventually to prediction and optimization of genes for heterologous protein production.

During my Ph.D. candidacy, I founded and led RiboLab Research, an interdisciplinary research laboratory engaged in discovering new mechanisms behind controlling protein synthesis levels in living organisms. I was also very fortunate to be able to work with some very wise and helpful professors who have taught me many valuable lessons in academia and in life. In addition, I was blessed to have the opportunity to mentor and work with some very bright and talented students who have helped make this project successful.
ACKNOWLEDGMENTS

First, I would like to give special thanks to my professors for their wisdom and guidance that made this project a success:

Thank you to Dr. Donald L. Bitzer for being my mentor since I was 15. Thank for teaching me valuable lessons in academia and life in general. You taught me how to think “creatively” and “out-of-the-box”. Many thanks for being there during my failures and successes. You are the person I can call upon to provide a solution. Thank you for all your support and mentorship over the many years. You inspire me to work on new ideas that contribute to and benefit society.

Thank you to Dr. Eric S. Miller for being my second mentor. Thank you for teaching me about life science and microbiology. Thank you for believing in the model during its infancy and for providing wet laboratory space and guidance in conducting experiments to verify the model. Many thanks for providing feedback and edits for my presentations and manuscripts. You taught me how to communicate my complex theories and data simply.

Thank you to Dr. Robert M. Kelly for your collaboration and providing target genes to verify the model. Thank you for showing me the world of biotechnology, and thank you for your critiques on the “value” of the model as it relates to biotechnology.

Thank you to Dr. David S. Lalush for providing guidance, edits, and feedback for the dissertation writing. Thank you, also, for providing the administration and structure support
during my Ph.D. candidacy and for taking your time to understand the model and algorithms and providing feedback.

Thank you to Dr. Mladen A. Vouk for inspiring me to shoot for the stars by integrating both science and industrial applications. You taught me to communicate my intuitions and theories by grounding them through first principles and published data. Most importantly, thank you for supporting the RiboScan™ project and RiboLab Research group.

Thank you to Dr. Tara Thompson for providing professional editing of this dissertation. Your editing quality was superb, and your turnaround time was extremely quick.

Second, I would like to give special thanks to my bright and talented students for their help and hard work, which made this project a success:

Thank you to Adriano Bellotti for helping with collecting the wet laboratory data, writing the RiboLab’s experimental protocols, editing the manuscripts and the dissertation, and drawing some of the figures. You are a great word smith and superb organizer. Thank you for excellent communication and teamwork.

Thank you to Chris Gabriel for help with collecting the wet laboratory data, writing up RiboLab’s experimental protocols, and editing the manuscripts. You have excellent wet laboratory workmanship. Thank you for the good teamwork.

Thank you to Hayden Brochu for help with collecting the wet laboratory data and proofreading the dissertation.
Thank you to Eric Whitmire, Tyler Cross, and Omar Estrella for their help in coding part of the model and also for their help in developing the alpha version of the RiboScan™ webserver.

Thank you to Liana Lin and Ying (Olivia) Huang for their help in general coding, gene analysis, and maintaining the RiboScan™ webserver.

Thank you to Tori Jefferson and Rangoli Bhattacharjee for standardizing assays and for general help around the laboratory.

Lastly, I would like to also thank anyone that I may have left out.

This work was supported in part by the GAANN Fellowship in Biotechnology and the Distinguished University Research Professorship funds.
# TABLE OF CONTENTS

LIST OF TABLES............................................................................................................................. xiv
LIST OF FIGURES............................................................................................................................ xv

CHAPTER 1 ......................................................................................................................................... 1
Introduction .......................................................................................................................................... 1
1.1 Background and Motivation ..................................................................................................... 1
1.2 Statement of Problem and Solutions ........................................................................................ 2
1.3 Summary and Outline of Dissertation ..................................................................................... 5
1.4 References................................................................................................................................... 8

CHAPTER 2 ....................................................................................................................................... 12
Fundamentals of Protein Synthesis in Prokaryotes......................................................................... 12
2.1 Overview of Gene Expression ................................................................................................. 12
2.2 Translation................................................................................................................................ 12
2.3 Fundamental Components of Translation ............................................................................. 13
2.4 Initiation.................................................................................................................................... 22
2.5 Elongation................................................................................................................................. 23
2.6 Termination.............................................................................................................................. 29
2.7 Reading Frame and Programmed Frameshifts..................................................................... 30
2.8 References................................................................................................................................. 33

CHAPTER 3 ....................................................................................................................................... 36
Determinants of Translation Efficiency ........................................................................................... 36
3.1 Introduction.............................................................................................................................. 36
3.2 Determinants of Protein Yield during Initiation................................................................... 37
   3.2.1 Shine-Dalgarno Sequence................................................................................................. 37
   3.2.2 Shine-Dalgarno to Start Codon Spacing........................................................................ 38
   3.2.3 GC Content and Secondary Structure during Initiation............................................... 39
   3.2.4 Optimizing Ribosome Binding Sites............................................................................... 42
3.3 Determinants of Protein Yield during Elongation................................................................. 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1 Codon Bias and Codon Bias Optimization Algorithms</td>
<td>45</td>
</tr>
<tr>
<td>3.3.2 tRNA Availability and Recharge Rates</td>
<td>49</td>
</tr>
<tr>
<td>3.3.3 16S rRNA 3’ Terminal End Binding to mRNA</td>
<td>52</td>
</tr>
<tr>
<td>3.3.4 GC Content and Secondary Structure in Open Readings Frame</td>
<td>54</td>
</tr>
<tr>
<td>3.3.5 Ribosome Displacement and Ribosome Spring Model</td>
<td>55</td>
</tr>
<tr>
<td>3.4 Conclusions</td>
<td>57</td>
</tr>
<tr>
<td>3.5 References</td>
<td>59</td>
</tr>
<tr>
<td>CHAPTER 4</td>
<td>64</td>
</tr>
<tr>
<td>A New Biophysical Model for Translation Elongation</td>
<td>64</td>
</tr>
<tr>
<td>4.1 Discovery of the Periodic Free Energy Signal</td>
<td>64</td>
</tr>
<tr>
<td>4.2 Ribosome “Spring” Model and Ribosome Displacement</td>
<td>66</td>
</tr>
<tr>
<td>4.2.1 Ribosome Spring Model</td>
<td>66</td>
</tr>
<tr>
<td>4.2.2 Ribosome Displacement</td>
<td>71</td>
</tr>
<tr>
<td>4.2.3 Translation Elongation Dynamics</td>
<td>73</td>
</tr>
<tr>
<td>4.3 Ribosome Wait Times and Translation Bottlenecks</td>
<td>75</td>
</tr>
<tr>
<td>4.3.1 Ribosome Wait Times</td>
<td>75</td>
</tr>
<tr>
<td>4.3.2 Translation Bottlenecks</td>
<td>76</td>
</tr>
<tr>
<td>4.4 Gene Optimization</td>
<td>77</td>
</tr>
<tr>
<td>4.5 Conclusions</td>
<td>80</td>
</tr>
<tr>
<td>4.6 References</td>
<td>81</td>
</tr>
<tr>
<td>CHAPTER 5</td>
<td>83</td>
</tr>
<tr>
<td>The Algorithms and Mathematics</td>
<td>83</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td>83</td>
</tr>
<tr>
<td>5.2 The Main Algorithm</td>
<td>84</td>
</tr>
<tr>
<td>5.3 The Probability Cycle</td>
<td>87</td>
</tr>
<tr>
<td>5.3.1 Output of the Probability Cycle</td>
<td>91</td>
</tr>
<tr>
<td>5.3.2 Mathematics of the Bottleneck Plot and Bottleneck Index (BNI)</td>
<td>92</td>
</tr>
<tr>
<td>5.4 Calculating Force from “Springy” 16S rRNA Tail</td>
<td>93</td>
</tr>
<tr>
<td>5.4.1 Calculating the Continuous Free Energy Signal</td>
<td>94</td>
</tr>
<tr>
<td>5.4.2 The “Big Picture” of Energetics and Force Calculations</td>
<td>103</td>
</tr>
<tr>
<td>5.5 View Curve and Aminoacyl-tRNA Abundance Table</td>
<td>104</td>
</tr>
</tbody>
</table>
5.5.1 View Curve ................................................................. 105
5.5.2 Aminoacyl-tRNA Abundance Table................................. 107
5.6 Gene Optimization Algorithm ............................................ 110
5.7 Conclusions ....................................................................... 113
5.8 References ........................................................................ 115

CHAPTER 6 ............................................................................. 117
Case Studies – Prediction and Optimization ............................... 117
6.1 Introduction ....................................................................... 117
6.2 Gene Descriptions ............................................................... 118
6.3 Comparison of Protein Yield Prediction Indices ..................... 120
   6.3.1 Design Rationale and Graphical Analysis of \textit{gst} Variants ........................................ 121
      6.3.1.1 Wildtype Graphical Plots ....................................................................................... 123
      6.3.1.2 Variant M Graphical Plots ....................................................................................... 125
      6.3.1.3 Variant VM Graphical Plots ..................................................................................... 127
      6.3.1.4 Variant 16 Graphical Plots ..................................................................................... 129
      6.3.1.5 Variant 23 Graphical Plots ..................................................................................... 131
      6.3.1.6 Variant 80 Graphical Plots ..................................................................................... 133
      6.3.1.7 Variant 82 Graphical Plots ..................................................................................... 135
      6.3.1.8 Variant 90D Graphical Plots .................................................................................. 137
      6.3.1.9 Variant 0D Graphical Plots ................................................................................... 139
   6.3.2 Experimental Procedures and Laboratory Results ............... 141
   6.3.3 Correlating GST Activity with Predictive Indices ............... 142
   6.3.4 Sensitivity Analysis of Window Sizes ................................ 146
6.4 Comparison of Gene Optimizations ....................................... 149
   6.4.1 GST, Glutathione-S-transferase ........................................ 150
      6.4.1.1 Gene Optimization and Graphical Analysis ...................................................... 150
      6.4.1.2 Experimental Procedures and Results .......................................................... 157
   6.4.2 CLJU_c11880, Alcohol Dehydrogenase .............................. 158
      6.4.2.1 Gene Optimization and Graphical Analysis ...................................................... 158
      6.4.2.1 Experimental Procedures and Results .......................................................... 162
   6.4.3 PF0132, Alpha-Glucosidase .............................................. 164
      6.4.3.1 Gene Optimization and Graphical Analysis ...................................................... 164
LIST OF TABLES

Table 4.1. Amino acid, codons, cognate tRNA, and tRNA abundance per codon. .............................. 79
Table 5.1. aa-tRNA abundance per codon using *E. coli* genomic codon frequency table. .................... 109
Table 5.2. aa-tRNA abundance per codon derived from Dong et al. ..................................................... 110
Table 6.1. Normalized GST activity and protein yield predictive indices for *gst* variants. .................. 142
Table 6.2. BNI for different window sizes and *gst* variants. ............................................................. 147
Table 6.3. BNI ratios for different window sizes and *gst* variants. .................................................. 148
Table 6.4. Percent errors for BNI ratios using different window sizes. ................................................ 149
Table 6.5. Normalized GST activity of *gst* variants and their predictive indices. ............................. 158
Table 6.6. *clju_c11880* variants and their protein yield prediction indices. ................................. 163
Table 6.7. *pf0132* variants and their protein yield prediction indices. .............................................. 168
Table 6.8. *bif1* variants and their protein yield prediction indices. .................................................. 177
Table 6.9. *rt8_0542* variants and their protein yield prediction indices. ........................................... 186
LIST OF FIGURES

Figure 2.1. Translation of mRNA by the ribosome. ................................................................. 13
Figure 2.2. Multiple views of the ribosome. ............................................................................. 15
Figure 2.3. 16S rRNA secondary structure. ............................................................................. 16
Figure 2.4. Translation is coupled with transcription in prokaryotes. ..................................... 17
Figure 2.5. The genetic code. ................................................................................................. 19
Figure 2.6. Depictions of transfer-RNA. ................................................................................. 20
Figure 2.7. General amino acid configuration. ......................................................................... 21
Figure 2.8. Translation initiation. .......................................................................................... 23
Figure 2.9. Transfer of peptide bond. ..................................................................................... 25
Figure 2.10. Steps in translation elongation. ........................................................................... 26
Figure 2.11. Recently proposed model for translation elongation........................................... 28
Figure 2.12. Translation termination....................................................................................... 30
Figure 2.13. Reading frames................................................................................................. 31
Figure 3.1. Spacing distance between Shine-Dalgarno sequence and start codon. ............... 39
Figure 4.1. Plot of the average free energy signal. ................................................................. 65
Figure 4.2. Plot of the average signal post stop codon. ............................................................ 66
Figure 4.3. Interactions between ribosomal tail and mRNA ...................................................... 67
Figure 4.4. Physical illustration during translation of prfB at frameshift site, codon 26. ........ 68
Figure 4.5. Simulation illustration during translation of prfB at frameshift site, codon 26. ..... 69
Figure 4.6. Physical illustration during translation of lacZ at codon 70. .................................. 70
Figure 4.7. Simulation illustration during translation of lacZ at codon 70. ............................. 71
Figure 4.8. Displaced vs non-displaced ribosome. ................................................................. 72
Figure 4.9. Ribosome displacement plot of prfB during translation elongation.................... 73
Figure 4.10. Ribosome wait time plot of prfB during translation elongation ......................... 75
Figure 4.11. Translation bottleneck plot of prfB throughout translation elongation. ............ 77
Figure 5.1. Flow chart of the main algorithm. ....................................................................... 86
Figure 5.2. Ribosome displacement plot of lacZ. ................................................................. 91
Figure 5.3. Ribosome wait time plot of lacZ. ...................................................................... 91
Figure 5.4. Translation bottleneck plot of lacZ during translation elongation ....................... 92
Figure 5.5. Variation in free energy based on tail:mRNA spacing....................................... 96
Figure 5.6. Simulation of ribosome of translating lacZ at codon 70 ...................................... 104
Figure 5.7. A-site when ribosome displacement is 0 ............................................................. 106
Figure 5.8. A-site when ribosome displacement is 1 ........................................................... 107
Figure 5.9. Graphical illustration of the algorithms. ............................................................. 114
Figure 6.1. Polar plot of gst wildtype. ............................................................................... 123
Figure 6.2. Ribosome displacement plot of gst wildtype....................................................... 124
Figure 6.3. Ribosome wait time plot of gst wildtype. ............................................................ 124
Figure 6.4. Translation bottleneck plot of gst wildtype. ....................................................... 124
Figure 6.5. Polar plot of gst variant M ............................................................................. 125
Figure 6.1. Ribosome displacement plot of wildtype model-optimized. .............................................. 126
Figure 6.2. Ribosome displacement plot of wildtype variant 0D. ......................................................... 126
Figure 6.3. Ribosome displacement plot of wildtype variant 16. .......................................................... 126
Figure 6.4. Ribosome displacement plot of wildtype variant 23. .......................................................... 126
Figure 6.5. Ribosome displacement plot of wildtype variant M. ........................................................... 126
Figure 6.6. Ribosome displacement plot of gst variant M. ................................................................... 126
Figure 6.7. Ribosome displacement plot of gst variant M. ................................................................... 126
Figure 6.8. Translation bottleneck plot of gst variant M. ..................................................................... 126
Figure 6.9. Polar plot of gst variant VM. .............................................................................................. 126
Figure 6.10. Ribosome displacement plot of gst variant VM. ............................................................... 126
Figure 6.11. Ribosome wait time plot of gst variant VM. .................................................................. 126
Figure 6.12. Translation bottleneck plot of gst variant VM. .............................................................. 126
Figure 6.13. Polar plot of gst variant 16............................................................................................. 128
Figure 6.14. Ribosome displacement plot of gst variant 16.............................................................. 128
Figure 6.15. Ribosome wait time plot of gst variant 16. .................................................................. 128
Figure 6.16. Translation bottleneck plot of gst variant 16. ............................................................... 128
Figure 6.17. Polar plot of gst variant 23............................................................................................ 130
Figure 6.18. Ribosome displacement plot of gst variant 23.............................................................. 130
Figure 6.19. Ribosome wait time plot of gst variant 23. .................................................................. 130
Figure 6.20. Translation bottleneck plot of gst variant 23. ............................................................... 130
Figure 6.21. Polar plot of gst variant 80.............................................................................................. 132
Figure 6.22. Ribosome displacement plot of gst variant 80........................................................... 132
Figure 6.23. Ribosome wait time plot of gst variant 80.................................................................... 132
Figure 6.24. Translation bottleneck plot of gst variant 80. ............................................................... 132
Figure 6.25. Polar plot of gst variant 82............................................................................................ 134
Figure 6.26. Ribosome displacement plot of gst variant 82........................................................... 134
Figure 6.27. Ribosome wait time plot of gst variant 82. .................................................................. 134
Figure 6.28. Translation bottleneck plot of gst variant 82. ............................................................... 134
Figure 6.29. Polar plot of gst variant 90D.......................................................................................... 136
Figure 6.30. Ribosome displacement plot of gst variant 90D........................................................... 136
Figure 6.31. Ribosome wait time plot of gst variant 90D.................................................................. 136
Figure 6.32. Translation bottleneck plot of gst variant 90D. ............................................................ 136
Figure 6.33. Polar plot of gst variant 0D............................................................................................ 139
Figure 6.34. Ribosome displacement plot of gst variant 0D. ........................................................... 139
Figure 6.35. Ribosome wait time plot of gst variant 0D. ................................................................. 139
Figure 6.36. Translation bottleneck plot of gst variant 0D. ............................................................ 140
Figure 6.37. Correlation of normalized GST activity with BNI. ..................................................... 144
Figure 6.38. Correlation of normalized GST activity with Total Wait Time ........................................ 145
Figure 6.39. Correlation of normalized GST activity and CAI ..................................................... 145
Figure 6.40. Correlation of normalized GST activity with CA1 (HEG). ........................................ 146
Figure 6.41. Polar plot of wildtype and optimized gst variants. ...................................................... 151
Figure 6.42. Displacement plot of wildtype and optimized gst variants ........................................ 152
Figure 6.43. Ribosome wait time plot of gst wildtype. ................................................................. 153
Figure 6.44. Translation bottleneck plot of gst wildtype. ............................................................ 153
Figure 6.45. Ribosome wait time plot of gst model-optimized. ................................................... 154
Figure 6.46. Translation bottleneck plot of gst model-optimized. ................................................ 154
Figure 6.47. Ribosome wait time plot of gst “codon bias-optimized 1”.......................................................... 155
Figure 6.48. Translation bottleneck plot of gst “codon bias-optimized 1”.................................................... 155
Figure 6.49. Ribosome wait time plot of gst “codon bias-optimized 2”......................................................... 156
Figure 6.50. Translation bottleneck plot of gst “codon bias-optimized 2”.................................................... 156
Figure 6.51. Polar plot of wildtype and optimized clju_c11880................................................................. 159
Figure 6.52. Displacement plot of wildtype and optimized clju_c11880........................................................ 159
Figure 6.53. Ribosome wait time plot of wildtype...................... 160
Figure 6.54. Translation bottleneck plot of wildtype................................................................. 160
Figure 6.55. Ribosome wait time plot of model-optimized......................................................... 161
Figure 6.56. Translation bottleneck plot model-optimized .......................................................... 161
Figure 6.57. SDS-PAGE gel showing inclusion body production from of clju_c11880 variants.............. 164
Figure 6.58. Polar plot of pf0132 variants................................................................. 165
Figure 6.59. Displacement plot of pf0132 variants........................................................................ 165
Figure 6.60. Ribosome wait time plot of wildtype pf0132. .......................................................... 166
Figure 6.61. Translation bottleneck plot wildtype pf0132......................................................................... 166
Figure 6.62. Ribosome wait time plot of model-optimized pf0132............................................................. 167
Figure 6.63. Translation bottleneck plot of model-optimized pf0132....................................................... 167
Figure 6.64. SDS-PAGE of wildtype and model-optimized pf0132..................................................... 169
Figure 6.65. Polar plot of wildtype and optimized bif1 variants.......................................................... 171
Figure 6.66. Ribosome displacement plot of wildtype and optimized bif1 variants................................. 171
Figure 6.67. Ribosome wait time plot of wildtype bif1................................................................. 173
Figure 6.68. Translation plot bottleneck plot of wildtype bif1............................................................... 173
Figure 6.69. Local GC content along the coding region of wildtype bif1................................................. 173
Figure 6.70. Ribosome wait time plot of model-optimized bif1............................................................. 174
Figure 6.71. Translation plot bottleneck plot of model-optimized bif1.................................................. 174
Figure 6.72. Local GC content along the coding region of model-optimized bif1.................................... 174
Figure 6.73. Ribosome wait time plot of codon bias-optimized bif1....................................................... 175
Figure 6.74. Translation plot neck plot of codon bias-optimized bif1...................................................... 175
Figure 6.75. Local GC content along the coding region of codon bias-optimized bif1................................ 175
Figure 6.76. SDS-PAGE of wildtype and optimized bif1, a β-galactosidase............................................ 178
Figure 6.77. Polar plot of wildtype and optimized rt8_0542 variants..................................................... 180
Figure 6.78. Ribosome displacement plot of wildtype and optimized rt8_0542 variants....................... 181
Figure 6.79. Ribosome wait time plot of wildtype rt8_0542................................................................. 183
Figure 6.80. Translation plot neck plot of wildtype rt8_0542............................................................... 183
Figure 6.81. GC content distribution along the gene for wildtype rt8_0542........................................... 183
Figure 6.82. Ribosome wait time plot of model-optimized rt8_0542..................................................... 184
Figure 6.83. Translation bottleneck plot of model-optimized rt8_0542................................................... 184
Figure 6.84. GC content distribution along the gene for model-optimized rt8_0542.......................... 184
Figure 6.85. Ribosome wait time plot of codon bias-optimized rt8_0542............................................... 185
Figure 6.86. Translation plot neck plot of codon bias-optimized rt8_0542.............................................. 185
Figure 6.87. GC content along the gene using for codon bias-optimized rt8_0542................................. 185
Figure 6.88. SDS-PAGE gel of codon bias-optimized rt8_0542 ................................................................. 187
Figure A.1. Calculating free energy from exposed tail binding with mRNA. ............................................. 211
Figure B.1. Polar plot of prfB .................................................................................................................. 214
Figure B.2. Polar plot of lacZ .................................................................................................................. 215
Figure C.1. Polar plot of E. coli genes from GenBank. ................................................................. 218
Figure C.2. Box plot of genes’ average phase angle with respect to length ........................................... 218
Figure C.3. Polar plot of highly-expressed genes from Karlin et al ...................................................... 219
Figure C.4. Displacement plot of E. coli genes from GenBank .............................................................. 219
Figure C.5. Polar plot of tufA ................................................................................................................. 220
Figure C.6. Ribosome displacement plot of tufA .................................................................................... 220
Figure C.7. Ribosome wait time plot of tufA .......................................................................................... 221
Figure C.8. Translation bottleneck plot of tufA ...................................................................................... 221
Figure C.9. Polar plot of ompA .............................................................................................................. 222
Figure C.10. Ribosome displacement plot of ompA ............................................................................... 222
Figure C.11. Ribosome wait time plot of ompA .................................................................................... 223
Figure C.12. Translation bottleneck plot of ompA .................................................................................. 223
Figure C.13. Polar plot of fusA ............................................................................................................. 224
Figure C.14. Ribosome displacement plot of fusA ................................................................................ 224
Figure C.15. Ribosome wait time plot of fusA ...................................................................................... 225
Figure C.16. Translation bottleneck plot of fusA ................................................................................... 225
Figure C.17. Polar plot of galR .............................................................................................................. 226
Figure C.18. Ribosome displacement plot of galR ................................................................................ 226
Figure C.19. Ribosome wait time plot of galR ...................................................................................... 227
Figure C.20. Translation bottleneck plot of galR .................................................................................. 227
Figure C.21. Polar plot of lacI ............................................................................................................... 228
Figure C.22. Ribosome displacement plot of lacI .................................................................................. 228
Figure C.23. Ribosome wait time plot of lacI ....................................................................................... 229
Figure C.24. Translation bottleneck plot of lacI ................................................................................... 229
Figure C.25. Polar plot of dnaE ............................................................................................................ 230
Figure C.26. Ribosome displacement plot of dnaE ............................................................................... 230
Figure C.27. Ribosome wait time plot of dnaE ...................................................................................... 231
Figure C.28. Translation bottleneck plot of dnaE ............................................................................... 231
Figure E.1. Insertion of gene into pBAD/Myc-His C plasmid ................................................................. 246
CHAPTER 1

Introduction

1.1 Background and Motivation

Proteins play critical roles in human life and society, from advancing agriculture to improving medicine, and researchers have placed a major emphasis on the study of protein production. However, the extraction of proteins directly from their source organisms is not very effective; for instance, extracting human proteins from human cells is neither efficient nor safe, and it brings forth ethical issues. A process has been developed called heterologous protein production to produce proteins not natively synthesized in a production organism (e.g. bacteria, yeast, or mammalian cells). These production organisms are easier to grow and harvest, have limited ethical implications, and are generally “benign” with respect to environmental safety. Examples of “high value” heterologous industrial proteins that are produced today include human insulin (for treating diabetes) [19], human interleukin (for treating cancer) [13], and human growth hormone (for treating growth hormone deficiency) [33]. Examples of often-used production organisms are Escherichia coli, Bacillus subtilis, Pichia pastoris (yeast), Chinese hamster ovarian cells (CHO cells), Aspergillus, Arabidopsis thaliana, and tobacco [6,9,10,26,27,29,38,44,48].

Heterologous protein production [44] takes advantage of an organism’s native-biological process called “translation” for protein synthesis. Translation uses ribosomes to translate messenger-ribonucleic acids (mRNAs) into specific amino acid chains (polypeptides) which fold into functional proteins [1,32]. Producing heterologous protein
often involves cloning the gene that encodes the protein into a plasmid and inserting that plasmid into a production organism for protein production; DNA can also be cloned into the chromosome for protein expression. Plasmids are circular DNA that reside in the cell apart from the chromosome. They are capable of replicating independently from the chromosome and contain the necessary regulatory signals for replication, transcription, and translation of the genes [12].

Unfortunately, heterologous protein production can be a hit-or-miss process that requires expensive, laborious trial-and-error laboratory work. Laboratory experiments attempting to circumvent these issues and improve protein production are highly variable and empirical for each gene, often with limited success rates. New approaches are needed for improved modeling and gene design that are likely to yield greater success rates for heterologous protein synthesis.

1.2 Statement of Problem and Solutions

Heterologous protein production is very complex and has many problems that can reduce protein yields. Attempts to translate unmodified exogenous genes in production organisms, such as *Escherichia coli*, often result in little to no expression of the desired protein. Some of these problematic issues are poor translation [24], non-optimal ribosome binding sites (RBS) [4,8,35], RBS to start codon spacing [5,7], frameshifting [20,37,47], premature truncation [34], and inclusion body formation [17,28,45]. Non-optimal RBS and spacing problems occur during initiation of translation; optimizing the RBS and the spacing recruits more ribosomes to initiate translation. Frameshifting occurs during translation elongation where the ribosome inadvertently shifts reading frames and produces an aberrant
polypeptide. Premature truncation happens when the ribosome pauses too long during elongation and dissociates from the mRNA or when the ribosome frameshifts and encounters a stop codon. Protein inclusion bodies are caused by several factors, but the primary reason is thought to be improper protein folding [3,28,49]. Although maximizing protein yield has been studied in detail for some time, the underlying processes and effects of translation elongation on protein yield have not been resolved. The intrinsic cause of low protein yield was originally thought to be codon bias [15,23,52] and rare codons coding for rare aminoacyl-tRNA (aa-tRNA) [21]; codon bias is defined as the organism’s preference for using a certain set of codons. However, recent experimental evidence suggests that low protein yield may result from a number of equally important additional factors [2,22,30,46].

Solutions have been developed to increase the yield of heterologous protein synthesis. One solution is to over-transcribe DNA into mRNA using a “hot” RNA promoter such as that found in the T7 bacteriophage [40,41]. This significantly increases the quantity of mRNA transcribed and thus also increases the production of protein. However, over-expression of proteins from over-transcribing mRNA causes the production organism to over-exert its resources, leading to growth inhibition, production of unwanted aggregates, and cell death [11]. The second solution is to circumvent problems caused by rare aa-tRNA by increasing the tRNA abundance encoding for rare codons. E. coli strains like Merck’s Rosetta [43] and Agilent’s Artic Express have been engineered to contain plasmids that express these rare aa-tRNAs and chaperones to aid in translation and protein folding. The third solution is to redesign the genes completely using gene synthesis techniques [18]. The gene can be redesigned to match the codon bias of the production organism using codon bias optimization.
algorithms [14,16,31]. These algorithms match the codon usage of the gene to the codon usage of highly-expressed reference genes from the production organism. The current “standard” index used to determine a gene’s codon bias and to predict protein yield is the codon adaptation index (CAI) by Sharpe et al. [36] (see Chapter 3.3.1). While codon bias optimization sometimes works [15,51,53,54], it often does not produce the more optimal sequence [2,22,46,51], and it can even be detrimental to protein yield [15,50,51,53,54]. For example, out of 94 case studies published by Maertens et al. [50], 19 cases showed negative improvement, 21 cases showed no expression level, and 3 cases showed no improvement after codon bias optimization. According to the results published by Maertens et al. [50], codon bias optimization has a 46% failure rate (43 out of 94 cases failed). Another study of codon bias optimization was published by Burgess-Brown et al. [53]. Burgess-Brown et al. [53] showed that out of 30 cases, 11 cases showed no improvement, and one case showed negative improvement after codon bias optimization. The results published by Burgess-Brown et al. demonstrated that codon bias optimization has a 40% failure rate (12 out of 30 cases failed).

The state-of-the-art example in gene optimization methods is that of Welch et al. [46,51] and is available through a service from DNA 2.0, Inc. The algorithms used by DNA 2.0, Inc - genetic algorithm and partial least square regression - are heuristic, i.e., testing and learning algorithms. Welch et al. [46] used a random codon generation approach to synthesize various genes, expressed these genes in *E. coli*, and used the resulting protein yields to train their heuristic algorithms. From these results, they re-trained their algorithm to produce another set of variants based on initial expression levels. They repeated this process
until the algorithm differentiated between highly-expressed genes and poorly-expressed genes. Their algorithm first deduced that certain stretches of codons should be avoided and, secondly, provided a codon usage pattern for gene optimization. Welch et al. [46] also observed that the “preferred” *E. coli* codon bias changed to a different “preferred” codon bias. They suggested that amino acid starvation during overexpression caused the depletion of available aa-tRNA. They also showed that genes optimized using *E. coli* codon bias expressed poorly compared to genes optimized by their algorithm. However, genes optimized by their algorithm still did not produce the best possible yield with expression levels only reaching ~60% of the yield for their best possible variant. Heuristic algorithms like those employed by DNA 2.0, Inc. only capture 70% of the information at best and are likely to get stuck in a local maximum [39,42]. Therefore, the best possible gene expression level is not likely to be achieved.

The aforementioned solutions to poor protein production are based on simple pattern observations rather than by exploring the fundamental mechanisms of protein synthesis. A more effective method of improving protein expression should consider the biophysical interactions involving the ribosome at the molecular level. An example of such a method is detailed in this dissertation.

1.3 Summary and Outline of Dissertation

A new computational-based biophysical model for translation elongation has been developed that provides: a) new insights into the mechanisms behind translation and b) a new approach to gene optimization for heterologous protein production. The model introduces two important concepts: i) ribosome displacement and ii) ribosome interactions as a “spring.”
Ribosome displacement is the misalignment between the A-site and the normal “zero” reading frame as defined by the start codon. Misalignment is due to secondary ribosome movements (whereas translocation is the primary movement) upstream or downstream while it waits for the delivery of the next aa-tRNA. This movement is caused by a force exerted on the ribosome from an energetic “spring” derived from the binding between the 16S rRNA “exposed tail” (anti-Shine-Dalgarno region) and the mRNA. The amount that the ribosome is displaced is based on three factors: 1) how much the ribosome has already been displaced, 2) the amount of force exerted on the ribosome, and 3) how long the ribosome has to wait for the delivery of the next aa-tRNA (related to aa-tRNA abundance). These three factors are convolved together to describe the “spring” model which produces new ribosome wait-time parameters. From these wait-time parameters, translation bottlenecks are identified and corrected to optimize genes for heterologous protein production. The computational and mathematical implementations of the model for gene sequence analysis and its optimization are described in this dissertation.

The model’s predictive and optimization accuracy were verified by optimizing and expressing five model genes: \textit{gst} (glutathione-S-transferase), \textit{pf0132} (alpha-glucosidase), \textit{clju_c11880} (alcohol dehydrogenase), \textit{bifl} (beta-galactosidase), and \textit{rt8_0542} (endoglucanase and exoglucanase) in \textit{Escherichia coli}. The model’s predictive and optimization power was also compared to that of codon bias manipulation, such as the CAI, which is considered the standard in gene optimization [14,16,31,36]. Multiple mRNA variants of GST were also designed to further verify the model’s predictive capabilities when compared to codon bias (CAI) prediction. Optimized genes yielded increased expression
compared to codon bias-optimized and wildtype versions, and expression levels from nine
GST variants showed high correlation to the model’s predictions and lower correlation to the
CAI. In addition to increased yield, optimized clju_c11880 (alcohol dehydrogenase) also
showed a reduction in inclusion body formation when compared to wildtype. Wildtype and
codon bias-optimized rt8_0542 from Caldicellulosiruptor were also analyzed using the
model to determine key differences. Model optimization of rt8_0542 was also included as a
reference for in silico comparison to codon bias optimization and wildtype variants.

Chapter 2 summarizes the fundamentals of protein synthesis. Chapter 3 contains a
literature review on the determinants of translation efficiency. Chapter 4 describes how the
biophysical model (the ribosome “spring” model) was developed from +1 programmed
frameshifting gene prfB using a free-energy base approach and how the principles of the
biophysical model are applied throughout translation elongation. Chapter 5 describes the
algorithms and mathematics behind the biophysical model and illustrates the optimization
strategy and optimization algorithm. Chapter 6 describes the rationale for gene designs used
to verify the model’s prediction and optimization capabilities and also reports on the
laboratory results from these gene designs. Chapter 7 summarizes the biophysical model,
describes the limitations of the model, and briefly proposes future research. Appendix A
describes the method used to calculate free energy values. Appendix B shows how to
calculate the polar plots and the phase angle. Appendix C presents the analysis of E. coli
genes using the model. Appendix D contains the DNA sequences of heterologous genes used
in this dissertation. Appendix E describes the laboratory procedures in detail.
1.4 References


CHAPTER 2
Fundamentals of Protein Synthesis in Prokaryotes

2.1 Overview of Gene Expression

Organisms store information in deoxyribonucleic acid (DNA) [15], which is used to encode and direct synthesis of proteins. Information in the genome is transferred to proteins in two stages: the first stage is transcription, and the second stage is translation. During transcription, the enzyme RNA polymerase synthesizes messenger-ribonucleic acid (mRNA) from DNA. In terms of structure, mRNA is identical to one strand of the double-stranded DNA, except that uracil replaces thymine, and the sugar molecule is a ribose rather than deoxyribose. The second process, translation, involves the synthesis of proteins by ribosomes, mRNAs, transfer-ribonucleic acids (tRNAs), and amino acids. The scope of this work focuses on translation elongation; therefore, transcription will not be summarized. The reader is referred to [18] for an overview of transcription.

2.2 Translation

Translation is the process in which functional proteins are synthesized from amino acids using mRNA as the coding sequence. The fundamental components of translation are ribosomes, mRNAs, tRNAs, and amino acids. The ribosome decodes the mRNA to assemble the amino acids into a polypeptide chain, which then folds into a functional protein. tRNAs are used to deliver the next amino acid to the ribosome where the amino acid is attached to the polypeptide chain. Translation is split into three stages: initiation, elongation, and termination. During initiation, the ribosome subunit forms a complex to begin elongation.
During elongation, the ribosome translocates the mRNA in the 5’ to 3’ direction; at this stage, the ribosome synthesizes a growing polypeptide chain using aminoacyl-tRNAs (aa-tRNA). Termination is when the ribosome recognizes a release factor at the stop codon and dissociates. See Figure 2.1 for an illustration of translation.

![Figure 2.1](image)

**Figure 2.1. Translation of mRNA by the ribosome.** Translation of the mRNA is split into three stages: initiation, elongation and termination. At initiation, the ribosome 50S and 30S subunits form at the ribosome binding site (RBS) to begin elongation. During elongation, the ribosome moves three bases (a codon) at a time to synthesize a growing polypeptide chain. The ribosome dissociates by recognizing a release factor coded by a stop codon at termination.

### 2.3 Fundamental Components of Translation

**Ribosomes**

Ribosomes function to decode the mRNAs and synthesize proteins using amino acids. Ribosomes consist of RNA and proteins assembled in a structure known as a ribonucleoprotein complex. The entire ribosome is called the 70S unit, which has a molecular weight of about 2.5 mega-Daltons and a diameter of 210 to 250 angstroms [1,3,16]. The two subunits of the ribosome are the larger 50S subunit and the smaller 30S subunit. The “S” refers to the Svedberg coefficient and is named after Theodor Svedberg; it measures the rate that molecules settle in a centrifugal field. The 50S subunit consists of the 23S and 5S ribosomal RNAs (rRNA) as well as 36 ribosomal proteins. The 30S subunit consists of the
16S rRNA and 21 different ribosomal proteins. During translation, the ribosome clamps onto the encoded mRNA and synthesizes proteins. The ribosome contains three functional sites for tRNA: E (exit) site, P (peptidyl) site, and the A (aminoacyl) site. The tRNA lands at the A-site, transfers the polypeptide chain at the P-site, and exits the ribosome at the E-site. The distance between each site is about 50-55 angstroms, and the distance between each codon on the mRNA is about 12 angstroms. Structure analysis suggests that the ribosome acts as a “channel” through which the mRNA flows, as shown in Figure 2.2.

New evidence from real time fluorescence resonance energy transfer (FRET) and cryo-electron microscopy suggests that the ribosomes move along the mRNA in a ratchet-like manner [1,2,13]. The ratcheting consists of the 30S subunit rotating approximately 8 to 9 degrees counterclockwise away from the 50s subunit (rotated or hybrid configuration) and then rotating clockwise back towards the 50S subunit (classic configuration) to move three nucleotides [1,3,13]. The energy required to change the ribosome configuration is thought to come from IF2 GTP hydrolysis at initiation and EF-G GTP hydrolysis during elongation. However, the mechanism by which the ribosome transduces hydrolysis energy to kinetic energy is still unclear [1].
The 16S rRNA, which is part of the 30S subunit, contains a nucleotide sequence at the 3’ terminal end called the “exposed tail” (3’-AUUCCUCCACUAG-5’) [28]. This sequence contains the anti-Shine-Dalgarno sequence (aSD) as shown in Figure 2.3 and binds to the mRNA Shine-Dalgarno (SD) sequence at ribosome binding sites during initiation [34]. This exposed tail has also been shown to continuously interact with the mRNA during translation elongation [26]. The aSD region of exposed tail was also shown to play a critical role in producing programmed frameshifts [13,21,22], which also implied 16S tail interactions with the mRNA during elongation.
Figure 2.3. 16S rRNA secondary structure. Boxed in green is the 3’ terminal end “exposed tail” which contains the anti-Shine-Dalgarno sequence. The “exposed tail” is known to interact with the mRNA during initiation and elongation; it is also an important component to produce ribosome frameshifts in the translation of \textit{prfB} and \textit{dnaX}. Image reproduced from CRW database [27] with permission from CRW group.
Messenger RNA (mRNA)

mRNAs are a family of RNA molecules that convey genetic information from the DNA to the ribosome. In general, prokaryotes do not require processing of mRNA after transcription; therefore, translation is considered to be directly coupled to transcription, as shown in Figure 2.4 [17]. Because mRNA is single-stranded and lacks a hydroxyl group (-OH) on the ribose, it is highly unstable and susceptible to many forms of decay (by ribonucleases). The coding region of mRNAs begins with a start codon (AUG, GUG, or UUG) and ends with a stop codon (UAA, UAG, or UGA). The open reading frame (ORF), which is the region between the start and stop codon, contains the nucleotides that specify the amino acids sequence during translation. The open reading frame is typically preceded by an initiation site, which consists of the Shine-Dalgarno appropriately spaced (5 – 12 bases) upstream from the start codon.

Figure 2.4. Translation is coupled with transcription in prokaryotes. This image shows the growth of multiple peptide chains from multiple ribosomes directly after transcription – a phenomena thought to be exclusive to prokaryotes.
**Codons and the Genetic Code**

Codons are groups of three nucleotide bases which code for initiator tRNAs like fMet-tRNAs (initiation), amino acids (elongation), or release factors (termination) [5]. The start codon recruits fMet-tRNA which interacts with the 30S to initiate the elongation phase while the stop codon recruits release factors to terminate translation. Codons between the start and stop codon recruit tRNAs that carry amino acids (aminoacyl-tRNAs). During elongation, the amino acids that are recruited to the polypeptide chain are highly dependent on decoding the proper reading frame (see Chapter 2.7 reading frame). The genetic code is “degenerate,” meaning that the first position and third position of a codon can differ but still code for the same amino acid; this is otherwise known as the “wobble” phenomenon [4], as shown in Figure 2.5. This degeneracy results in 61 codons specifying 20 amino acids and 3 stop codons for the release factors. For example, the amino acid leucine can be coded using 6 synonymous codons (UUA, UUG, CUU, CUC, CUA, or CUG), albeit using 5 different tRNA species [30].
Figure 2.5. The genetic code. This table outlines the conversion between nucleotide sequence and amino acid sequence in which three nucleotide groups called codons are translated to one amino acid. The degeneracy of the code can be seen here, and the phenomena of synonymous codons become clear after studying this code. Image reproduced from Wikimedia Commons under Public Domain Mark 1.0, no copyright restrictions [30].

Transfer-RNAs (tRNAs)

tRNAs are structured ribonucleic acids containing 75-100 nucleotides that form a three-dimensional L-form structure (Figure 2.6b) but are often drawn as two-dimensional “cloverleaf” structures to more easily show the functional regions (Figure 2.6a). Amino acids are attached onto tRNAs by aminoacyl tRNA synthetases. Thus, tRNAs carrying amino acids are called aminoacyl-tRNAs (aa-tRNA) or charged tRNAs. The acceptor end of the tRNA binds an amino acid for transportation, while the anti-codon end contains the tri-nucleotide sequence which pairs with the mRNA codon during protein synthesis. tRNAs
transport the correct amino acid to the polypeptide chain on the ribosome during translation elongation. For *E. coli*, there are currently 46 different known species of tRNA transferring 20 different amino acids to the ribosome [29]. For example, there are 5 different leucine tRNAs (Leu1, Leu2, Leu3, Leu4, Leu5) [29]. However, the intra-cellular abundance differs for each of those leucine tRNAs, and these abundances also change under different growth conditions. It is also known that the cell recharges these “synonymous” tRNAs at different rates under different states of amino acid abundance [6,7,31].

![Depictions of transfer-RNA](image)

**Figure 2.6. Depictions of transfer-RNA.** a) Transfer-RNAs can be depicted as a two-dimension cloverleaf structure. Image reproduced with modifications from Wikimedia Commons [32]. b) Transfer-tRNAs can also be depicted as a three-dimensional L-shaped molecule. Image reproduced with modifications from Wikimedia Commons [33]. Both images reproduced under Creative Commons Attribution-Share Alike 3.0 Unported license.
**Amino Acids**

Amino acids are the monomeric units that compose the polypeptides (or proteins). In total, there are 23 amino acids, 20 of which can be coded during the process of translation. Amino acids are composed of an amine functional group, a carboxylic acid functional group, a hydrogen, and a side-chain (R group) bonded to the α-carbon shown in Figure 2.7. The properties of the side chains, including hydrophobicity, polarity, and charge, are integral to proper protein folding and eventual protein function. To polymerize, the two functional groups form an amide (peptide) bond in a condensation reaction. This is essential to the formation of the polypeptide chain during translation.

**Figure 2.7. General amino acid configuration.** This general model of an amino acid shows the carboxyl group, the amino group, the hydrogen atom, and the variable side chain, the last of which is responsible for the properties (acidity, polarity, affinity) of polypeptides and proteins. Image reproduced from Wikimedia Commons under Creative Commons Attribution 3.0 Unported license [35].
2.4 Initiation

Prior to initiation, the ribosome has dissociated into its 30S and 50S subunits. During initiation, the 30S subunit of the ribosome attaches to the mRNA at the Shine-Dalgarno sequence upstream from the start codon [34]. The Shine-Dalgarno (SD) sequence (AGGAGG - the consensus sequence in E. coli) is a Watson-Crick complement to the 3’ end anti-Shine-Dalgarno (aSD) of the 16S rRNA [34]. Once bound, the 30S subunit-mRNA complex is further stabilized by initiation factor 1 (IF1), initiation factor 3 (IF3), and initiator tRNA (fMet-tRNA). IF1 and IF3 guide the 30S assembly by placing the mRNA and initiator tRNA (fMet-tRNA) into the P-site of the 30S and also prevent premature association of the 50S and aa-tRNA. fMet-tRNA is an initiator tRNA that is uniquely charged with N-formyl-methionine, which prevents it from binding with the C-terminus of the previously translated polypeptide. This ensures that N-formyl-methionine is always at the beginning of a polypeptide. Initiation factor 2 (IF2) binds to 30S preinitiation complex and recognizes the formyl group on the initiator tRNA; this further stabilizes the 30S:initiator tRNA interaction. Marshall et al. [10] showed that binding of IF2 to the 30S initiation complex accelerated the joining of the 50S subunit to the 30S subunit to form the 70S initiation complex under the rotated configuration. Their experiments also showed that GTP hydrolysis by IF2 brings the 70S ribosome back into the classic (non-rotated) configuration; in the absence of IF2 and GTP hydrolysis, the ribosome does not enter the elongation phase [10]. The initiation factors and GDP then disassociate from the ribosome:mRNA:initiation-tRNA complex, and translation enters into the elongation phase ready to accept the first aminoacyl-tRNA (aa-tRNA). The stages of translation initiation are depicted in Figure 2.8.
Uemura et al. [14] used the optical tweezer method to measure the rupture force between the ribosome initiation complex and the mRNA. This experiment tested the stability of the ribosome initiation complex resulting from the SD-aSD binding. They found that the rupture force was higher on ribosome initiation complexes that included SD-aSD interactions than those that did not. This indicated that the SD-aSD interaction contributes significantly to the stability of the complex. From rupture force measurements, they found that formation of the first peptide bond destabilizes the SD-aSD interaction, which loosens the grip of the ribosome on the mRNA at the initiation site. They proposed that this destabilization is an important step to facilitate the ribosome moving into elongation.

2.5 Elongation

Elongation is the step after initiation in which the ribosome reads each individual codon on the mRNA. These codons specify the different amino acids that form the polypeptide chain. The ribosome contains three sites: E, P, and A. The E-site is where uncharged-tRNAs exit the ribosome. Next to the E-site is the P-site, where the amino acid attached to the aa-tRNA is added to the polypeptide chain. The codon in the A-site determines the next aa-tRNA to be recruited to the translation complex. In total, elongation
consists of three major steps: aa-tRNA binding, transpeptidation, and translocation as shown in Figure 2.10.

The first step, aa-tRNA binding, is driven by the codon-anticodon interaction between the aa-tRNA and the codon located at the A-site. For this to occur, the ribosome pauses until an aa-tRNA containing the correct cognate nucleotide triplet, dubbed the anticodon, enters the A-site. However, this interaction only loosely follows the canonical Watson-Crick base-pairing, meaning that a single aa-tRNA can recognize more than one codon. This interaction is also aided by elongation factor Tu (EF-Tu), which binds to the aa-tRNA and GTP to form a ternary complex. When this complex interacts with the ribosome, GTP is hydrolyzed to GDP. This lowers the affinity between EF-Tu and the ribosome, which releases EF-Tu and leaves the aa-tRNA at the A-site. This aa-tRNA binding process is shown in Figure 2.10. The EF-Tu / GTP complex with aa-tRNA also serves in proofreading that the correct amino acid is being added.

The second major step of elongation is transpeptidation, or the formation of the peptide bond. During this step, the aa-tRNA at the A-site interacts with the peptidyl-tRNA located at the P-site. The amino group on the aa-tRNA acts as a nucleophile and “attacks” the carbonyl group in the ester bond between the peptide residue and tRNA moiety at the P-site (Figure 2.9a). This results in the transfer of the peptide bond from the peptidyl-tRNA to the aa-tRNA (Figure 2.9b). The free energy gained from transpeptidation helps drive translocation, the next step of elongation.
The third and final step of elongation is translocation. Here, the deacylated-tRNA in the P-site is released through the E-site, and the newly formed peptidyl-tRNA in the A-site shifts to the P-site. This leaves the A-site vacant and ready for the next aa-tRNA to bind. This translocation process is catalyzed by elongation factor G (EF-G). Similar to EF-Tu, EF-G forms a complex with GTP and binds to the ribosome. Upon translocation, GTP is hydrolyzed to GDP, and the affinity between EF-G and the ribosome is decreased, thereby releasing EF-G from the ribosome. In short, translocation consists of the ribosome moving in frame downstream from the initiation codon in controlled movements that are a distance of three bases (one codon) long.
Figure 2.10. Steps in translation elongation. The three steps of translation elongation, including codon recognition, peptide bond formation, and translocation are depicted here in their cyclical, repeating mechanism.

New technologies like cryo-electron microscopy, single molecule real time fluorescence resonance energy transfer (FRET), x-ray crystallography, and the optical tweezer assay have illuminated the structure composition and ribosome dynamics during elongation. New translation models have been proposed based on the observations from the aforementioned technology [3]. The precise mechanics during elongation are still up for debate. However, a general consensus model has been developed.

The consensus model proposes that the ribosome “ratchets” to translocate the mRNA. This ratcheting is caused from multitudes of ribosome subunit rotation and conformational changes. Two primary rotations were identified as the 30S subunit rotation relative to the 50S as well as the 30S head domain rotation. Rotation of the 30S relative to the 50S puts the ribosomes in two different conformations. The first conformation is the classic (non-rotated)
state. In this state the three tRNA-binding sites on both the 30S and 50S are aligned directly on top of each other. The second conformation is the hybrid (rotated) state in which the 30S is rotated approximately 8 to 9 degrees counterclockwise relative to the 50S [1,3,13].

The ribosome begins the elongation cycle in the classic state. In this state, the ribosome awaits the delivery of the next aa-tRNA. After the aa-tRNA delivery by EF-Tu, transpeptidation begins. Transfer of the polypeptide chain from the peptidyl-tRNA to the aa-tRNA triggers the rotation of the 30S relative to the 50S and L1 stalk fluctuations. Now the ribosome is in the hybrid state. During the hybrid state, the acceptor end of the peptidyl-tRNA fluctuates between the E-site and P-site while the aminoacyl-tRNA fluctuates between the P-site and A-site on the 50S. Rotation of the 30S head domain and binding of the EF-G:GTP complex stabilizes the hybrid state and minimizes tRNA fluctuation on the acceptor end. The acceptor end of the peptidyl-tRNA is now stabilized at the E-site of the 50S while the anti-codon end is located at the P-site on the 30S. The same motion occurs with the aminoacyl-tRNA for the P and A-sites. Hydrolysis of GTP by EF-G “unlocks” mRNA-ribosome binding and translocates the ribosome. After translocation, the ribosome changes confirmation back to the classic state and awaits delivery of the next aa-tRNA. The cycle is then repeated until a stop codon is recognized leading to termination of elongation. See Figure 2.11 for the overview of recently proposed translation elongation model.
Vu et al. [19] proposed the “ribosome spring model” (see Chapter 4.2), which is an addition to the current translation elongation model. The ribosome spring model was proposed based on the observation of a periodic binding pattern between the 16S rRNA 3’ terminal end and mRNA during in-frame translation. The observation that this energetic binding pattern drastically changed during a +1 programmed frameshift event also supports the “spring” model [11]. Using a fundamental law of physics, the negative change in energy with respect to the change in distance equals the force:

\[
\Delta \text{energy} = -\text{force} \cdot \Delta \text{distance} \quad (2.1)
\]

\[
\text{force} = -\Delta \text{energy}/\Delta \text{distance} \quad (2.2)
\]
Vu et al. [19] proposed that the binding between the 16S rRNA 3’ terminal end and the SD-like sequence in the +1 programmed frameshift acts like a spring that exerts a force on a ribosome relative to the mRNA to displace the ribosome towards the +1 reading frame. This spring model was applied to normal in-frame translation that resulted in the ribosome being displaced by fractions of a nucleotide at every codon by this energetic spring as it waits for the delivery of the next aa-tRNA. The displacement can result in small misalignments between the A-site and the zero reading frame. The author of this dissertation proposes that the periodic 16S rRNA 3’ terminal end binding to the mRNA acts as a reading frame maintenance mechanism and believes this binding occurs right after translocation when the mRNA is “unlocked” [3,13] from the ribosome but before delivery of the aa-tRNA. Tinoco et al. [13] also proposed the same stage in elongation for frameshifting events.

2.6 Termination

Termination of translation involves the recognition of one of three termination codons: UAA, UAG, or UGA [9,17,36]. It is important to note that termination codons are often found in the other reading frames, meaning that accidental frameshifting during translocation will often result in termination of the faulty protein. Termination codons do not code for any amino acid but instead induce the binding of release factors (RF1, RF2, RF3). RF1 is specific to termination codons UAA and UAG; whereas, RF2 is specific to UAA and UGA. RF3 acts in a GTP-dependent fashion similar to EF-Tu and EF-G. These release factors simulate the aa-tRNA interaction at the A-site and break the peptidyl-tRNA ester bond rather than the usual transpeptidation, detaching the polypeptide chain from the P-site [17,36]. GTP is then hydrolyzed to GDP, and the release factors are released from the
ribosome. The deacylated-tRNA is then released, and the ribosome separates into the 50S and 30S subunits, terminating the process of translation. This is shown schematically in Figure 2.12.

**Figure 2.12. Translation termination.** The termination of translation and binding of release factors are depicted as the mRNA is released from the dissociating ribosome. Image reproduced with modifications from Wikimedia Commons under Creative Commons Attribution 3.0 Unported license [37].

### 2.7 Reading Frame and Programmed Frameshifts

**Reading Frame**

Reading frames are a perspective for analyzing a nucleotide sequence based on the three base compositions of codons in protein coding regions of mRNA. The addition or elimination of a single nucleotide to a coding sequence would shift the reading frame. A change in reading frame can drastically alter the primary sequence of a polypeptide and cause early termination. Since the genetic code calls for translation of three nucleotides to one amino acid, from a fixed starting position (as set by the initiation codon), there exist three reading frames from which the codons can be read, as depicted in Figure 2.13. The reading frame defined by the start codon is called the 0 reading frame. The frame that would be one base downstream from the 0 frame is called the +1 reading frame, and the frame with a one base shift upstream is called the -1 reading frame. Most coding regions in genomes code for
one protein in one reading frame, but exceptions such as overlapping (internal) genes [24,25,26] and programmed frameshifts have been observed [8,13,21,22].

![Figure 2.13. Reading frames.](image)

**Programmed Frameshifts**

Frameshifting is a mechanism in which the ribosome shifts out of the 0 frame and into one of the two other frames relative to the start codon. This produces a different amino acid sequence compared to the original reading frame. There are two common types of frameshifts: +1 and -1 frameshifts. These events rarely occur in nature; however, some genes have evolved to contain embedded frameshifts. These are called “programmed” frameshifts. An example of a +1 programmed frameshift is prfB in *E. coli* [8,21]. The phenomenon of programmed frameshifting has been studied intensively because of its potential to reveal the mechanisms that maintain the appropriate reading frame of the ribosome during translation. The model gene prfB in *E. coli* has been studied in detail to determine the sequence features that are associated with a specific shift in reading frame.
Utilizing extensive mutant analysis and modifications of the 16S rRNA aSD and mRNA, a model for frameshifting was developed [8,21]. In the descriptions by Farabaugh et al. [8] and Weiss et al. [21], frameshifting requires three components to be present in the mRNA sequence for a frameshift to occur. The required motifs are: a) a Shine-Dalgarno-like sequence upstream of the codon where the frameshift occurs, b) a “slippery” codon, and c) a stop codon just downstream of the slippery sequence region. More complex features such as RNA structures termed pseudoknots have been described at other programmed frameshift sites [20,22].

More recent models of frameshifting by Vu et al. [19] and Tinoco et al. [13] suggest that the binding between the 16S rRNA aSD and internal SD sequence acts like a spring to move the ribosome relative to the mRNA. This movement results in a frameshift where the ribosome will then pick up the aa-tRNA corresponding to the new reading frame. The author of this dissertation proposes that the “slippery” codon serves to interact with the 16S rRNA 3’ terminal end after the frameshift to further maintain the new reading frame. The author also proposes that the pseudoknot and hairpin in a -1 programmed frameshift act to slow the ribosome and exert excess force that moves the ribosome to the -1 reading frame; Tinoco et al. [13] also proposed similar functionalities for the hairpin.
2.8 References


CHAPTER 3

Determinants of Translation Efficiency

3.1 Introduction

The central dogma of molecular biology describes the transfer of genetic information from genes to proteins. Protein synthesis occurs during the process of translation in which polypeptide chains are synthesized from mRNA using ribosomes. Translation mechanisms are highly complex and regulated to control protein production to perform various functions within living organisms. Although the basic biochemical principles of protein synthesis have been established, the determinants of protein yield during translation remain elusive, and heterologous protein production further complicates these factors.

In the past ten years, scientific advances have revealed some of the determinants of protein yield that were not previously considered by the early predictive indices. In this review, the author provides a comprehensive summary of the recent scientific advances elucidating the factors influencing protein yield during initiation and elongation. The author also provides extensive overviews of the most substantial discoveries that unravel the workings of translation. Techniques and models relevant to translation that are used to optimize protein yield are also discussed. Each of these factors plays a crucial role in regulating translation and thus modeling protein yield.
3.2 Determinants of Protein Yield during Initiation

Salis et al. [1] provide an extensive analysis of translation initiation in their 2009 publication. They contend that the four molecular interactions that determine initiation rates are (1) 16S rRNA to ribosome binding site (RBS) hybridization, (2) 16S rRNA binding site to start codon spacing, (3) RNA secondary structures that occlude either the 16S rRNA binding site or the standby site, and (4) the binding of fMet-tRNA to the start codon. These first three determinants are integral to modeling translation initiation and are discussed here. The fourth determinant, involving binding of the first tRNA to the start codon, stabilizes the initiation complex and aids in the transition to elongation.

3.2.1 Shine-Dalgarno Sequence

In 1974, Shine and Dalgarno [2] proposed that a sequence of nucleotides located upstream from the start codon on mRNA attracts ribosomes to bind to the mRNA and initiate protein synthesis. This sequence became known as the Shine-Dalgarno (SD) sequence. The most frequent nucleotides that appear in this initiation site (known as the consensus sequence), vary slightly across different organisms. The consensus sequence can be used for identifying ribosome binding sites (RBS). In *Escherichia coli*, this consensus sequence is 5’-AGGAGG-3’. The Shine-Dalgarno sequence was found to bind to the anti-Shine-Dalgarno (aSD) sequence (3’-UCCUCC-5’) on the 16S ribosomal-RNA 3’ terminal end (see Ribosomes in Chapter 2.3). An SD sequence with complementarity to the aSD has a greater affinity for the 16S ribosomal-RNA tail and thus recruits more ribosomes to initiate translation and increase protein yield [3].
3.2.2 Shine-Dalgarno to Start Codon Spacing

In 1994, Chen et al. [4] determined that the optimal spacing between the Shine-Dalgarno sequence and the start codon on a transcribed mRNA was 5 nucleotides in *Escherichia coli*. Chen et al. [4] determined this optimal spacing through a series of laboratory experiments using the chloramphenicol acetyltransferase (*cat*) gene and varying the length of spacing between the SD sequence and the start codon. This distance of 5 nucleotides allows for the 16S rRNA anti-Shine-Dalgarno to bind to the Shine-Dalgarno sequence such that the P-site of the ribosome aligns perfectly with the start codon. If the distance is substantially shorter or longer, the ribosome will not be able to initiate translation by binding fMet-tRNA to the start codon. Chen et al. [4] showed that altering this optimal spacing reduces protein yield presumably by altering the translation initiation complex, a complex composed of 30S ribosomal subunits, initiation factors, and fMet-tRNA.

Shultzaberger et al. [5] also looked at the optimal spacing between the SD sequence and the start codon using information theory and Gaussian distribution. They reported that ribosomes can initiate translation with various spacing distances but prefer a consensus spacing of 6-8 nucleotides. Malys [6] also looked at the spacing for phage genes and found that the consensus “aligned spacing” was around 7 nucleotides for “early” phage genes; this corresponded to 5 nucleotides between the start codon and the aSD of the 16S rRNA, as shown in Figure 3.1.
Figure 3.1. Spacing distance between Shine-Dalgarno sequence and start codon. This spacing has an optimal distance of five nucleotides (depicted as five ‘X’) in *E. coli*, which is integral to regulating ribosome displacement during elongation. Image re-drawn from [6].

Both Shultzaberger et al. [5] and Salis et al. [1] proposed that the ribosome acted like a “rigid spring” during initiation. The author of this dissertation proposes that the binding between the 16S rRNA aSD and mRNA SD sequence acts like a spring during initiation when IF1 and IF3 guides the initiator-tRNA:start codon into the P-site. The optimal spacing between the SD sequence and the start codon corresponds to a “relaxed spring” on the ribosome. This ribosome spring can compress or extend to yield various spacing distance as observed by Shultzaberger et al. [5] and Malys [6]. Deviations from the “relaxed spring” results in a lower initiation rate, and thus protein yield decreases. The author of this dissertation believes that a relaxed spring corresponds to a spacing distance of 5 nucleotides (see Figure 3.1) as described in Vu et al.’s ribosome spring model [7] as well as the works of Chen et al. [4] and Malys [6].

3.2.3 GC Content and Secondary Structure during Initiation

In 1990, de Smit et al. [8] introduced the effects of mRNA secondary structure on translation initiation. They analyzed the effects of a hairpin structure on protein yields and
found a correlation between the stability of the secondary structure and translation efficiency. Their results indicated that initiation depends on the spontaneous unfolding of the initiation region. In 2006, Studer et al. [9] reinforced this correlation. They assayed different mRNA sequences that formed secondary structures with varying stabilities and correlated mRNA secondary structure with association and dissociation rate constants. Heavily folded regions require extensive unfolding by the bacterial translation initiation complex and can interfere with initiation.

In 2009, Kudla et al. [10] also reinforced this idea by showing that yield of green fluorescent protein was not based on codon bias but rather the 5’ secondary structure of the encoded mRNA. The 5’ secondary structure or folding of the mRNA appears to inhibit translation initiation by affecting ribosome binding. Kudla et al. [10] studied the effects of random silent mutations in the green fluorescent protein (GFP) gene. After introducing random silent mutations across 226 of the 240 GFP codons to create a library of GFP variable sequences, they expressed the genes in *Escherichia coli* using a T7 promoter. Expression was quantified using spectrofluorometry, and fluorescence levels varied 250-fold across the synthetic library and were highly reproducible. This variation was supported by western blot and Coomassie staining.

Kudla et al. [10] then compared codon usage to their measured fluorescence and found that neither common measure of codon bias, codon adaptation index (CAI), nor the frequency of optimal codons correlated to fluorescence levels with correlation coefficients of $r=0.14$ and $r=0.11$ respectively. They also noted that the first 42 bases in the GFP gene at the 5’ terminus, which is typically considered important for expression, was not significantly
correlated with fluorescence intensity with $r=0.1$. Likewise, neither the number of rare codons in a sequence nor the number of consecutive rare codons showed a strong correlation to fluorescence with correlation coefficients of $r=-0.02$ and $r=-0.14$ respectively.

Statistical analysis of Kudla et al.’s data [10] suggested a greater influence of local sequence patterns compared to global codon bias in expression levels, which is consistent with the emerging idea that secondary structure around the RBSs may shape expression levels. To explore this idea, Kudla et al. [10] calculated the predicted minimum free energy for the secondary structure of the entire mRNA and correlated it with fluorescence. This also yielded no strong correlation ($r=0.16$), but the folding energy of the first third of the mRNA was strongly correlated with fluorescence ($r=0.60$). A moving window analysis identified the nucleotide region from -4 to +37 as explaining 44% of the variation based on predicted folding energy across the GFP library ($r=0.66$). The same folding energies explained 59% of fluorescence variation when the constructs were expressed in a pBAD vector.

The strong correlation between mRNA folding and fluorescence suggests that tightly folded messages obstruct translation initiation to reduce protein synthesis. Kudla et al. [10] found the mRNA folding energy near the start codon (-4 to +37) explained nearly 10 times as much variation in expression levels as any other predictor variable, including global GC content, CAI, number of rare codon sites or consecutive pairs, length of the longest rare-codon stretch, and a number of other predictors. To test their conclusion, Kudla et al. [10] added a 28-codon tag with low folding energy to the 5’ terminus of 72 GFP constructs and expressed them. Each tagged construct produced high yields, including those previously
expressed in low yields. Kudla et al. [10] ultimately concluded that *Escherichia coli* genes have undergone selection for weak 5’ secondary structures in the translation initiation region.

In 2010, Allert et al. [11] revealed the relationship between increased protein expression and low GC content in the initiation region of the encoded mRNA. They quantitatively describe the relations between protein expression levels and codon identity, mRNA secondary structure, and nucleotide composition within open reading frames. They sequenced the open reading frames of 816 bacterial genomes to show regional trends based on the 5’ region (first 35 bases), the middle region, and the 3’ region (last 35 bases). They optimized 285 new genes based on codon usage, AU/GU composition, and secondary structure to assess their effects on protein yield using *Escherichia coli* extracts *in vitro*. Their results revealed that high protein expression levels are dependent on high AU content (compared to high GC content, high AU content leads to less stable RNA structures) and low secondary structure in the first 35 bases. They also showed that optimizing AU content in the last 35 bases and using high frequency codons throughout the gene is less effective at increasing yield. Allert et al. [11] contend that codon bias contributes to a lesser extent, and they reinforce the idea that secondary structure formation in mRNA translation initiation region leads to decreased protein production.

### 3.2.4 Optimizing Ribosome Binding Sites

Salis et al. [1,38] developed a systematic method of optimizing ribosome binding sites (RBSs) to control translation initiation and protein expression levels. They call this method the “Ribosome Binding Site Calculator” and propose a biophysical model of translation initiation based on molecular free energy changes. They contend that initiation is
the rate-limiting step of translation and is determined by molecular interactions including the
determined by molecular interactions including the
are: (1) 16S rRNA to mRNA RBS hybridization, (2) the binding of fMet-tRNA to the start codon, (3) mRNA SD to start codon spacing, and (4) RNA secondary structures that occlude either the 16S rRNA binding site or start codon. They define translation
initiation rate ($r$) as:

$$r \propto e^{-\beta \Delta G_{tot}}$$ (3.1)

where $\Delta G_{tot}$ is the change in Gibb’s free energy between the two energy states and $\beta$ is the Boltzmann constant that converts thermodynamic free energies to temperature differences.

For an mRNA sequence surrounding a start codon, $\Delta G_{tot}$ is predicted according to:

$$\Delta G_{tot} = \left( \Delta G_{mRNA:rRNA} + \Delta G_{start} + \Delta G_{spacing} - \Delta G_{standby} \right) - \Delta G_{mRNA}$$ (3.2)

where

1) a reference state is a fully unfolded subsequence with $\Delta G_{tot} = 0$

2) $\Delta G_{mRNA:rRNA}$ is the energy released when the last nine nucleotides of the *E. coli* 16S rRNA hybridizes to the mRNA SD

3) $\Delta G_{start}$ is the energy released when the start codon hybridizes to the initiating fMet-tRNA anticodon loop (3’-UAC-5’)

4) $\Delta G_{spacing}$ is the free energy penalty caused by a non-optimal spacing between the SD and the start codon

5) $\Delta G_{mRNA}$ is the work required to unfold the mRNA subsequence when it folds to its most stable secondary structure called the minimum free energy structure
6) $\Delta G_{\text{standby}}$ is the work required to unfold any secondary structures sequestering the standby site after the 30S complex binds. The free energy terms are not orthogonal

Salis’s RBS calculator [38] computes the terms in **Equation (3.2)** using the NUPACK [53] or the Mfold 3.0 RNA [52] free energy parameters. **Equation (3.2)** is evaluated for each AUG or GUG start codon in an mRNA transcript and only considers the subsequence of 35 nucleotides before and 35 after the start codon, including the RBS and part of the protein coding sequence. This model allows for reverse engineering, which predicts the initiation rate of an RBS sequence upstream from a protein coding sequence. To prove this, they obtained 28 RBS sequences and used flow cytometry to measure steady-state protein expression for each. Log protein fluorescence and $\Delta G_{\text{tot}}$ correlated according to **Equation (3.1)** with r-squared of 0.54. Their model worked for initiation rate control over a wide dynamic range of 16S rRNA binding strength. Salis et al.’s model [38] also allows for forward engineering, which uses the model to output a new nucleotide sequence of synthetic RBS to drive protein translation at a specified rate. The algorithm begins with a random mRNA sequence upstream of the protein coding region. New sequences are generated by replacing, inserting, or deleting nucleotides, and $\Delta G_{\text{tot}}$ is continuously computed and compared to a target $\Delta G_{\text{tot}}$. When $\Delta G_{\text{tot}}$ is within a given threshold of the target, the sequence is complete, and there are often multiple solutions possible. The model was tested with 29 synthetic RBS sequences over a range of $\Delta G_{\text{tot}}$ with great success [1].
3.3 Determinants of Protein Yield during Elongation

The development of a model for translation elongation dates back several decades. More recently, various predictive indices and optimization methods for translation have emerged [12,17,22,40,45,46]. Predictive indices and optimization methods have largely focused on the elongation step, during which the ribosome translocates down the mRNA while tRNAs transfer amino acids to the growing polypeptide chain.

3.3.1 Codon Bias and Codon Bias Optimization Algorithms

Codon Bias

During protein translation in *E. coli*, 61 different codons code for only 20 amino acids. In most cases, several codons can code for the same amino acid. This occurrence is referred to as synonymous codon usage. Some organisms prefer to use one or two codons over the other synonymous codons, a preference known as codon bias [12,39,40,41].

Codon bias has been incorporated into predictive indices such as the widely-used Codon Adaptation Index (CAI) developed by Sharp et al. in 1987 [12]. Highly-expressed genes serve as the reference genes when computing CAI. Each of the 61 codons is given a “weight”, a ratio of usage when compared to other synonymous codons in the reference genes. The CAI of a gene is the geometric mean of the weight associated to each codon over the length of the gene sequence. Genes with codon usage similar to that of highly-expressed genes score higher, with CAI ranging from 0 to 1. For any given gene, CAI can be used as a measure of codon bias relative to the total genomic codon bias or the codon bias of highly-expressed genes. CAI has become the standard for predicting and optimizing protein yield.
However, one problem with the CAI is the need for a reference set of highly-expressed genes of an organism. This is not an issue for *Escherichia coli*, *Saccharomyces cerevisiae*, or other model organisms, but it might be difficult to determine the most commonly used codons for other less well-characterized species [13].

It is proposed that protein yield can be increased by modifying the codon bias of a heterologous gene towards the codon bias of highly expressed genes of the production organism. Early experiments [16,40,41] established a correlation between codon bias and protein expression levels. It is the general belief that using the codon bias of highly expressed genes will result in using the most abundant tRNAs during translation [12,40,41,49,50,60]. However, codon bias optimization has had mixed results in practice, with particularly inconsistent outcomes in heterologous protein production [10,11,13,14,22,40,42,56,63]; codon bias optimization has also been shown to worsen protein expression [40,42,56,57,63]. For example, out of 94 case studies published by Maertens et al. [56], 19 cases showed negative improvement, 21 cases showed no expression level, and 3 cases showed no improvement after codon bias optimization. According to the results published by Maertens et al. [56], codon bias optimization has a 46% failure rate (43 out of 94 cases failed). Another study of codon bias optimization was published by Burgess-Brown et al. [63] showed that out of 30 cases, 11 cases showed no improvement, and one case showed negative improvement after codon bias optimization. The results published by Burgess-Brown et al. [63] demonstrated that codon bias optimization has a 40% failure rate (12 out of 30 cases failed).
In 2010, Tuller et al. [15] analyzed the APEX *E. coli* proteomics database to verify Kudla’s work [10] on the free energy of secondary structures and their effect on initiation. Tuller et al. [15] examined the dual role of codon bias and mRNA folding energy at initiation in determining translational efficiency across the entire *Escherichia coli* and *Saccharomyces cerevisiae* genomes. They examined the mean folding energy in each 40-nucleotide window in each gene and found that the first window was significantly higher energy than the other windows in the gene. In *Escherichia coli*, the authors compared mRNA folding energy of the first 40 nucleotides with translational efficiency and found a subtle correlation. They also compared translational efficiency with other 40-nucleotide windows in each gene and found no significant correlation. They did find a significant correlation between translation efficiency and codon bias, which held true for both local and global translational efficiency. Tuller et al. [15] concluded that codon bias was the main determining factor for translation efficiency, but folding energy of the first 40 nucleotides also plays a key role. The authors observed that folding energy modulates the association between codon bias and translation efficiency. Genes that had low folding energy in the first 40 nucleotides (higher stability of secondary structure) showed low correlation to codon bias. However, genes with high folding energy (lower stability of secondary structure) showed higher correlation to codon bias. Genes with lower folding energy obstruct translation initiation and slow translation initiation rate [1,38]; therefore, yield from these genes correlate less with codon bias because codon bias only measures yield from elongation.

The author of this dissertation believes that codon bias correlates with protein yield when examining *native protein abundance*. The author also believes that mRNA secondary
structure folding energy at initiation does not play a major role when examining native genes and initiation sites. However, heterologous genes are not optimized for E. coli’s RBS; therefore, the 5’ end of exogenous genes needs to be optimized for initiation in E. coli. The author proposes that organisms have evolutionarily adapted their own genes to match the 16S rRNA 3’ terminal end binding patterns and optimize tRNA usage. Therefore, this adaptation appears as “codon bias”, an organism’s preference to use a subset of synonymous codons. Expression of heterologous genes changes specific tRNA abundance; therefore, codon bias is not a good indicator of protein yield in heterologous expression [22].

**Codon Bias Optimization Algorithms**

There are two codon bias optimization algorithms, both of which use a set of genes as reference to determine codon bias. This set of genes could be highly expressed genes [46] or annotated genomic coding sequences [47,48]. The first algorithm, dubbed “one amino acid, one codon,” uses one codon out of the set of other synonymous codons to code for an amino acid [45,46]. For example, leucine has potential 6 codons, but since codon CUG is used most often as compared to the other synonymous codons, the algorithm always uses CUG to code for leucine. This optimization applies to all other amino acids as well. This algorithm is deterministic; i.e., the resulting optimized sequence is always the same given the same input sequence.

The second algorithm, dubbed “guided random, codon table matching” [40], matches the codon usage table of the gene to the codon usage table of the production organism’s highly-expressed genes [45] or genome [47,48]. As the name suggest, this algorithm is random or stochastic; the resulting optimized sequence is different with each optimization.
attempt given the same input. A minimal codon frequency “threshold” is set for each codon; this threshold is usually between 5% and 10%. Codons with frequencies lower than that threshold are set to zero so that the chances of using rare codons are completely eliminated. Genes are then designed using a Monte Carlo algorithm that matches the codon usage table of the genes to the referenced codon usage table. Synonymous codons are “picked” based on probabilities derived from the codon frequencies.

The guided random method is superior to the “one amino acid, one codon” method because it spreads the tRNA usage among different tRNA species [62]; the reasoning is that aa-tRNA abundance is a limiting factor in heterologous expression (see Chapter 3.3.2). However, the guided random method does not take into account the “ordering” of codons, and it may lead to a displaced ribosome (misaligned ribosome) due to secondary effects from the 16S rRNA 3’ terminal end binding with the mRNA (see Chapter 3.3.3 and Chapter 3.3.5). The author contends that the effects of misalignments occur more when expressing long genes.

3.3.2 tRNA Availability and Recharge Rates

Another predictive index for protein yield, the tRNA adaptation index (tAI), incorporates tRNA usage based on codon usage throughout a gene. The rationale behind tAI is that tRNA usage and availability can be the limiting factor when producing proteins [13,16]. The tAI was developed in 2004 by dos Reis et al. [17] and provides a prediction of protein yield based on the tRNA gene copy number in established genomes [17]. tAI uses tRNA gene copy number as an approximation of tRNA abundance. This principle is
supported by various other sources (Tuller et al. [15], Dong et al. [18], Percudani et al. [19], Kanaya et al. [20], and Duret et al. [61]). Codons coding for the most abundant tRNAs are presumed to correspond with higher protein expression [18,49,50]. The tAI index is limited because it relies on “static” tRNA gene copy number rather than direct aminoacyl-tRNA abundance, which changes under different growth conditions. A more robust tAI should consider a codon’s tRNA workload “sensitivity” as explained by Elf et al. [21].

In 2009, Welch et al. [22] showed that protein yield was not caused by codon bias but rather by a change in tRNA “recharge rates” because of amino acid starvation due to overexpression. They state that although many gene optimization algorithms base their designs on the codon bias within the production organism, recent experiments have shown that translational efficiency is compounded by many complex factors. In their experiments, the authors created 81 variants of two different genes to assess the factors that control gene expression in *Escherichia coli*. The two genes selected were DNA polymerase from Bacillus phage phi29 and a synthetic single-chain variable fragment (scFv) developed by OncoMed in Redwood City, CA. These genes were chosen because they have different amino acid composition and also have significant commercial value. The genes were synthesized and expressed using a T7 promoter in a pLysS vector in *E. coli*. Expression levels were quantified using SDS-PAGE with Sypro Ruby (Pierce) stain and fluorescence imaging. All initial variants were created using a Monte Carlo repeated random sampling algorithm.

Welch et al. [22] also created a number of chimera genes to determine the local effects of gene segments using high-expression and low-expression variants. After dividing the genes into 5’, middle, and 3’ segments and recombining different chimeras, the protein
expression was measured. The substitution of high expression fragments into low expression genes increased protein yield; whereas, low expression fragments decreased yield when added to higher expression genes. Expression was not correlated to any tested motifs, including 5’ or internal mRNA secondary structures, GC bias in the first 15 codons, internal Shine-Dalgarno-like motifs, or transcriptional terminator motifs.

Using a genetic algorithm and a Partial Least Squares (PLS) regression, Welch et al. [22] created unique methods to predict expression based on codon usage. These models were applied to the expression datasets and showed a high correlation of expression to codon usage. The models and expression data were compared to the codon bias index, CAI. Welch et al.’s [22] algorithm revealed that highly expressed genes from their experiments frequently used codons that are labeled as uncommon in *Escherichia coli*. They found no correlation between the CAI and expression levels. Welch et al. [22] proposed that during overexpression, the tRNA recharge rates change due to amino acid starvation [21,23,54]. This changes the availability of synonymous tRNAs; thus, minor and less common tRNAs can become more abundant than major tRNAs. Welch et al. [22] proposed that using codons for minor and less common tRNA rather than using the codons in highly expressed genes will improve protein yield.

During heterologous protein production, the cell commits most of its resources to producing ribosomes, RNA polymerases, elongation factors, etc., which in turn produce the heterologous protein. Translation of transcription and translation proteins, which are coded by highly expressed genes, use the major tRNAs. Optimizing heterologous genes using the codon bias of highly expressed genes would result in the heterologous gene competing for
the same tRNA pool. The author of this dissertation contends that the aforementioned competition would deplete the major aminoacyl-tRNAs at the detriment of protein yield. This tRNA workload sensitivity phenomenon was explained by Elf et al. [21] and experimentally verified by Dittmar et al. [23]. The author of this dissertation proposes that optimizing heterologous genes should consider distributing tRNA usage between minor and major tRNAs based on workload sensitivity as proposed by Elf et al. [21]. However, optimizing heterologous protein synthesis should consider the distribution of tRNA workload and the energetics of 16S rRNA tail:mRNA binding as explained by Vu et al. [7] and explained in Chapter 3.3.3 and Chapter 3.3.5.

3.3.3 16S rRNA 3’ Terminal End Binding to mRNA

In 2012, Li et al. [24] used a ribosome profiling strategy [25,26,27] to measure translation rates of 2,257 genes in *Escherichia coli* and 1,580 genes in *Bacillus subtilis*. Their results suggested that ribosome transit time across an mRNA is proportional to ribosome occupancy at each position based on: (1) negligible internal initiation and early termination associated with ribosome pause sites, (2) unimpaired ribosomes, and (3) minimal variability during the technical conversion of RNA fragments to sequenced DNA. Li et al. [24] found little correlation between the average ribosome codon occupancy and existing measurements of tRNA abundance [18]. The authors cross-correlated intragenic ribosome occupancy with trinucleotide sequences on the mRNA independent of reading frames. They found strong correlation for six trinucleotide sequences that resembled the Shine-Dalgarno (SD) [2] sequences. This correlation was highest at positions 8-11 nucleotides upstream in the mRNA relative to the ribosomal A-site and was found across two bacteria of divergent phyla but not
in *Saccharomyces cerevisiae* whose ribosomes do not contain an anti-SD sequence. Li et al. found strong correlations between: (1) hybridization free energy of mRNA nucleotides to the 16S rRNA aSD sequence and downstream ribosome occupancy, (2) SD-like sequences on transcripts and ribosomal pauses, and (3) synonymous mutations and reduced ribosome occupancy, suggesting a causal relationship between SD-like sequences and ribosomal pausing [24].

Li et al. [24] also showed that the excess ribosome footprint density at internal SD-like sequences was due to ribosome pausing during elongation rather than from internal initiation. They used an orthogonal SD sequence that binds exclusively to a mutated ribosome to translate a *lacZ* message to confirm internal pausing; the mutated ribosome contained a different anti-Shine-Dalgarno sequence. A strong pause is defined as ribosomal occupancy ten standard deviations above the mean. The authors found that 70% of strong pauses are caused by internal SD sites. They relate translational pausing to decreased bacterial growth and suggested that internal SD-like sequences are generally avoided in *Escherichia coli* and *Bacillus subtilis*. Their results explain the selection against two consecutive codons that resemble SD sequences in native genes; however, Li et al. [24] suggested a functional purpose for ribosome pausing caused by SD-like sequences, citing programmed translational frameshift gene *prfB* [28,32], co-translational peptide folding [27,35,54], and transcriptional regulation in bacteria [29,30,31].

In summary, Li et al.’s [24] ribosome profiling data suggests that the 16S rRNA anti-Shine-Dalgarno continuously interact with the mRNA during translation elongation. They found no correlation with ribosome translation elongation rate and tRNA abundance or rare
codons. Their data suggests that ribosome pausing is due to hybridization between 16S rRNA anti-Shine-Dalgarno and SD-like sequence on the mRNA. This corroborates the earlier observations of Weiss et al. [32] and Larsen et al. [33] on the role of the 16S aSD sequence during translation elongation. The author conjectures that the ribosome is physically displaced by the hybridization between 16S rRNA anti-Shine-Dalgarno and SD-like sequence on the mRNA which acts like a spring and causes ribosome pausing [7].

3.3.4 GC Content and Secondary Structure in Open Readings Frame

Heterologous protein expression was thought to be more effective if the two species considered - the production organism and the species from which the gene is obtained - had similar GC content. In 2004, Chen et al. [34] analyzed the genomes of 100 eubacterial and archaeal organisms and found that differences in genome-wide codon biases were due to GC content and nucleotide bias. In their 2010 review of codon bias, Plotkin and Kudla [14] contend that “the strongest single determinant of codon-usage variation across species is genomic GC content.”

The molecular mechanism through which GC content affects translation varies. For instance, mRNA sequences with high GC content can form secondary structures that inhibit initiation as previously discussed. In contrast, mRNA transcripts with low GC content are less likely to form secondary structures but still show decreased expression, such as in the expression of Plasmodium falciparum genes in Escherichia coli [14]. Furthermore, it is estimated that over 40% of human genes would express poorly in Escherichia coli due to differences in mRNA folding caused by changes in GC content (~40% GC in humans compared to ~50% in the bacterium) [14]. It was previously believed that high GC content
produced secondary structure in the ORF that impedes translation elongation; however, recent experiments suggest that the ribosome act as a helicase and “melts” the secondary structure during elongation [35,43,44]. In addition, experiments of Welch et al. [22], Allert et al. [11], Kudla et al. [10], and Sørensen et al. [16] showed little correlation between GC content/mRNA secondary structure and heterologous protein yields.

It is known that the nucleotide composition of the 16S rRNA exposed tail varies across bacterial species [58,59]. It appears that variation in genomic GC content across bacterial species correlates with the variation in nucleotide composition of the 16S rRNA exposed tail [59]. As calculated from the interaction between 16S rRNA exposed tail and mRNA, different bacterial species favor different periodic “energetic” binding patterns, to translate their genes efficiently. This interaction might contribute to different GC contents between bacterial species.

### 3.3.5 Ribosome Displacement and Ribosome Spring Model

In 2014, Vu et al. [7] revealed yet another determinant of protein yield based on second order free energy effects from the binding between the anti-Shine-Dalgarno sequence of the 3’ terminal 16S rRNA tail and the mRNA transcript. Vu et al. [7] described a new factor dubbed ribosome displacement that arises from these second-order free energy effects. Ribosome displacement is the misalignment between the A-site and normal “zero” reading frame and is proposed to be “cumulative” during elongation; i.e., misalignments are not reset after translocation. This paradigm suggests that ribosome has “memory” during elongation.
Vu et al. [7] used the well-established programmed frameshifting genes *prfB* and *dnaX* as the basis to establish their model. They computed a “spring-like” reaction force that displaces the ribosome relative to the mRNA after translocation. They refer to this displacement as the secondary ribosome movement; translocation is the primary movement. The spring-like reaction force is modeled by calculating the change in energetic interactions between the 16S rRNA exposed tail and mRNA using the fundamental law of physics ($\Delta$energy $= -$force $\cdot$ $\Delta$distance). Vu et al. [7] call this ribosome spring-like reaction the ribosome spring model. The ribosome spring model convolves tRNA abundance, ribosome displacement, and spring-like reaction force to simulate the secondary ribosome movements during elongation. Shultzaberger et al. [5] and Salis et al. [1] also referred to the ribosome acting like a “rigid spring” when investigating optimal SD to start codon spacing at initiation. Tinoco et al. [36] also proposed that a “springy” 16S rRNA aSD:SD helix produced a -1 frameshift.

Vu et al. [7] then used the ribosome spring model to optimize heterologous protein production. The spring-like force can be realized as a change in the “phase” of the free energy sinusoidal signal. By changing codons of a gene using synonymous codons, this force can be minimized to reduce ribosome displacement and improve protein yield. Vu et al. [7] described the “phase” of the free energy sinusoidal signal that minimizes the spring-like force and ribosome displacement. That phase is termed the “species angle,” which is, as the name implies, unique to each organism. By minimizing displacement, Vu et al. [7] optimized protein production for two genes: Glutathione S-Transferase (GST) from *Schistosoma japonicum* and alcohol dehydrogenase (ADH) from *Clostridium ljungdahlii*. GST optimized
with their model showed an 11.6% increase over the wildtype, compared to a 4% increase over wildtype for CAI-optimized GST. ADH activity was 45% higher from the optimized gene compared to wildtype. The ribosome spring model takes into account ribosome displacement, tRNA abundance, and spring-like reaction force to produce a “wait-time” parameter at each codon. Minimizing these wait-time parameters is crucial to producing proteins efficiently [7].

3.4 Conclusions

Protein yield is expected to be proportional to the rate at which ribosomes initiate translation and the rate at which ribosomes finish translating the gene; i.e. bottlenecks exist in both initiation and elongation. Initiation and elongation optimization should be considered to maximize translation efficiency and protein yield. Optimization of initiation should consider SD sequence, SD sequence to start codon spacing, and mRNA secondary structures. Experimental evidence suggests that the free energy from -4 to +38 nucleotides should be less than -11 kcal/mol to minimize secondary structures at initiation [10]. Salis’s RBS calculator [1] optimizes all of these determinants for initiation. Optimization of elongation also needs to be considered. Optimizing elongation should consider tRNA usage, ribosome displacement, and 16S rRNA tails:mRNA bindings. Vu’s RiboScan™ [7,37] software optimizes all of these determinants during elongation. Optimizing codon bias alone is not considered because codon bias plays less of a role in heterologous protein expression [22,57], especially during protein overexpression; tRNA usage and workload plays a greater role in this regard [21,22,23,54]. Vu et al. [7], Welch et al. [22,57], Kudla et al. [10], and Allert et al. [11] demonstrated that high codon bias does not indicate maximum protein yield. They
showed that genes with lower codon bias can produce more protein than genes with higher codon bias. Vu et al. [7] showed that maximal protein yield can be gained by optimizing tRNA usage while also optimizing 16S rRNA:mRNA tail bindings. This results in maximal ribosome speed and minimal ribosome displacement during elongation. Optimizing GC content alone is not considered; however, optimizing 16S rRNA tail:mRNA binding throughout the gene using Vu’s RiboScan™ algorithm [7,37] matches the GC’s content of the gene (locally and globally) to the GC content of the production organism. In conclusion, maximizing protein yield can be achieved by optimizing initiation using Salis’s RBS calculator software and optimizing elongation using Vu’s RiboScan™ software.
3.5 References


CHAPTER 4

A New Biophysical Model for Translation Elongation

4.1 Discovery of the Periodic Free Energy Signal

A periodic free energy signal was observed from the average Watson-Crick binding between the 16S rRNA 3’ terminal end “exposed tail” (3’-AUUACCUCUCCACUAG-5’) and the mRNA during translation, as shown in Figure 4.1 [1,2]. The most prominent binding energy is at initiation which corresponds to the anti-Shine-Dalgarno (aSD) of the “exposed tail” binding to the Shine-Dalgarno (SD) sequence. After that, there is a periodic (sinusoidal-like) binding signal (in which negative free energy indicates binding) that corresponds to “in-frame” ribosome translocation during elongation [1,2,3]. Fourier transformation of this periodic signal revealed a prominent peak at a frequency of 1/3 cycles per nucleotide [2,4]. This illustrates that the exposed tail binds to the mRNA at every codon during elongation. However, after the stop codon is reached, the signal quickly attenuates, which indicates that the ribosome has stopped translating, as shown in Figure 4.2. Ribosome profiling experiments of Li et al. [5] provide physical evidence of the continuous interaction between the 16S rRNA “exposed tail” and the mRNA during elongation. Weiss et al. [6] and Larsen et al. [7] also showed that the aSD of the 16S rRNA “exposed tail” binds with the mRNA to produce a programmed translational frameshift; their observations implied that the exposed tail interacts with the mRNA during elongation. Li et al. [5], Weiss et al. [6], and Larsen et al. [7] provide corroborating physical evidence for the periodic free energy signal which suggested that the 16S rRNA “exposed tail” interacts with mRNA during elongation. The author of this dissertation suggests that the functional purpose of the 16S rRNA tail:mRNA
binding is to keep the ribosome in the proper reading frame, or in other words, the binding serves as reading frame maintenance. Heterologous protein translation disrupts this binding and therefore “displaces” the ribosome. This periodic free energy signal inspired a new mechanistic “spring” model for translation elongation as described in Chapter 4.2.

![Figure 4.1. Plot of the average free energy signal. This signal is obtained by calculating hybridization energy at every nucleotide for 200 non-frame-shifting E. coli genes with length greater than 1000 nucleotides.](image)
Figure 4.2. Plot of the average signal post stop codon. This signal is also obtained by calculating hybridization energy of 200 non-frame-shifting *E. coli* endogenous genes after the stop codon of the open reading frames.

4.2 Ribosome “Spring” Model and Ribosome Displacement

4.2.1 Ribosome Spring Model

Analysis of the free energy signal for the *E. coli* gene *prfB* using the cumulative magnitude and phase method showed a change in free energy for both the magnitude and phase at the frameshift site [1,8]. Using a fundamental law of physics, see Equation (4.1), the force exerted on the ribosome at the frameshift site can be calculated.

\[
\Delta \text{energy} = -\text{force} \cdot \Delta \text{distance}
\]  

(4.1)

This inspired the development of the ribosome “spring” model for translation elongation using second-order free energy effects from the 16S rRNA 3’ terminal end:mRNA bindings that elucidate frameshifts and illustrate a key concept: ribosome displacement. The
The ribosome spring model was also developed to predict and optimize protein yield in heterologous protein production applications [9].

As the ribosome translocates along the mRNA, the 16S rRNA 3’ terminal end interacts with bases upstream of the ribosome [5,6,7], as shown in Figure 4.3. If that binding energy is sufficiently large and is on the “wrong” side of favoring normal in-frame binding, the binding between the 16S rRNA:mRNA acts like a spring and may exert extra force on the ribosome and displace it. In the extreme case, a frameshift may occur. The “spring-like” reaction force is modeled using a sinusoidal or polynomial curve.

Figure 4.3. Interactions between ribosomal tail and mRNA. The 16S rRNA “exposed tail” continuously interacts with the mRNA during translation elongation for which the free energy from the Watson and Crick binding can be calculated.
For example, in the +1 programmed frameshifting gene prfB (Figure 4.4), the aSD of the 16S exposed tail binds to the SD-like sequence 3 bases (UAU in Figure 4.4) upstream from the P-site [6], resulting in -9.5 kcal/mol of binding energy. This binding is too close to the P-site and thus compresses the distance between the tail and the ribosome. It behaves like a compressed spring, which upon relaxation displaces the ribosome three quarters of a base downstream into the +1 reading frame where the spring is at minimal energy. The displaced ribosome then picks up the aminoacyl-tRNA in the +1 frame. This leads to a one-base frameshift after which the new reading frame is maintained [1,3,8].

Figure 4.4. Physical illustration during translation of prfB at frameshift site, codon 26. The aSD of 16S “exposed tail” binds too close, 3 nucleotides, to the P-site at codon 26 (stop) and compresses the “spring” that displaces ribosome towards +1 reading frame. The simulation illustration is depicted in Figure 4.5. Vu et al. [9] © 2014 IEEE.
Figure 4.5. Simulation illustration during translation of prfB at frameshift site, codon 26.
The energy from compressed spring is shown. This energy is calculated from the free energy resulting from the binding between the SD-like sequence on the mRNA and the aSD of 16S “exposed tail”. The binding compresses the “spring” that displaces ribosome towards minimal energy in +1 reading frame. Vu et al. [9] © 2014 IEEE.

In contrast, the SD in the -1 frameshifting gene of dnaX is 10 bases away from the P-site [7]. This binding extends the “spring” and displaces the ribosome backwards to produce a “partial” frameshift (on which the ribosome now lies between reading frames; producing both τ and γ subunits) [7]. Recent optical tweezer and structural experiments of Tinoco et al. [10] suggest that the aSD forms a springy helix with the SD-like sequence to produce a -1 programmed frameshift.

Figure 4.6 shows the E. coli 16S aSD during in-frame translation of lacZ. The binding energies around that site range from 0 to -1 kcal/mol. Because the aSD is bound 8
nucleotides away from the P-site, the “spring” is in its “relaxed” state. While slight misalignments between the zero reading frame and A-site are possible, there is not enough “spring force” to cause a frameshift. Interestingly, the spacing between the aSD and the P-site of 5 nucleotides in the relaxed spring during elongation also corresponds to the optimal SD and start codon spacing at initiation, which was observed by Chen et al. [11]. The author of this dissertation proposes that a “relaxed spring” state also occurs when the “optimal” spacing between the SD and start codon is 5 to 8 bases at the translational initiation site [11,12,13]. This compression or extension of the spring leads to the proposed concept: ribosome displacement.

**Figure 4.6. Physical illustration during translation of lacZ at codon 70.** The aSD binds 8 nucleotides from the P-site resulting in a “relaxed spring” with little to no ribosome displacement. The distance between the aSD to the P-site is 5 bases. The simulation illustration is depicted in Figure 4.7. Vu et al. [9] © 2014 IEEE.
4.2.2 Ribosome Displacement

Ribosome displacement is the misalignment between the ribosome A-site and the normal 0 reading frame (see Figure 4.8) caused by the force from the aforementioned “spring”. This misalignment is a distance on the order of fractions of a nucleotide. The degree to which a ribosome is displaced is dependent on the magnitude of the force and the aa-tRNA abundance of the codon in the 0 reading frame. For example, if the force is high and the aa-tRNA abundance for the codon in the 0 reading frame is also high, then displacement will be mitigated. However, if the force is high and the aa-tRNA abundance in the 0 reading frame is low, then the displacement will be amplified. A ribosome that is more...
displaced will take longer to choose between the two aa-tRNAs of the two available reading frames because tRNA binding is disrupted; this, therefore, increases the ribosome “wait-time”. The aa-tRNA abundance of codons in the three reading frames has a major impact on the ribosome wait time; this is modeled by tRNA binding competition as described in Chapter 5. A displaced ribosome can also pick up the wrong aa-tRNA in either the +1 or -1 reading frame, as shown in Figure 4.8.

**Figure 4.8. Displaced vs non-displaced ribosome.** Left) Illustration of a non-displaced ribosome; the A-site is in perfect alignment with the 0 reading frame. Right) Illustration of a displaced ribosome between the 0 reading frame and +1 reading frame resulting in a misalignment. The A-site sees both the codon in the 0 reading frame and +1 reading frame and is capable of picking up the aminoacyl-tRNA in either reading frame. In this example, the ribosome can pick up either the aa-tRNA coded by ACG codon or aa-tRNA coded by CGG codon.

Ribosome displacement is cumulative and does not reset after translocation. This suggests that the ribosome has “memory” during translation elongation. Because the ribosome has memory, ribosome displacement can be “fixed” anywhere on the mRNA by the use of “slow” codons and favorable force. “Slow” codons are codons that use low abundance aa-tRNA. For example, if the ribosome is displaced towards the +1 reading frame, a force towards the 5’ end (modeled as an extended spring) coupled with a slow codon can realign the ribosome.
In the model, one displacement unit corresponds to a misalignment of half a nucleotide. Two displacement units is a misalignment of a full nucleotide or a shift in reading frame. Ribosome displacement is calculated at each codon during translation elongation. Figure 4.9 shows the ribosome displacement plot for translation of +1 programmed frameshifting gene \textit{prfB}.

![Figure 4.9. Ribosome displacement plot of \textit{prfB} during translation elongation. Displacement shifts towards +2 displacement units (1 nucleotide) at the frameshift site (codon 26) indicating a +1 frameshifting event.]

\textbf{4.2.3 Translation Elongation Dynamics}

After initiation, the ribosome enters into the elongation phase. During elongation, the ribosome awaits the delivery of the next aminoacyl-tRNA (aa-tRNA). After delivery of the aa-tRNA, transpeptidation begins where the polypeptide chain is transferred from the peptidyl-tRNA (p-tRNA) to the aa-tRNA. This triggers the rotation of the ribosome into the hybrid conformation where the acceptor end of the p-tRNA and aa-tRNA fluctuates. Rotation of the 30S head and binding of the EF-Tu:GTP complex stabilizes this fluctuation and “unlocks” the mRNA from the ribosome in preparation for translocation. EF-Tu hydrolysis
of GTP translocates the ribosome 3 nucleotides downstream. The author proposes that just after translocation when the mRNA is “unlocked”, the 16S rRNA “exposed tail” binds to the mRNA and acts like a spring to displace the ribosome in either the +1 or -1 reading frame while it waits for the delivery of the next aminoacyl-tRNA. At this stage, the ribosome also rotates back into the classic confirmation. The degree to which the ribosome is displaced is a function of the force magnitude and how long the ribosome has to wait for the delivery of the next aminoacyl-tRNA, which is related to aa-tRNA abundance. A ribosome that is more displaced has to wait a longer period before picking up the next aminoacyl-tRNA due to tRNA binding disruption. A displaced ribosome is also more likely to pick up the wrong aminoacyl-tRNA from the +1 or -1 reading frame. After the ribosome has picked up the next aa-tRNA, the ribosome enters transpeptidation, and the cycle repeats.

The author of this dissertation suggests this spring mechanism acts to keep the ribosome in the appropriate reading frame when translating native genes with the exception of programmed frameshifts. The corollary is that the purpose of this mechanism is to reduce the “noise” produced from translocation; i.e. translocation does not always move the ribosome exactly three nucleotides, but approximately 3 nucleotides plus or minus fractions of a nucleotide. This “noise” is due to the entropy from transducing hydrolysis energy into kinetic energy to translocate the ribosome. Translating heterologous genes disrupts “normally evolved” 16S rRNA tail:mRNA binding, which in turn may lead to a displaced ribosome that increases ribosome wait time, creates bottlenecks, or causes frameshifts. Programmed translational frameshift sites like those found in prfB and dnaX exist [6,7,10,14,15] and their efficiency relies on the aforementioned ribosome interactions. The mechanisms behind
programmed translation frameshifts [9,10] support the model and are directly relevant to ribosome dynamics during normal translation elongation.

4.3 Ribosome Wait Times and Translation Bottlenecks

4.3.1 Ribosome Wait Times

The ribosome “wait time” at each codon is calculated as a function of the force from the spring, ribosome displacement, and tRNA abundance. The model computes the ribosome “wait time” at each codon during translation elongation. The output of the “wait time” is in units of “cycle” as measure of how long it takes the ribosome to load the next aa-tRNA and is described in Chapter 5. Figure 4.10 shows the ribosome “wait time” plot of frameshifting gene prfB. The ribosome “pauses” at the frameshift site (codon 26) as it is being “displaced.” Li et al. [5] also proposed that the ribosome “pauses” at the SD-like sequence in prfB due to SD:aSD binding; however, they do not propose a biophysical mechanism for the pause.

![Figure 4.10. Ribosome wait time plot of prfB during translation elongation.](image)

Figure 4.10. Ribosome wait time plot of prfB during translation elongation. High ribosome wait time of 147 cycles at codon 26 indicates the ribosome pauses at the frameshift site. This illustrates that ribosome displacement (Figure 4.9) contributes to ribosome wait time.
Total Wait Time is the sum of all the ribosome wait time at each codon. Total Wait Time can be used as an indicator of a gene’s translation efficiency.

### 4.3.2 Translation Bottlenecks

While Total Wait Time is a good “global” predictor of protein yield, a “local” predictor index may be a better estimator. Because mRNA is concurrently translated into proteins by multiple ribosomes, multiple “slow” translation regions clustered in local proximity are likely to be more detrimental to yield than impeding regions distributed across the gene. Two genes can have the same total cycle but different “slow” translation clustering sites. This is referred to as “translation bottlenecks” or “ribosome traffic jams”. To compute the local translation bottleneck index, a summing sliding window of size 20 codons (i.e., an approximate ribosome footprint on the mRNA [23]) was used. The ribosome “wait times” were summed within this window and plotted on a translation bottleneck plot, as shown in Figure 4.11. The window was then advanced one codon, and the sum was recomputed and then plotted as the next value. The traffic jam principle states that the translation efficiency of a gene is bottlenecked by slowest region of translation. Therefore, the max peak (BNI) of the translation bottleneck plot (Figure 4.11) was used as the yield predictor index to determine the gene’s translation efficiency; the max peak represents the slowest region during translation. BNI stands for bottleneck index and shows the location of the slowest ribosome traffic jam. The mathematics to calculate the translation bottleneck plot and BNI is described in Chapter 5.3.1. Bottlenecks also exist during initiation and termination; however, these are not automatically computed in the model. Bottlenecks at initiation and termination can be added manually to improve protein yield predictions.
Figure 4.11. Translation bottleneck plot of prfB throughout translation elongation. High ribosome wait time at frameshift site due to ribosome frameshifting produces a bottleneck of 329 at codon 26.

4.4 Gene Optimization

The author of this dissertation postulates that keeping the displacement of a gene close to zero and maximizing tRNA arrival time by using the most abundant aa-tRNA decreases ribosome wait time and translation bottleneck and therefore increases protein yield. A displacement close to zero ensures that the A-site is aligned close to the zero reading frame. This can be accomplished by choosing the codon that codes for the most abundant aa-tRNA while keeping the phase angle of the gene close to the “species angle” [9]. This results in minimal “spring” compression or extension during translocation and therefore minimizes force and displacement. Optimizing in this manner increases ribosome speed by using the most abundant tRNA and maximizing tRNA binding accuracy by minimizing displacement. The result is that ribosome wait times and translation bottlenecks will be reduced throughout translation elongation. Therefore, total ribosome wait time and BNI will also be lowered. Gene designs incorporating these concepts are made by changing the genetic sequence using synonymous codons while conserving the amino acid sequence. Conversely, genes can be
altered to make protein production “less optimal”. A decrease in protein yield, with potential applications in pathway optimization and production of “toxic” protein can be accomplished by keeping displacement further away from 0 but between -1 and +1 to avoid possible frameshifts. This increases the ribosome wait time at each codon.

This method is feasible at low mRNA transcript levels when the aa-tRNA abundance pool is not taxed and depleted. However, at high mRNA transcript levels, it is best to spread the workload between many different tRNA (coding for the same amino acid) rather than using one single tRNA or a single set of tRNA. This relates to the “recharge” rates of tRNA from uncharge to charge [16,17,18,19]. One still needs to stay close to the species angle while spreading the workload across multiple tRNA to avoid displacing the ribosome.

There are 61 codons coding for 20 amino acids using 45 different tRNA species [20]. This implies that amino acids have multiple codons coding for multiple aa-tRNAs. For example serine has 6 codons coding for 4 different serine-tRNAs. This was explained in the wobble phenomenon by Crick et al. [21]. However, not all codons code the most abundant tRNA. Valine, for example, has four codons (GUA, GUC, GUG, GUU) coding for three valine-tRNA (val1, val2a, val2b) [20]. Only GUU codes for all three tRNA and is considered “the codon that codes the most abundant tRNA” (see Table 4.1). GUA and GUG codes for val1, and GUC codes for val2a and val2b. tRNA val1 is the most abundant out of the three valine-tRNA. The second most abundant codon is GUA and GUG; tRNA abundance information can be found in Dong et al. [20]. Table 4.1 lists the amino acids, codons, tRNA, and tRNA abundance for the codons. The sum of abundances for the respective tRNA is used when a codon codes for more than one tRNA; i.e., the tRNA abundance of valine codon
GUU is the sum of the abundance for val1, val2a, and val2b tRNAs. Table 4.1 also shows the codon preference in gene optimization. Only amino acids that code for multiple tRNAs are listed.

Table 4.1. Amino acid, codons, cognate tRNA, and tRNA abundance per codon. tRNA abundance and cognate tRNA information taken from Dong et al. [20]. tRNA abundance is taken from Table 3 in Dong et al. using 0.7 doublings/hour. Green indicates preferred codon used for optimization. Yellow indicates secondary preferred codon.

<table>
<thead>
<tr>
<th>Amino Acid</th>
<th>Codon</th>
<th>tRNA</th>
<th>tRNA Abundance</th>
<th>Amino Acid</th>
<th>Codon</th>
<th>tRNA</th>
<th>tRNA Abundance</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALA</td>
<td>GCA</td>
<td>ala1b</td>
<td>0.54</td>
<td>LEU</td>
<td>CUU</td>
<td>leu2</td>
<td>0.27</td>
</tr>
<tr>
<td>ALA</td>
<td>GCC</td>
<td>ala2</td>
<td>0.1</td>
<td>LEU</td>
<td>UUA</td>
<td>leu5</td>
<td>0.16</td>
</tr>
<tr>
<td>ALA</td>
<td>GCG</td>
<td>ala1b</td>
<td>0.54</td>
<td>LEU</td>
<td>UUG</td>
<td>leu4,5</td>
<td>0.45</td>
</tr>
<tr>
<td>ALA</td>
<td>GCU</td>
<td>ala1b</td>
<td>0.54</td>
<td>PRO</td>
<td>CCA</td>
<td>pro3</td>
<td>0.09</td>
</tr>
<tr>
<td>ARG</td>
<td>AGA</td>
<td>arg4</td>
<td>0.11</td>
<td>PRO</td>
<td>CCC</td>
<td>pro2</td>
<td>0.12</td>
</tr>
<tr>
<td>ARG</td>
<td>AGG</td>
<td>arg4,5</td>
<td>0.18</td>
<td>PRO</td>
<td>CCG</td>
<td>pro1,3</td>
<td>0.2</td>
</tr>
<tr>
<td>ARG</td>
<td>CGA</td>
<td>arg2</td>
<td>0.67</td>
<td>PRO</td>
<td>CCU</td>
<td>pro2,3</td>
<td>0.21</td>
</tr>
<tr>
<td>ARG</td>
<td>CGC</td>
<td>arg2</td>
<td>0.67</td>
<td>SER</td>
<td>AGC</td>
<td>ser3</td>
<td>0.2</td>
</tr>
<tr>
<td>ARG</td>
<td>CGG</td>
<td>arg3</td>
<td>0.12</td>
<td>SER</td>
<td>AGU</td>
<td>ser3</td>
<td>0.2</td>
</tr>
<tr>
<td>ARG</td>
<td>CGU</td>
<td>arg2</td>
<td>0.67</td>
<td>SER</td>
<td>UCA</td>
<td>ser1</td>
<td>0.25</td>
</tr>
<tr>
<td>GLN</td>
<td>CAA</td>
<td>gln1</td>
<td>0.12</td>
<td>SER</td>
<td>UCC</td>
<td>ser5</td>
<td>0.12</td>
</tr>
<tr>
<td>GLN</td>
<td>CAG</td>
<td>gln2</td>
<td>0.14</td>
<td>SER</td>
<td>UCG</td>
<td>ser1,2</td>
<td>0.37</td>
</tr>
<tr>
<td>GLY</td>
<td>GGA</td>
<td>gly2</td>
<td>0.198</td>
<td>SER</td>
<td>UCU</td>
<td>ser1,5</td>
<td>0.37</td>
</tr>
<tr>
<td>GLY</td>
<td>GCC</td>
<td>gly3</td>
<td>0.7</td>
<td>THR</td>
<td>ACA</td>
<td>thr4</td>
<td>0.14</td>
</tr>
<tr>
<td>GLY</td>
<td>GGG</td>
<td>gly1,2</td>
<td>0.33</td>
<td>THR</td>
<td>ACC</td>
<td>thr1,3</td>
<td>0.19</td>
</tr>
<tr>
<td>GLY</td>
<td>GGU</td>
<td>gly3</td>
<td>0.7</td>
<td>THR</td>
<td>ACG</td>
<td>thr2,4</td>
<td>0.23</td>
</tr>
<tr>
<td>ILE</td>
<td>AUA</td>
<td>ile2</td>
<td>0.027</td>
<td>THR</td>
<td>ACU</td>
<td>thr1,3,4</td>
<td>0.33</td>
</tr>
<tr>
<td>ILE</td>
<td>AUC</td>
<td>ile1</td>
<td>0.513</td>
<td>VAL</td>
<td>GUA</td>
<td>val1</td>
<td>0.55</td>
</tr>
<tr>
<td>ILE</td>
<td>AUU</td>
<td>ile1</td>
<td>0.513</td>
<td>VAL</td>
<td>GUC</td>
<td>val2a, val2b</td>
<td>0.2</td>
</tr>
<tr>
<td>LEU</td>
<td>CUA</td>
<td>leu3</td>
<td>0.11</td>
<td>VAL</td>
<td>GUG</td>
<td>val1</td>
<td>0.55</td>
</tr>
<tr>
<td>LEU</td>
<td>CUC</td>
<td>leu2</td>
<td>0.16</td>
<td>VAL</td>
<td>GUU</td>
<td>val1,2a,2b</td>
<td>0.75</td>
</tr>
<tr>
<td>LEU</td>
<td>CUG</td>
<td>leu1,3</td>
<td>0.79</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.5 Conclusions

A new computational-based biophysical model for translation elongation has been developed to elucidate frameshifting genes prfB, dnaX and to simulate ribosome dynamics during normal translation elongation through an mRNA. The model incorporates an energetic “spring” of 16S rRNA tail and mRNA interactions, ribosome displacement, and aa-tRNA abundance, leading to a ribosome "wait time" parameter for the gene(s) of interest. From those wait time parameters, location of translation bottlenecks can be identified.

This represents a comprehensive strategy for evaluating ribosome dynamics and translational efficiency. The model exists as a fully implemented software package (RiboScan™) that provides a new approach to protein production engineering; this software package will be released as a webserver at [22]. The model can be used for predicting and optimizing genes for heterologous protein production in E. coli, can be expanded to different production organisms, and has potential applications for predicting protein synthesis levels from complete genome sequences.
4.6 References


CHAPTER 5

The Algorithms and Mathematics

5.1 Introduction

The ribosome spring model and ribosome displacement as described in Chapter 4 are modeled using a step-size integrator and a “probability cycle” algorithm. This probability cycle algorithm calculates ribosome displacement and ribosome “wait time” at every codon using the force convolved with the probabilities of picking up an aminoacyl-tRNA (aa-tRNA). The ribosome is displaced at every “cycle” by a step size function which is computed as the force multiplied by a step size constant, delta T (dT); a cycle is defined as the state in which the ribosome has not picked up an aa-tRNA. The probability cycle is described in Chapter 5.3.

The force exerted on the ribosome can be calculated using a fundamental law of physics: force equals negative change in energy with respect to change in distance. The free energy signal captures the energetic binding of the 16S rRNA tail:mRNA with respect to distance (displacement) and is modeled using a sinusoid or polynomial. Therefore, the force from the spring is computed as the negative derivative of the free energy signal; the derivative of the free energy signal represents the change in energy with respect to change in distance. Calculations of force and free energy signal are described in Chapter 5.4.

Three probabilities are used to determine if the ribosome has “picked up” the aa-tRNA from the three reading frames. Another probability is used to determine if the ribosome has not picked up an aa-tRNA and “do nothing” which re-executes the cycle. The
probabilities of picking up an aa-tRNA are computed as a function of the “view curve” and of the aa-tRNA abundance for codons in the three reading frames. The view curve models the degree to which the A-site “sees” each of the three codons in the three reading frame as a function of ribosome displacement. Descriptions of the view curve and aa-tRNA abundance are in Chapter 5.5.

After the ribosome picks up the next aa-tRNA, the cycle ends, and the ribosome translocates to the next codon. Ribosome displacement is cumulative and does not reset after translocation. The “probability cycle” algorithm is executed at every codon until a stop codon is reached. The “main algorithm” oversees the execution of all the sub-algorithms. The inputs into the algorithm are: 1) the 13 nucleotides from the 16S rRNA “exposed tail”, 2) the mRNA sequence with 15 nucleotides before the start codon, and 3) the aa-tRNA abundance for each codon.

5.2 The Main Algorithm

The main algorithm begins simulating ribosome translation with codon 2 in the ribosome A-site and codon 1 in the P-site. The ribosome displacement is initialized to 0. The discrete free energy signal is transformed into a continuous free energy signal that is modeled by a sinusoid or polynomial. The sinusoid or polynomial is then superimposed onto displacement which is in units of nucleotides; this continuous free energy signal represents the rigid “spring” of the 16S rRNA tail:mRNA binding. The force from the spring can be computed by taking the derivative of this continuous free energy signal evaluated at the current ribosome displacement.
The probability cycle computes the ribosome displacement at every codon using a step size function convolved with the probabilities of picking an aa-tRNA; i.e., the ribosome is displaced by the step size function every cycle until it picks up an aa-tRNA. The step size function is computed as the force evaluated at the current displacement multiplied by a step size delta T (dT). Ribosome displacement changes every cycle; thus, the force exerted on the ribosome changes as well. After the ribosome has picked up an aa-tRNA, the ribosome will 1) frameshift if it picks up the wrong aa-tRNA in the +1 or -1 reading frame then translocate, or 2) perform normal in-frame translocation to the next codon if it picked up the aa-tRNA in the 0 reading frame.

Ribosome displacement from the previous codon is cumulative and does not reset. After translocation, the new force is computed from the new continuous free energy signal at the new ribosome displacement position, and the “probability cycle” algorithm is executed for that codon. The ribosome “wait time” at each codon is determined by the number of times the probability cycles are executed; the wait time is a measure of the amount of cycles needed for the ribosome to pick up an aa-tRNA. The main algorithm executes until a stop codon is recognized, which terminates the ribosome translation simulation. Refer to Figure 5.1 for the flow chart of the main algorithm.
Figure 5.1. Flow chart of the main algorithm. The main algorithm starts with codon 2 in the A-site. Force is computed from free energy signal. The probability cycle executes to determine ribosome displacement at each codon. After picking up an aa-tRNA, the ribosome either frameshifts then translocates or performs regular in-frame translocation. If a stop codon is recognized, the main algorithm terminates ribosome translation simulation.
5.3 The Probability Cycle

The probability cycle algorithm computes ribosome displacement at each codon using a step size function convolved with the probabilities of picking up or not picking up an aa-tRNA. The ribosome is displaced at every cycle by a step size function that is computed as the force at the current ribosome displacement multiplied by step size constant delta T ($dT$) as shown in Equation (5.1). The force calculation is described in Chapter 5.4. $dT$ is a parameter that can be adjusted. The force changes every time the cycle is re-executed because the ribosome displacement has also changed. Therefore, the force and ribosome displacement are co-dependent on each other.

$$RD_{New} = RD_{current} + force(RD_{current}) \cdot dT$$  \hspace{1cm} (5.1)

$RD_{New}$ is the new ribosome displacement calculated at every cycle, $RD_{current}$ is the current ribosome displacement, and $dT$ is a step size constant. At start of translation, $RD_{current}$ is initialized to 0. $RD_{New}$ becomes $RD_{current}$ at the beginning of each cycle. $RD_{current}$ does not reset and keeps the same value after translocation.

The probability cycle repeats until the ribosome has picked up an aa-tRNA, which is modeled by the four probabilities. The four probabilities are defined as $P_{-1}$, $P_0$, $P_{+1}$, and $P_n$ where $P_{-1}$, $P_0$, and $P_{+1}$ are the probabilities of picking up the aa-tRNA in the -1, 0, +1 reading frames, respectively, and $P_n$ is the probability of not picking up an aa-tRNA and re-iterating through the cycle.

The probabilities of picking up an aa-tRNA are computed as a function of the “view curve” (the degree to which the A-site “sees” the codon in the respective reading frame) and aa-tRNA abundance for the respective reading frames. A ribosome that is more displaced
“sees” less of the codon in the 0 reading frame and more of the codon in the adjacent reading frame. The probability of picking up an aa-tRNA in a given reading frame is the product of how much of the codon is seen by the A site, as modeled by the view curve, and the aa-tRNA abundance for that codon. Therefore, a displaced ribosome has a smaller chance of picking up the aa-tRNA in the 0 reading frame and a greater chance of picking up the aa-tRNA in the adjacent reading frame compared to a non-displaced ribosome. The view curve and aa-tRNA abundance are described in Chapter 5.5.

A probability that “wins” is defined by having the greatest value relative to the other probabilities. If P_0 “wins”, (i.e. the ribosome has picked up the aa-tRNA in the 0 reading frame), then the probability cycle terminates, and the ribosome does an in-frame translocation to the next codon. If P_1 wins, the ribosome picks up the aa-tRNA in the -1 reading frame, “frameshifts” to the -1 reading frame, and then translocate in the new reading frame. If P_{-1} wins, the ribosome picks up the aa-tRNA in the +1 reading frame, “frameshifts” to the +1 reading frame, and then translocate in the new reading frame. In this context, “frameshifting” means that the probabilities and force for the next codon are computed from the perspective of the new reading frame as opposed to the 0 reading frame; ribosome displacement still remains the same. If P_n wins, the ribosome is still waiting for the delivery of aa-tRNA, and the algorithm re-executes the probability cycle; therefore, the ribosome is again displaced by the step size function, and the probabilities are re-computed. The probability cycle continues until either P_{-1}, P_0, or P_{+1} wins. P_{-1}, P_0, and P_{+1} are computed “cumulatively” at every cycle i by calculating the probabilities of not picking an aa-tRNA.
$(\bar{P}_{-1}, \bar{P}_0, \bar{P}_{+1})$ at every cycle $i$ then subtracting those probabilities from 1 as shown in Equations (5.2), (5.3), and (5.4).

$$\bar{P}_{-1}^i = \bar{P}_{-1}^{i-1} * (1 - NF * (VC_{-1}(RD_{current}) * TA_{-1}))$$  \hspace{1cm} (5.2)$$

$$\bar{P}_0^i = \bar{P}_0^{i-1} * (1 - NF * VC_0(RD_{current}) * TA_0))$$  \hspace{1cm} (5.3)$$

$$\bar{P}_{+1}^i = \bar{P}_{+1}^{i-1} * (1 - NF * (VC_{+1}(RD_{current}) * TA_{+1}))$$  \hspace{1cm} (5.4)$$

$i$ is the index for of the current cycle; therefore $\bar{P}_0^{i-1}$ would be the probability computed from the previous cycle. $NF$ is the normalization constant to compensate for changes in $dT$. $VC_{-1}$, $VC_0$, $VC_{+1}$ is the view curve of the -1, 0, +1 reading frame respectively (Chapter 5.5.1). $RD_{current}$ is the current ribosome displacement. $TA_{-1}, TA_0, TA_{+1}$ is the aa-tRNA abundance of the codon in the -1, 0, +1 reading frame respectively (Chapter 5.5.2). View curves and aa-tRNA abundance are described in Chapter 5.5.

Therefore, $P_{-1}^i, P_0^i, P_{+1}^i$ can be computed as:

$$P_{-1}^i = 1 - \bar{P}_{-1}^i$$ \hspace{1cm} (5.5)$$

$$P_0^i = 1 - \bar{P}_0^i$$ \hspace{1cm} (5.6)$$

$$P_{+1}^i = 1 - \bar{P}_{+1}^i$$ \hspace{1cm} (5.7)$$

And $P_n$ is computed as:

$$P_n = 1 - (P_{-1}^i + P_0^i + P_{+1}^i)$$ \hspace{1cm} (5.8)$$

Modeling $P_n$ in this manner (Equation 5.8) implies that the aa-tRNAs of the respective reading frame are competing for binding to the A-site of the ribosome.
Pseudo Code of “The Probability Cycle”:

// Initial Conditions:
\[ P_{-1} = 1, P_{0} = 1, P_{+1} = 1, \text{ Ribosome Displacement (RD)} = 0, dT = 0.004, \text{ Cycle Count} = 0 \]

Normalization Constant (NF) = \( dT/0.004 \)

// Start probability cycle at codon 2

// probability of not picking up a tRNA is set to 1 at beginning of cycle (\( P_{-1} = 1, P_{0} = 1, P_{+1} = 1 \))

// Cycle Begins
1. \( RD_{\text{New}} = RD_{\text{Current}} + \text{force (}RD_{\text{Current}}\text{)} \times dT \) // displace ribosome
2. \( RD_{\text{Current}} = RD_{\text{New}} \) // RD is ribosome displacement
3. Compute cumulative probabilities: \( P_{-1}, P_{0}, P_{+1} \) // probability of not picking up aa-tRNA
   i. \( P_{-1} = P_{-1} \times (1 - NF \times (VC_{-1}(RD_{\text{Current}}) \times TA_{-1})) \)
   ii. \( P_{0} = P_{0} \times (1 - NF \times (VC_{0}(RD_{\text{Current}}) \times TA_{0})) \)
   iii. \( P_{+1} = P_{+1} \times (1 - NF \times (VC_{+1}(RD_{\text{Current}}) \times TA_{+1})) \)
4. Compute \( P_{-1}, P_{0}, P_{+1}, P_{n} \) // probability of picking up aa-tRNA
   i. \( P_{-1} = 1 - P_{-1}; P_{0} = 1 - P_{0}; P_{+1} = 1 - P_{+1} \)
   ii. \( P_{n} = 1 - \text{sum} (P_{-1}, P_{0}, P_{+1}) \)
5. Cycle Count = Cycle Count + 1 // compute ribosome wait time
6. Repeat #1 until \( P_{n} \) is less than \( P_{-1}, P_{0} \), or \( P_{+1} \).

// Cycle Ended
7. Ribosome “wait time” at this codon = Cycle Count
8. Ribosome displacement at this codon = \( RD_{\text{Current}} \)
9. If ribosome displacement is greater than +1 units, “frameshift” to +1 reading frame and translocate
10. If ribosome displacement is less than -1 units, “frameshift” to -1 reading frame and translocate
11. If ribosome displacement is between -1 and +1 units, translocate in-frame to next codon
12. Keep current ribosome displacement (\( RD_{\text{Current}} \)) the same after translocation. Do not reset.
13. \( P_{-1} = P_{0} = P_{+1} = 1, \text{ Cycle Count} = 0 \). // reset probabilities and cycle count every cycle
14. Repeat #1 for next codon until a stop codon reached. If stop codon reached, terminate translation
5.3.1 Output of the Probability Cycle

The output of the probability cycle is ribosome displacement and ribosome wait time at each codon. Therefore, a graph of the ribosome displacement of a gene can be plotted, as shown in Figure 5.2. This is called the ribosome displacement plot. Ribosome wait time is calculated as the number of cycles required for the ribosome to pick up an aa-tRNA as described in the pseudo code; therefore, ribosome wait time is a function of ribosome displacement, aa-tRNA abundance, and force. A graph of the ribosome wait time of a gene can also be plotted, as shown in Figure 5.3.

![Figure 5.2. Ribosome displacement plot of lacZ.](image1)

Figure 5.2. Ribosome displacement plot of lacZ. This plot is generated as the algorithm progresses through each cycle and outputs a value for ribosome displacement at each codon position.

![Figure 5.3. Ribosome wait time plot of lacZ.](image2)

Figure 5.3. Ribosome wait time plot of lacZ. This plot is generated as the number of cycles required for the ribosome to “pick up” an aa-tRNA at each codon position.
5.3.2 Mathematics of the Bottleneck Plot and Bottleneck Index (BNI)

The ribosome wait times (as shown in Figure 5.3) were used to calculate the translation bottleneck plot and bottleneck index (BNI), which is depicted in Figure 5.4. The translation bottleneck plot and BNI show locations of ribosome traffic jams as explained in Chapter 4.3.2.

![Figure 5.4. Translation bottleneck plot of lacZ during translation elongation.](image)

Illustrates the partial sums using a sliding window size of 20 codons. Bottleneck (BNI) is 174 at codon 839.

The computations used to calculate the translation bottleneck plot and BNI are as follows:

Let $WS$ be the window size of the sliding window in units of codons. $WS$ corresponds to the length of the ribosome footprint on the mRNA [3], which is approximately 20 codons. Let $PS_i$ be the partial sum of the ribosome wait times in the sliding window where $i$ is the “codon position” on the translation bottleneck plot. Let $WT_j$ be the ribosome wait time at the $j^{th}$ codon position on the ribosome wait time plot where $j$ is the codon position on the ribosome wait time plot. Therefore the partial sum at the $i^{th}$ position ($PS_i$) on the translation bottleneck plot can be computed as:
\[ PS_i = \sum_{j=i}^{j+WS-1} WT_j \]  

where the max value for \( i \) is the window size (\( WS \)) subtracted from the max codon position in the ribosome wait time plot (or length of the gene in codons). For example, if the gene length is 200 codons, then the max codon position in the ribosome wait time plot is 200 codons. Therefore, the max value for \( i \) is 180. So the partial sum is calculated from codon position \( i=1 \) to codon position \( i=180 \) where the partial sum at \( i=180 \) is the sum of the ribosome wait time from \( j=180 \) to \( j=200 \). The partial sum at codon position \( i=180 \) represents the bottleneck for the ribosome to translate the last 20 codons.

The bottleneck index (BNI) is calculated by taking the maximum of the partial sums on the translation bottleneck plot:

\[ \text{BNI} = \text{MAX}(PS_i) \]  

5.4 Calculating Force from “Springy” 16S rRNA Tail

The probability cycle uses the force to displace the ribosome an incremental amount each “cycle”; see Equation (5.1). The fundamental law of physics states that the force equals the change in energy with respect to the change in distance. The continuous free energy signal captures the energetic binding of the 16S rRNA tail:mRNA with respect to distance (displacement) at every codon. Therefore, the force is calculated as the derivative of the continuous free energy signal; this derivative represents the change in energy with respect to distance.
change in distance. However, before the force can be calculated, the continuous free energy signal at each codon must first be computed.

5.4.1 Calculating the Continuous Free Energy Signal

The continuous free energy signal is computed from the discrete free energies of the Watson-Crick binding between the 16S rRNA exposed tail (3’-auuccacucusg-5’) and the mRNA. The discrete free energy signal can be converted into a continuous free energy signal by fitting the discrete free energy value on either a sinusoid or polynomial. The sinusoid uses three discrete free energy values ($\Delta G_{s-1}$, $\Delta G_s$, $\Delta G_{s+1}$) while the polynomial uses five discrete free energy values ($\Delta G_{s-2}$, $\Delta G_{s-1}$, $\Delta G_s$, $\Delta G_{s+1}$, $\Delta G_{s+2}$). The free energy values are calculated using the methods described in Appendix A and Mishra et al. [1]. The free energy values used in this dissertation are from the Freier et al. [4] free energy model. All modeling and calculations use the sinusoidal fitting in this dissertation. Polynomial fitting is an option that can be used instead of the sinusoidal fit. The sinusoidal fitting is nearly identical to the polynomial fitting.

Center Point and “Tail Distance”

The center point, $\Delta G_s$, of the continuous free energy signal is the free energy calculated when the exposed tail binds five bases away from middle base of the A-site codon (see Figure 5.5a). This distance is called the “tail distance” and is a parameter in the model. The tail distance was derived from modeling the frameshift site of prfB. Any distance greater or less than five bases would not frameshift the ribosome and/or keep the ribosome frameshifted in the +1 reading frame after frameshift. Surprisingly, this distance puts the
anti-Shine-Dalgarno sequence (3’-UCCUCC-5’) of the 16S rRNA tail 5 bases away from the P-site that corresponds to the optimal spacing between the Shine-Dalgarno to start codon at initiation [5]. The author believes that the tail distance may differ from organism to organism.

**Discrete Free Energy Indexing**

A sinusoid or polynomial fitting method can be used to convert the discrete free energy values into a continuous free energy signal. The sinusoid fitting uses three discrete free energy values annotated as $\Delta G_{n-1}$, $\Delta G_n$, and $\Delta G_{n+1}$, while the polynomial fitting uses five discrete free energy values annotated as $\Delta G_{n-2}$, $\Delta G_{n-1}$, $\Delta G_n$, $\Delta G_{n+1}$, and $\Delta G_{n+2}$. $\Delta G_{n-1}$ is the free energy value calculated when the exposed tail is shifted one base **upstream** relative to the $\Delta G_n$ binding position, as shown in Figure 5.5b. In this position, the tail is 6 bases away from the center of the A-site. $\Delta G_{n+1}$ is the free energy value calculated when the exposed tail is shifted one base **downstream** relative to the $\Delta G_n$ binding position, as shown in Figure 5.5c. In this position, the tail is 4 bases away from the center of the A-site. $\Delta G_{n-2}$ is free energy values calculated by shifting the exposed tail 2 bases upstream relative to the $\Delta G_n$ binding position or 7 bases from the center of the A-site. $\Delta G_{n+2}$ indicates free energy values calculated by shifting the exposed tail 2 bases downstream relative to the $\Delta G_n$ binding position or 3 bases away from the center of the A-site.
Figure 5.5. Variation in free energy based on tail:mRNA spacing. (a) Free energy value ($\Delta G_n$) is calculated from the binding between the 16S rRNA exposed tail and the mRNA 5 bases away from the center of the A-site. (b) Free energy value ($\Delta G_{n-1}$) is calculated from 16S rRNA exposed tail:mRNA binding 6 bases away from the center of the A-site, or a one base shift upstream from $\Delta G_n$ binding position. (c) Free energy value ($\Delta G_{n+1}$) is calculated from the binding between 16S rRNA exposed tail and mRNA 4 bases away from the center of the A-site, or one base shift downstream relative to the $\Delta G_n$ binding position. Free energy calculation is described in Appendix A.

Sinusoidal Fitting of Discrete Free Energy Signal and Force Calculations

The sinusoidal fitting uses discrete free energy values ($\Delta G_{n-1}$, $\Delta G_n$, and $\Delta G_{n+1}$) to convert to a continuous free energy signal. For simplicity in annotation, $\Delta G_{n-1}$, $\Delta G_n$, and $\Delta G_{n+1}$ are re-annotated as $A$, $B$, and $C$, respectively. Therefore, the continuous free energy signal can be calculated using the following method.
Compute the DC constant from the free energy values.

\[ DC = \frac{A+B+C}{3} \]  \hspace{1cm} (5.11)

Subtract DC constant from the free energy values. DC is not needed to compute the force.

\[ a = A - DC \]  \hspace{1cm} (5.12)
\[ b = B - DC \]  \hspace{1cm} (5.13)
\[ c = C - DC \]  \hspace{1cm} (5.14)

Modified free energy values \(a\), \(b\), and \(c\), can be fitted on a sinusoid with a frequency of 1/3 cycles per nucleotide; this can be done because the average free energy signal has a frequency of 1/3 cycles per nucleotide [6,7]. \(a\) is fitted at zero degrees on the sinusoidal as defined in Equation (5.15). \(b\) is fitted with a 120 degree phase shift on the sinusoid as defined in Equation (5.16). \(c\) is fitted with a 240 degrees phase shift on the sinusoidal as defined in Equation (5.17).

\[ a = M \cdot \sin(\Phi) \]  \hspace{1cm} (5.15)
\[ b = M \cdot \sin(\Phi + \frac{2\pi}{3}) \]  \hspace{1cm} (5.16)
\[ c = M \cdot \sin(\Phi + \frac{4\pi}{3}) \]  \hspace{1cm} (5.17)

This leaves 3 equations and 2 unknowns \((M\) and \(\Phi\)). Equations (5.16) and (5.17) are expanded using the following trigonometric identity for sine expansion:
\[ \sin(x + y) = \sin(x) \cdot \cos(y) + \cos(x) \cdot \sin(y) \quad (5.18) \]

Subtracting the resulting expanded equations of \textbf{Equation (5.16)} and (5.17) will result in:

\[ M \cdot \cos(\Phi) = \frac{b - c}{\sqrt{3}} \quad (5.19) \]

Using the trigonometric identity for the expansion of tangent:

\[ \tan(\Phi) = \frac{\sin \Phi}{\cos \Phi} \quad (5.20) \]

and \textbf{Equations (5.15) and (5.19)}, \( \Phi \) can be computed:

\[ \Phi = \tan^{-1}\left(\frac{a \cdot \sqrt{3}}{b - c}\right) \quad (5.21) \]

Using the trigonometric identity:

\[ \sin(x)^2 + \cos(x)^2 = 1 \quad (5.22) \]

and \textbf{Equations (5.15) and (5.19)}, \( M \) can be computed:

\[ M = \sqrt{(a)^2 + \frac{(b-c)^2}{3}} \quad (5.23) \]

The continuous free energy signal using sinusoidal fitting is then superimposed on displacement units of -3 to 3:

\[ M \cdot \sin\left(\frac{x \cdot \pi}{3} + \Phi\right) \quad (5.24) \]
The signal is then shifted for minimal energy at the species angle and to compensate for the “tail distance”. \textbf{Equation (5.25)} models the energy of the ribosome “spring” as function of ribosome displacement (RD).

\begin{equation}
\text{Energy(RD)} = M \cdot \sin \left( \frac{x \cdot \pi}{3} + \Phi - \theta_{sp} - \delta - \frac{\pi}{2} \right) \quad (5.25)
\end{equation}

$\theta_{sp}$ (radians) is the “species angle”: The species angle for \textit{E. coli} is -23 degrees or 0.401 radian. $\delta$ is the shift to compensate for the tail distance. This tail distance is 5 bases for \textit{E. coli} as mentioned in \textbf{Chapter 5.4.1}.

\begin{equation}
\delta = \text{tail distance} \cdot \frac{2\pi}{3} \quad (5.26)
\end{equation}

The shift in tail distance is required because the species angle was calculated every 3, 6, 9 nucleotides rather than every 2, 5, 8 nucleotides.

The force is calculated as $-\Delta \text{energy} / \Delta \text{distance}$. Hence, the force is calculated as the derivative of \textbf{Equation (5.25)}. Therefore, the force as function of ribosome displacement is:

\begin{equation}
\text{force (RD)} = M \cdot 3 \cdot \pi \cdot \cos \left( \frac{x \cdot \pi}{3} + \Phi - \theta_{sp} - \delta - \frac{\pi}{2} \right) \quad (5.27)
\end{equation}

Alternatively \textbf{Equation (5.27)} can be written as a sine function in which a phase shift of $\frac{\pi}{2}$ is applied:

\begin{equation}
\text{force (RD)} = -M \cdot 3 \cdot \pi \cdot \sin \left( \frac{x \cdot \pi}{3} + \Phi - \theta_{sp} - \delta \right) \quad (5.28)
\end{equation}
Either Equation (5.27) or (5.28) can be used in Equation (5.1) of the probability cycle to compute the force as a function of ribosome displacement if a sinusoidal fitting is used.

**Polynomial Fitting of Discrete Free Energy Signal and Force Calculations**

A polynomial fitting of degree 4 can also be used to convert the discrete free energy signal into a continuous free energy signal. The fitting uses the 5 discrete free energy values: $\Delta G_{n-2}, \Delta G_{n-1}, \Delta G_n, \Delta G_{n+1},$ and $\Delta G_{n+2}$. For simplicity, $\Delta G_{n-2}, \Delta G_{n-1}, \Delta G_n, \Delta G_{n+1}, \Delta G_{n+2},$ are referred to as $A, B, C, D, E$ respectively. A polynomial of degree 4 is generalized using the following equation:

$$y = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3 + a_4 \cdot x^4 \quad (5.29)$$

Therefore, the free energy values $A, B, C, D,$ and $E$ are fitted on a polynomial function which is represented as 5 polynomial equations:

$$A = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3 + a_4 \cdot x^4 \quad (5.30)$$

$$B = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3 + a_4 \cdot x^4 \quad (5.31)$$

$$C = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3 + a_4 \cdot x^4 \quad (5.32)$$

$$D = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3 + a_4 \cdot x^4 \quad (5.33)$$

$$E = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3 + a_4 \cdot x^4 \quad (5.34)$$

where $x$ is in displacement units. The free energy value $A$ is fitted on the polynomial at $x$ equals -4, $B$ is fitted at $x$ equals -2, $C$ is fitted at $x$ equals 0, $D$ is fitted at $x$ equals 2, and $E$ is fitted at $x$ equals 4. Equations (5.35), (5.36), (5.37), (5.38), and (5.39) are derived by
substituting in the \( x \) values for each free energy value \( A, B, C, D, E \); i.e. substitute \(-4\) into \( x \) for Equation (5.30) to get Equation (5.35):

\[
A = a_0 + a_1 \cdot (-4) + a_2 \cdot (-4)^2 + a_3 \cdot (-4)^3 + a_4 \cdot (-4)^4 \tag{5.35}
\]

\[
B = a_0 + a_1 \cdot (-2) + a_2 \cdot (-2)^2 + a_3 \cdot (-2)^3 + a_4 \cdot (-2)^4 \tag{5.36}
\]

\[
C = a_0 + a_1 \cdot (0) + a_2 \cdot (0)^2 + a_3 \cdot (0)^3 + a_4 \cdot (0)^4 \tag{5.37}
\]

\[
D = a_0 + a_1 \cdot 2 + a_2 \cdot (2)^2 + a_3 \cdot (2)^3 + a_4 \cdot (2)^4 \tag{5.38}
\]

\[
E = a_0 + a_1 \cdot (4) + a_2 \cdot (4)^2 + a_3 \cdot (4)^3 + a_4 \cdot (4)^4 \tag{5.39}
\]

Thus, there are 5 equations and 5 unknowns \((a_0, a_1, a_2, a_3, a_4)\). The five coefficients \((a_0, a_1, a_2, a_3, a_4)\) can be solved using the following matrix:

\[
\begin{bmatrix}
1 & x & x^2 & x^3 & x^4 \\
1 & x & x^2 & x^3 & x^4 \\
1 & x & x^2 & x^3 & x^4 \\
1 & x & x^2 & x^3 & x^4 \\
1 & x & x^2 & x^3 & x^4
\end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
a_2 \\
a_3 \\
a_4
\end{bmatrix}
= 
\begin{bmatrix}
A \\
B \\
C \\
D \\
E
\end{bmatrix}
\tag{5.40}
\]

Substituting in values of \( x \) that correspond to the free energy values:

\[
\begin{bmatrix}
1 & -4 & -4^2 & -4^3 & -4^4 \\
1 & -2 & -2^2 & -2^3 & -2^4 \\
1 & 0 & 0^2 & 0^3 & 0^4 \\
1 & 2 & 2^2 & 2^3 & 2^4 \\
1 & 4 & 4^2 & 4^3 & 4^4
\end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
a_2 \\
a_3 \\
a_4
\end{bmatrix}
= 
\begin{bmatrix}
A \\
B \\
C \\
D \\
E
\end{bmatrix}
\tag{5.41}
\]

and using this matrix equation:

\[
a = (X^T X)^{-1} X^T \tag{5.42}
\]
Therefore solving for $a_0, a_1, a_2, a_3, a_4$ in terms of $A, B, C, D, E$:

\[
a_4 = \frac{A+E}{384} - \frac{B+D}{96} + \frac{C}{64} \quad (5.43)
\]

\[
a_3 = \frac{B-D}{48} - \frac{A-E}{96} \quad (5.44)
\]

\[
a_2 = \frac{B+D}{6} - \frac{A+E}{96} - \frac{5\cdot C}{16} \quad (5.45)
\]

\[
a_1 = \frac{A+E}{24} - \frac{B+D}{3} \quad (5.46)
\]

\[
a_0 = C \quad (5.47)
\]

where $A, B, C, D, E$ are free energy values $\Delta G_{n-2}, \Delta G_{n-1}, \Delta G_n, \Delta G_{n+1}, \Delta G_{n+2}$, respectively. After solving for $a_0, a_1, a_2, a_3, a_4$, the energy of ribosome spring as function of displacement is:

\[
\text{Energy}(RD) = a_0 + a_1 \cdot (x - \tau) + a_2 \cdot (x - \tau)^2 + a_3 \cdot (x - \tau)^3 + a_4 \cdot (x - \tau)^4 \quad (5.48)
\]

where $\tau$ is the shift for minimal energy at species angle and tail distance. $\tau$ is calculated as:

\[
\tau = \frac{-3 \cdot \theta_{sp}}{\pi} + 0.5 \quad (5.49)
\]

where $\theta_{sp}$ is the species angle in radians.

The force is then computed as $-\Delta \text{ energy}/\Delta \text{distance}$. This means the force is the derivative of Equation (5.48). Therefore, the force as function of ribosome displacement is:

\[
\text{force}(RD) = -(a_1 + a_2 \cdot 2 \cdot (x - \tau) + a_3 \cdot 3 \cdot (x - \tau)^2 + a_4 \cdot 4 \cdot (x - \tau)^3) \quad (5.50)
\]

Equation (5.50) can be used in Equation (5.1) of the probability cycle to compute the force as a function of ribosome displacement if polynomial fitting is used.
5.4.2 The “Big Picture” of Energetics and Force Calculations

At every codon during translation, the aSD of the 16S rRNA exposed tail binds with the mRNA. The binding distance relative to the P-site determines if the spring is compressed, extended, or relaxed. Bindings that locate the aSD (3’-UCCUCC-5’) 5 bases away from the P-site result in a relaxed spring (see Figure 4.6). In the simulation, a relaxed spring is illustrated as having the minimal energy of the free energy signal close to the center of the A-site (Figure 5.6). The force near minimal energy is close to zero, and therefore, ribosome displacement is small. Bindings that locate the aSD closer than 5 bases away from the P-site compress the spring. Bindings that locate the aSD farther than 5 bases away from the P-site extend the spring. The distance of “5 bases” was determined by the “tail distance” (Chapter 5.4.1) when modeling the frameshift site of prfB. The change in Φ measures the relative binding location of the aSD to the P-site. Changing in the phase angle results in the shifting of the free energy signal relative to the center of the A-site. Therefore, minimal energy may no longer be located near the center of the A-site, resulting in ribosome displacement by the force.
Figure 5.6. Simulation of ribosome of translating lacZ at codon 70. At codon 70, the aSD binds to the mRNA 8 nucleotides from the P-site (Figure 4.5), resulting in a “relaxed spring”. This is modeled by minimal energy located close to the center of the A-site. Therefore the magnitude of the force is small and results in little to no ribosome displacement. The free energies $\Delta G_{n-2}$, $\Delta G_{n-1}$, $\Delta G_n$, $\Delta G_{n+1}$, and $\Delta G_{n+2}$ used to calculate the continuous energy function are displayed on the bottom right.

5.5 View Curve and Aminoacyl-tRNA Abundance Table

The probability cycle (Chapter 5.3) uses the view curve and aminoacyl-tRNA abundance to calculate the probability of picking an aminoacyl-tRNA from the three reading frames. Because there are three reading frames, there are three view curves and three aa-tRNA abundances. The three view curves are VC_{-1}, VC_0, and VC_1. The three aa-tRNA abundances are TA_{-1}, TA_0, TA_{+1}, corresponding to the -1, 0, and +1 reading frames respectively. A ribosome with greater displacement will see less of the codon in the 0 reading frame and more of the codons in the adjacent reading frames. This decreases the chance of the ribosome picking up the aa-tRNA in the 0 reading frame and increases the chance of picking up the aa-tRNA in the adjacent reading frame.
5.5.1 View Curve

The view curve models the degree of the codon in the respective reading frame that the A-site “sees”. For example, when the A-site is in perfect alignment with the 0 reading frame (ribosome displacement is 0), the A-site sees all of the codon in the 0 reading frame and no part of the codon in the -1 or +1 reading frame (Figure 5.7). A displaced ribosome sees both the codon in the 0 reading frame and the adjacent reading frame. For example, when the ribosome displacement is 1, the ribosome sees 50% of the codon in the 0 frame and 50% of the codon in the +1 reading frame. This is modeled by the fact that VC₀ = VC₊₁ when the ribosome displacement is 1 (see Figure 5.8). The view curve is modeled using the function \( \cos x^n \). The current model uses \( \cos x^4 \). The power of the cosine function \( n \) is a parameter that can be adjusted. A greater value of \( n \) makes the view curve narrower, while a smaller \( n \) makes the view curve wider.

The view curve for the codon in the 0 reading frame (VC₀) is modeled using the function \( \cos x^4 \) evaluated from \(-\frac{1}{2} \pi\) to \(\frac{1}{2} \pi\) and superimposed from -2 to 2 displacement units (Equation 5.51). Displacement has units of nucleotides where 2 displacement units is 1 nucleotide and 1 displacement unit is \(\frac{1}{2}\) nucleotide. The view curve for the codon in the +1 reading frame (VC₊₁) is \( \cos x^4 \) evaluated from 0 to \(\pi\) and superimposed from 0 to 4 displacement units (Equation 5.53). The view curve for the codon in the -1 reading frame (VC₋₁) is \( \cos x^4 \) evaluated from \(-\pi\) to 0 and superimposed from -4 to 0 displacement units (Equation 5.51).
\[ VC_{-1} = \cos\left(\frac{x \cdot \pi}{4} + \frac{1}{2} \pi\right)^n \bigg|_{-\pi}^0 \]  
(5.51)

\[ VC_0 = \cos\left(\frac{x \cdot \pi}{4}\right)^n \bigg|_{-0.5\pi}^{0.5\pi} \]  
(5.52)

\[ VC_{+1} = \cos\left(\frac{x \cdot \pi}{4} - \frac{1}{2} \pi\right)^n \bigg|_{0}^{\pi} \]  
(5.53)

\( VC_{-1}, VC_0, VC_{+1} \) is the view curve of the -1, 0, +1 reading frame, respectively. \( n \) is a parameter of the model that can be changed. It is currently set to 4. \( x \) is \( RD_{current} \) (see pseudo code of probability cycle), the current ribosome displacement value.

**Figure 5.7. A-site when ribosome displacement is 0.** The A-site is in perfect alignment with the codon in the 0 reading frame. Therefore, the A-site “sees” all of codon ccg and none of the codons in the +1 or -1 reading frame. This is modeled by the view curve \( VC_0 \) being at its peak and \( VC_{-1} \) and \( VC_{+1} \) being at zero.
5.5.2 Aminoacyl-tRNA Abundance Table

The probability cycle algorithm uses the aminoacyl-tRNA abundance of the codons in the -1, 0, and +1 reading frame \((TA_{-1}, TA_0, TA_{+1})\) in Equations (5.2), (5.3), (5.4) to calculate the probability of picking up an aa-tRNA for that codon. Aminoacyl-tRNAs with high abundance have a higher chance of being picked up by the ribosome, which decreases the amount of time the ribosome waits at that codon. On the other hand, aminoacyl-tRNAs with lower abundance have a lower chance of being picked by the ribosome, which increases the amount of time the ribosome waits for aa-tRNA delivery. When determining aa-tRNA abundance for codons that code for multiple tRNAs, the sum of the respective aa-tRNA abundance are used. tRNA abundance measurements for \(E. coli\) at different growth rates can be found in Dong et al. [9]. Table 5.2 shows the aa-tRNA abundance for each codon derived from tRNA abundance measurements of Dong et al. [9]. When tRNA abundance information
is not known, the genomic codon distribution can be used (Table 5.1) instead. Genomic codon distribution loosely matches intracellular tRNA abundance [9,10,11,12]. tRNA gene copy number can also be used as estimates of aa-tRNA abundance [13,14,15,16]. tRNA abundances are estimates of the actual aa-tRNA abundance. aa-tRNA abundance changes depending on growth conditions and mRNA expression levels; this relates to the “recharge” rates of tRNAs from uncharged to charged states as described in Chapter 3.3.2. The aa-tRNA abundance values (TA) of the respective codon are substituted for $TA_{-1}, TA_0, TA_{+1}$ of Equations (5.2), (5.3), (5.4).
Table 5.1. aa-tRNA abundance per codon using *E. coli* genomic codon frequency table. Genomic codon frequency table can be found in Nakamura et al. [17]. Stop codons are assigned very small values to indicate low abundance and termination of translation. TA stands for tRNA abundance values used in Equations (5.2), (5.3), and (5.4).

<table>
<thead>
<tr>
<th>Codon</th>
<th>aa-tRNA Abundance ((T_A))</th>
<th>Codon</th>
<th>aa-tRNA Abundance ((T_A))</th>
<th>Codon</th>
<th>aa-tRNA Abundance ((T_A))</th>
<th>Codon</th>
<th>aa-tRNA Abundance ((T_A))</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>0.336</td>
<td>caa</td>
<td>0.144</td>
<td>gaa</td>
<td>0.397</td>
<td>uaa</td>
<td>0.000001</td>
</tr>
<tr>
<td>aac</td>
<td>0.256</td>
<td>cac</td>
<td>0.068</td>
<td>gac</td>
<td>0.211</td>
<td>uac</td>
<td>0.162</td>
</tr>
<tr>
<td>aag</td>
<td>0.133</td>
<td>cag</td>
<td>0.31</td>
<td>gag</td>
<td>0.18</td>
<td>uag</td>
<td>0.000001</td>
</tr>
<tr>
<td>aau</td>
<td>0.186</td>
<td>cau</td>
<td>0.16</td>
<td>gau</td>
<td>0.292</td>
<td>uau</td>
<td>0.161</td>
</tr>
<tr>
<td>aca</td>
<td>0.049</td>
<td>cca</td>
<td>0.106</td>
<td>gca</td>
<td>0.201</td>
<td>uca</td>
<td>0.08</td>
</tr>
<tr>
<td>acc</td>
<td>0.245</td>
<td>ccc</td>
<td>0.075</td>
<td>gcc</td>
<td>0.287</td>
<td>ucc</td>
<td>0.076</td>
</tr>
<tr>
<td>acg</td>
<td>0.144</td>
<td>ccg</td>
<td>0.244</td>
<td>gcg</td>
<td>0.339</td>
<td>ucg</td>
<td>0.079</td>
</tr>
<tr>
<td>acu</td>
<td>0.108</td>
<td>ccu</td>
<td>0.05</td>
<td>gcu</td>
<td>0.152</td>
<td>ucu</td>
<td>0.094</td>
</tr>
<tr>
<td>aga</td>
<td>0.03</td>
<td>cga</td>
<td>0.035</td>
<td>gga</td>
<td>0.059</td>
<td>uga</td>
<td>0.000001</td>
</tr>
<tr>
<td>agc</td>
<td>0.171</td>
<td>cgc</td>
<td>0.243</td>
<td>gcc</td>
<td>0.278</td>
<td>ugc</td>
<td>0.064</td>
</tr>
<tr>
<td>agg</td>
<td>0.011</td>
<td>cgg</td>
<td>0.064</td>
<td>ggg</td>
<td>0.14</td>
<td>ugg</td>
<td>0.132</td>
</tr>
<tr>
<td>agu</td>
<td>0.127</td>
<td>cggu</td>
<td>0.2</td>
<td>gggu</td>
<td>0.247</td>
<td>ugu</td>
<td>0.041</td>
</tr>
<tr>
<td>aua</td>
<td>0.042</td>
<td>cua</td>
<td>0.028</td>
<td>gua</td>
<td>0.129</td>
<td>uua</td>
<td>0.128</td>
</tr>
<tr>
<td>auc</td>
<td>0.272</td>
<td>cuc</td>
<td>0.071</td>
<td>guc</td>
<td>0.173</td>
<td>uuc</td>
<td>0.185</td>
</tr>
<tr>
<td>aug</td>
<td>0.268</td>
<td>cug</td>
<td>0.571</td>
<td>gug</td>
<td>0.293</td>
<td>uug</td>
<td>0.156</td>
</tr>
<tr>
<td>auu</td>
<td>0.304</td>
<td>cuu</td>
<td>0.11</td>
<td>guu</td>
<td>0.182</td>
<td>uuu</td>
<td>0.202</td>
</tr>
</tbody>
</table>
Table 5.2. aa-tRNA abundance per codon derived from Dong et al. Aminoacyl-tRNA abundance is derived tRNA abundance per ribosome at 1.7 doubling per hour from Dong et al. [9]. Stop codons are assigned very small values to indicate low abundance and termination of translation. $T_A$ stands for tRNA abundance values used in Equations (5.2), (5.3), and (5.4).

<table>
<thead>
<tr>
<th>Codon</th>
<th>aa-tRNA Abundance ($T_A$)</th>
<th>Codon</th>
<th>aa-tRNA Abundance ($T_A$)</th>
<th>Codon</th>
<th>aa-tRNA Abundance ($T_A$)</th>
<th>Codon</th>
<th>aa-tRNA Abundance ($T_A$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>0.1427</td>
<td>caa</td>
<td>0.0519</td>
<td>gaa</td>
<td>0.3958</td>
<td>uaa</td>
<td>0.000001</td>
</tr>
<tr>
<td>aac</td>
<td>0.0973</td>
<td>cac</td>
<td>0.0519</td>
<td>gac</td>
<td>0.1946</td>
<td>uac</td>
<td>0.1622</td>
</tr>
<tr>
<td>aag</td>
<td>0.1427</td>
<td>cag</td>
<td>0.0843</td>
<td>gag</td>
<td>0.3958</td>
<td>uag</td>
<td>0.000001</td>
</tr>
<tr>
<td>aau</td>
<td>0.0973</td>
<td>cau</td>
<td>0.0519</td>
<td>gau</td>
<td>0.1946</td>
<td>uau</td>
<td>0.1622</td>
</tr>
<tr>
<td>aca</td>
<td>0.0779</td>
<td>cca</td>
<td>0.0389</td>
<td>gca</td>
<td>0.2855</td>
<td>uca</td>
<td>0.1103</td>
</tr>
<tr>
<td>acc</td>
<td>0.0843</td>
<td>ccc</td>
<td>0.0649</td>
<td>gcc</td>
<td>0.0519</td>
<td>ucc</td>
<td>0.0584</td>
</tr>
<tr>
<td>acg</td>
<td>0.1233</td>
<td>cgc</td>
<td>0.0843</td>
<td>ggc</td>
<td>0.2855</td>
<td>ucg</td>
<td>0.1298</td>
</tr>
<tr>
<td>acu</td>
<td>0.1622</td>
<td>ccu</td>
<td>0.1038</td>
<td>gcu</td>
<td>0.2855</td>
<td>ucu</td>
<td>0.1687</td>
</tr>
<tr>
<td>aga</td>
<td>0.0519</td>
<td>cga</td>
<td>0.3893</td>
<td>gga</td>
<td>0.1090</td>
<td>uga</td>
<td>0.000001</td>
</tr>
<tr>
<td>agc</td>
<td>0.0908</td>
<td>cgc</td>
<td>0.3893</td>
<td>ggc</td>
<td>0.3244</td>
<td>ugc</td>
<td>0.1168</td>
</tr>
<tr>
<td>agg</td>
<td>0.0908</td>
<td>cgg</td>
<td>0.0389</td>
<td>ggg</td>
<td>0.1817</td>
<td>ugg</td>
<td>0.0649</td>
</tr>
<tr>
<td>agu</td>
<td>0.0908</td>
<td>cgu</td>
<td>0.3893</td>
<td>ggu</td>
<td>0.3244</td>
<td>ugu</td>
<td>0.1168</td>
</tr>
<tr>
<td>aua</td>
<td>0.0159</td>
<td>cua</td>
<td>0.0519</td>
<td>gua</td>
<td>0.3114</td>
<td>uua</td>
<td>0.0584</td>
</tr>
<tr>
<td>auc</td>
<td>0.2891</td>
<td>cuc</td>
<td>0.0779</td>
<td>guc</td>
<td>0.1038</td>
<td>uuc</td>
<td>0.0779</td>
</tr>
<tr>
<td>aug</td>
<td>0.0649</td>
<td>cug</td>
<td>0.4022</td>
<td>gug</td>
<td>0.3114</td>
<td>uug</td>
<td>0.2141</td>
</tr>
<tr>
<td>auu</td>
<td>0.2891</td>
<td>cuu</td>
<td>0.1298</td>
<td>guu</td>
<td>0.4152</td>
<td>uuu</td>
<td>0.0779</td>
</tr>
</tbody>
</table>

5.6 Gene Optimization Algorithm

Genes were optimized using the concepts and methods described in Chapter 4.4. The optimization chooses the codons that code for the most abundant aa-tRNA while maintaining a phase angle that is close to the species angle. The algorithm optimizes a gene by redesigning the entire coding region from the second codon (after AUG) to the stop codon. Six codons were optimized concurrently. The optimization algorithm starts with the first 6 codons beginning at codon 2, optimizes those 6 codons, then moves to the next 6 codons until the end of the gene is reached. Codons where changed using synonymous codons while conserving the amino acid sequence.
For each set of 6 codons, a list of all the different combinations for those codons was determined. A “total value” for each permutation of the 6 codons was calculated. The total value is computed by taking the product of each codon’s “value”. The codon’s value can be defined as either the aa-tRNA abundance or a user “predefined” value. If the codon’s value is the aa-tRNA abundance, then the algorithm will almost always use the same codon in the set of synonymous codons. Using a predefined value is helpful in disbursing the codon usage and thus spreading the usage of aa-tRNA. For example, proline codons CCU and CCG can be assigned the same value while proline codons CCC and CCA can be assigned a lower value. This will distribute the usage of CCU and CCG codons evenly while limiting the usage of CCC and CCA codons. Codon preferences are shown in Table 4.1.

The permutations were then ranked from highest to lowest by “total value”. The algorithm then takes the highest ranking permutation and computes the phase angle of the 6 codons. If the phase angle of the permutation is within a predefined “threshold” of the species angle, then that permutation is chosen as the “optimized” permutation, and the algorithm moves to the next 6 codons in the gene. However, if the phase angle is not within the predefined threshold of the species angle, the algorithm chooses the next permutation in the list until a permutation is found that is within the range of the species angle. If no permutation within the threshold of the species angle is found, then the highest ranking permutation is used as the “optimized” permutation. The calculation of the species angle ($\theta_{sp}$) and “phase angle” of the partially optimized gene and its 6-codon permutation is described in [1,18] and Appendix B. The phase angle is computed by calculating $\Phi_k$ for the partially optimized sequence (including the new 6 codon permutation); calculation of $\Phi_k$ is
published in [1]. After finding the optimized permutation, the algorithm moves to the next 6 codons and repeats the process until the end of the gene. The resulting product is an optimized gene that uses the codons that code for the most abundant aa-tRNAs or the sets of codons that code for multiple abundant aa-tRNAs, all while maintaining the phase angle close to the species angle.

Optimized gene sequence is then processed by the model to determine ribosome displacement, translation bottlenecks (BNI), and total wait cycle. These parameters are compared to that of the wildtype gene or the parameters of codon bias-optimized gene to determine if the gene can be further optimized; if ribosome displacement is not close enough to zero, then BNI or total wait can still be lowered. The “most optimized” sequence can be determined empirically by (1) varying the “acceptable threshold” of the gene’s phase angle compared to the species’ angle and also by (2) changing preferred codon(s) through varying the codon’s “value”.

The aforementioned gene optimization algorithm is best described as a heuristic algorithm rather than a classical optimization algorithm with defined optimal conditions. However, the algorithm is deterministic rather than stochastic because it outputs the same sequence given the same inputs rather than outputting different sequences each time. The algorithm improves the translation efficiency of genes based on using the most abundant aa-tRNA (through codon usage) while also reducing ribosome displacement and force. By using these principles, ribosome wait times and bottlenecks are lowered throughout translation elongation. This results in a “more optimal” gene rather than the “most optimal” gene
because the most optimal conditions are not known. In theory, there can be multiple “optimal” gene variants that have the same displacement, wait times, and BNI.

5.7 Conclusions

Algorithms and mathematical equations were developed to model the ribosome “spring” action that results in ribosome displacement. At every codon, the “probability cycle” displaces the ribosome a small step size during each “cycle” until the ribosome “picks up” an aminoacyl-tRNA. The degree of ribosome displacement is a function of the force and how long the ribosome has to wait for the delivery of the next amino-acyl tRNA, which is related to aa-tRNA abundance. The force is calculated by converting the discrete free energy signal into a continuous free energy signal (using sinusoidal fitting or polynomial fitting) and taking the derivative of this continuous free energy signal. The amount of time the ribosome spends waiting for the delivery of the next aminoacyl-tRNA is modeled by multiple probabilities convolved with the “view curve” and aminoacyl-tRNA abundance. The probability cycle convolves the energetic binding of the “springy” 16S rRNA exposed tail (the force exerted on the ribosome from this spring) with the aminoacyl-tRNA abundance to output ribosome displacement and ribosome wait time at every codon. See Figure 5.9 for a graphical illustration of the algorithms. Parameters considered in the algorithm include the step size constant delta T (dT), power of the cosine function (n) for the “view curve,” the “tail distance,” normalization constant (NF) for the probabilities, and the aminoacyl-tRNA abundance information. The accuracy of the model depends on all these parameters. A gene optimization algorithm was also developed to efficiently optimize genes, which is more efficient and accurate than optimizing genes manually. This algorithm designs genes from
scratch by optimizing 6 codons at a time starting with the second codon. Optimization of those 6 codons is a “tradeoff” between choosing codons that closely match the species angle and choosing codons that confer with the best aa-tRNA usage. The aforementioned algorithms and mathematics were incorporated in a fully implemented software package called RiboScan™; this software package will be available as webserver [19]. Outputs of RiboScan™ includes polar plots (see Appendix B), ribosome displacement plots, ribosome wait time plots, translation bottleneck plots, and simulations of single ribosome dynamics during translation elongation.

**Figure 5.9. Graphical illustration of the algorithms.** The binding between the mRNA and 16S rRNA “exposed tail” is modeled using a continuous free energy signal from which the force can be calculated. The probability cycle uses probabilities $P_0, P_1, P_n$, the view curve, aa-tRNA abundance, and the force to displace the ribosome while it waits for the delivery of the next aa-tRNA. After the aa-tRNA has been delivered, the main algorithm moves the ribosome three bases downstream, and the probability cycle begins again.
5.8 References


CHAPTER 6

Case Studies – Prediction and Optimization

6.1 Introduction

The model’s prediction and optimization capabilities were assessed by expressing five model genes: \textit{gst} (glutathione-S-transferase), \textit{pf0132} (alpha-glucosidase), \textit{clju_c11880} (alcohol dehydrogenase), \textit{bifl} (beta-galactosidase), and \textit{rt8_0542} (endoglucanase and exoglucanase) in \textit{Escherichia coli}. The optimization power of the model was also compared to that of codon bias optimization algorithms as described in \textbf{Chapter 3.3.1}. Multiple mRNA variants of GST were also designed and expressed to further verify the model’s prediction as compared to that of codon bias’s prediction. The codon adaptation index (CAI) by Sharpe et al. [1] was used to measure a gene’s codon bias. Measured protein yield (GST activity) was correlated with predicted protein yield for both the codon adaptation index (CAI) and the model’s index (BNI). The rationale behind comparing the correlations was to demonstrate that determinants of protein yield are not limited to just codon bias but rather encompass a convolution between ribosome displacement, force, and tRNA usage. This convolution leads to new ribosome wait time parameters and translation bottlenecks (BNI) that allow for a superior predictor of protein yield. Description of ribosome wait time parameters and translation bottlenecks is described in \textbf{Chapter 4.3}.

Gene optimization was performed using the concepts described in \textbf{Chapter 4.4}. The algorithm used for optimization is described in \textbf{Chapter 5.6}. Polar plots (see \textbf{Appendix B}), ribosome displacement plots, ribosome wait time plots, and translation bottleneck plots were
used as analysis tools. Detailed descriptions of the materials and methods used in cloning, expressing, and quantifying protein yield for these genes are described in Appendix E. Genes were synthesized by either gene synthesis company Genewiz, Inc. or Genscript, Inc. The genes’ DNA sequences are provided in Appendix D.

6.2 Gene Descriptions

Gene \(gst\) came from \(Schistosoma japonicum\), a parasite commonly called the liver fluke. GST is a relatively small protein with a molecular weight of 26 kilo-Daltons (218 amino acids, 654 nucleotides). \(gst\) is highly expressed in \(E. coli\) and is soluble in the cytoplasm. GST was chosen as a candidate for model optimization because it is a well-studied eukaryotic protein that has a commercially-available assay for quantifying protein yield. Even though wildtype \(gst\) is highly expressed in \(E. coli\), the model showed that ribosome displacement can be further minimized to improve expression level [8]. The source of wildtype \(gst\) used for prediction and optimization came from the pET-41a+ plasmid (Novagen, Inc.). \(gst\) from pET-41a+ is identical to that cloned from \(Schistosoma japonicum\) [7].

\(pf0132\) (GenBank: NP_577861.2) is an alpha-glucosidase from archaeon \(Pyrococcus furiosus\) DSM 3638 that was characterized by Costantino et al. [2]. \(Pyrococcus furiosus\) is a hyperthermophile that lives at 100 °C. \(pf0132\) was chosen because it belongs to a domain of life (Archaea) different from that of prokaryotic \(E. coli\). Furthermore, \(pf0132\) showed high ribosome displacement which can be minimized to increase protein yield. The molecular weight of PF0132 is 55 kilo-Daltons and contains 489 amino acids (1,467 nucleotides).
*clju_11880* (GenBank: CLJU_c11880) is an alcohol dehydrogenase that comes from *Clostrium ljungdahlii* DSM 13528. *Clostrium ljungdahlii* is an anaerobic, enteric bacterium. *clju_11880* was chosen because it comes from a genome with high AT content compared to the genome of *E. coli*. Wildtype *clju_11880* also showed high displacement which could be minimized to increase protein yield. The molecular weight of CLJU_c11880 is 42 kilo-Daltons, and it contains 380 amino acids (1,140 nucleotides).

*bif1* (GenBank: AJ272131) is a beta-galactosidase from *Bifidobacterium bifidum* DSM 20215 [3]. *Bifidobacterium bifidum* is a gram-positive, probiotic, anaerobic bacterium. *bif1* was chosen because it is an exceptionally long gene, and the host organism, *Bifidobacterium bifidum* DSM20215, has a genome with high GC content compared to *E. coli*. According to the model, optimizing long genes should produce more substantial improvements in protein yield (*in vivo*) than compared to codon bias optimization. Optimization of *bif1* gene using the model showed a greater increase (*in silico*) than wildtype and codon bias-optimized variants. The molecular weight of BIF1 is 112 kilo-Daltons, and it contains 1,020 amino acids (3,060 nucleotides).

*rt8_0542* is both an endoglucanase (containing the GH9 domain) and exoglucanase (containing the GH48 domain) from an unpublished *Caldicellulosiruptor* species. *Caldicellulosiruptor* is a thermophilic, anaerobic, gram-positive bacterium. Although wildtype *rt8_0542* has not been successfully expressed in *E. coli*, a codon bias-optimized variant has produced working protein. Both wildtype and codon bias-optimized *rt8_0542* were analyzed using the model to determine key differences. *rt8_0542* was chosen to gain insight into the dynamics of codon bias optimization and to better compare codon bias
optimization to model optimization in silico. The molecular weight of rt8_0542 is 210 kilo-Daltons, and it contains 1,875 amino acids (5,625 nucleotides).

### 6.3 Comparison of Protein Yield Prediction Indices

Multiple mRNA variants of GST were designed and expressed to assess the model’s capabilities in predicting protein yields. The rationale for this experiment was to determine a feasible index from the model for protein yield prediction. The bottleneck index (BNI) was the most feasible predictor for protein yield (see Chapter 4.3 and Chapter 5.3.2). The bottleneck index is calculated as the function of displacement, aminoacyl-tRNA usage, force, and ribosome wait time. This index identifies clusters of high ribosome wait time and indicates the location of translational bottlenecks on the mRNA.

The BNI’s accuracy was confirmed by correlating BNI predictions with GST activity from nine mRNA variants. The codon adaptation index (CAI) was also evaluated in the same manner. CAI using *E. coli* genomic coding sequences as references [15,16] and CAI (HEG) using highly expressed genes as references [6] were both tested. Total Wait Time from the model (see Chapter 4.3) was also correlated to GST activity. The resulting correlation of the four indices was compared to assess their prediction accuracy. The nine *gst* variants were referred to as W, M, VM, 16, 23, 80, 82, 90D, 0D. The first 90 bases of all *gst* variants were unchanged from the wildtype gene (except for variant M) to normalize protein yield deviations caused by initiation. Design rationales and graphical analyses of *gst* variants are provided in Chapter 6.3.1. Experimental procedures and laboratory results can be found in Chapter 6.3.2. Comparisons of the correlation for various protein yield prediction indices can be found in Chapter 6.3.3, and sensitivity analysis for different window sizes is in
Chapter 6.3.4. DNA sequences for \textit{gst} variants used for comparing prediction indices can be found in Appendix D.1. Detailed experimental procedures can be found in Appendix E.1 through E.4.

6.3.1 Design Rationale and Graphical Analysis of \textit{gst} Variants

\textit{gst} variant M was “manually” optimized by minimizing displacement throughout the gene (see Chapter 6.3.1.2). This optimization was done because displacement was initially thought to be the major predictor of protein yield; however, after various predictions, ribosome wait times and translation bottlenecks were found to be better indicators of protein yield. Ribosome wait times and translation bottlenecks are a function of ribosome displacement, aminoacyl-tRNA usage, and force from the ribosome spring. Variant VM (see Chapter 6.3.1.3) is similar to variant M except that the first 90 bases in variant VM are that of the wildtype. Variant 80 (see Chapter 6.3.1.6) was codon bias-optimized using the “one amino acid, one codon” algorithm. A comparison between model-optimized (variant VM), codon bias-optimized (variant 80), and wildtype was published in Vu et al. [7]. Variant 82, like variant 80, was chosen for its high CAI, but interestingly, the phase angle of variant 82 was closer to the species angle, and the displacement was closer to zero throughout the entire gene (see Chapter 6.3.1.7). One would expect this variant to produce more protein than variant 80. Variant 16 was designed using a random mRNA sequence generator. Variant 16 was chosen because it has a low CAI, but its displacement remained close to zero (see Chapter 6.3.1.4). Expression of variant 16 was performed to assess whether ribosome displacement was a better predictor of protein yield than codon bias or if Total Wait Time and translation bottlenecks (BNI) might be better predictors. Variant 23 was also designed
using a random mRNA generator. Variant 23 was chosen because of its low CAI and its strange polar plot behavior at zero degrees (see Chapter 6.3.1.5). Variant 90D was designed using the model to have an average phase angle at around 90 degrees (see Chapter 6.3.1.8); this phase angle is out of the “normal” range for *E. coli* (see Appendix C). Therefore, the ribosome is displaced from codon 40 to the stop codon with maximum displacement of around +1 units; this corresponds to a misalignment of half a nucleotide in the +1 reading frame. Variant 0D was designed using the model to have an average phase angle at around 0 degrees (see Chapter 6.3.1.9); this phase angle is on the boundary but within the “normal” range of *E. coli*. Even though the phase angle is “on the border,” the ribosome displacement remains close to zero. Both 90D and 0D variants have the same codon bias as determined by the CAI but different displacement, total number of wait cycles, and BNIs (see Table 6.1). According to the model’s prediction, variant 90D should produce lower protein yield than variant 0D. The polar plots, ribosome displacement plots, ribosome wait time plots, and translation bottleneck plots of *gst* variants are provided in Chapters 6.3.1.1 through 6.3.1.9.
6.3.1.1 Wildtype Graphical Plots

Wildtype *gst* showed an average phase angle of -64 degrees (see Figure 6.1). *E. coli*’s species angle is -23 degrees. Genes that have an average phase angle that deviates from *E. coli*’s species angle exert extra force on the ribosome during translation, which is caused by the compression or extension of the ribosome spring (see Chapter 4). This extra force can displace the ribosome when aminoacyl-tRNAs with low abundances are used (see ribosome displacement in Chapter 4.2.2). In the case of wildtype *gst*, the ribosome is displaced, as shown in Figure 6.2.

![Figure 6.1. Polar plot of *gst* wildtype. The average phase angle of the wildtype *gst* is -63 degrees.](image)

The ribosome is displaced from codon 50 to codon 224 (see Figure 6.2). The maximum displacement is +0.75 at around codon 160 (ribosome displacement of 0.375 nucleotides towards the +1 reading frame). The ribosome wait time plot shows clustering of high wait times around codons 65 and 200 with a Total Wait Time of 1694 cycles (see Figure 6.3).
The clustering leads to bottlenecks at around codon 65 and codon 200 with the maximum bottleneck (BNI of 228) at codon 199 (see Figure 6.4).

**Figure 6.2.** Ribosome displacement plot of *gst* wildtype. This plot shows the displacement of the ribosome during translation wildtype *gst* gene.

**Figure 6.3.** Ribosome wait time plot of *gst* wildtype. Total Wait Time for translation of the entire *gst* wildtype gene is 1694 cycles.

**Figure 6.4.** Translation bottleneck plot of *gst* wildtype. The BNI for *gst* wildtype is 228 at codon 199.
6.3.1.2 Variant M Graphical Plots

\textit{gst} variant M was manually optimized to minimize displacement (see Figure 6.6). Variant M showed an average phase angle of -57 degrees (see Figure 6.5). Even though the gene’s average phase angle differs from the \textit{E. coli}’s species angle and exerts extra force on the ribosome, displacement is mitigated because aminoacyl-tRNAs with high abundance are used at key codon positions.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{gst_variant_M.png}
\caption{Polar plot of \textit{gst} variant M. The average phase angle of the variant M gene is -57 degrees.}
\end{figure}

However, even after minimizing displacement, the ribosome wait time plot shows clustering of high wait times around codon 65, codon 110, and codon 200 (see Figure 6.7); the Total Wait Time is lowered to 1492 cycles down from 1696 cycles in the wildtype. However, because of high wait-time clustering, bottlenecks exists at codons 65, 100, and 200, as shown in Figure 6.8. The maximum bottleneck (BNI of 218) is computed at codon 67.
Figure 6.6. Ribosome displacement plot of \textit{gst} variant M. This plot shows the displacement of the ribosome during translation of the \textit{gst} variant M.

Figure 6.7. Ribosome wait time plot of \textit{gst} variant M. The Total Wait Time for variant M is 1492 cycles.

Figure 6.8. Translation bottleneck plot of \textit{gst} variant M. The BNI for \textit{gst} variant M is 218 at codon 67.
6.3.1.3 Variant VM Graphical Plots

*gst* variant VM is similar to variant M except that the first 90 bases are directly copied from the *gst* wildtype. Variant VM showed an average phase angle of -59 degrees (see Figure 6.9).

![Polar plot of *gst* variant VM](image)

**Figure 6.9. Polar plot of *gst* variant VM.** The average phase angle of variant VM is -59 degrees.

Ribosome displacement (see Figure 6.10) is minimized almost identical to that of variant M. However, even though ribosome displacement is minimized, the ribosome wait time plot shows clusters of high wait times around codon 65, codon 110, and codon 200 (see Figure 6.11); the Total Wait Time is lowered to 1494 cycles as compared to wildtype. Because of high wait-time clustering, bottlenecks exist at around codon 65, codon 100, and codon 200, as shown in Figure 6.12. The maximum bottleneck (BNI of 218) is at codon 67.
Figure 6.10. Ribosome displacement plot of $gst$ variant VM. The ribosome displacement of $gst$ variant VM is remarkably similar to that of variant M.

Figure 6.11. Ribosome wait time plot of $gst$ variant VM. The Total Wait Time is 1494 cycles.

Figure 6.12. Translation bottleneck plot of $gst$ variant VM. The BNI for $gst$ variant VM is 218 at codon 67.
6.3.1.4 Variant 16 Graphical Plots

*gst* variant 16 was designed using a random mRNA sequence generator. Variant 16 was chosen because it has a low CAI, yet its displacement remained close to zero. Variant 16 was tested to determine whether ribosome displacement is a better indicator of protein yield than codon bias or if Total Wait Time and translation bottlenecks might be better predictors. Variant 16 showed an average phase angle of -43 degrees (see Figure 6.13).

![Figure 6.13](image.png)

**Figure 6.13. Polar plot of *gst* variant 16.** The average phase angle of variant 16 is -43 degrees.

Ribosome displacement was minimized throughout elongation (see Figure 6.14). The ribosome wait time plot shows clustering of high wait times around codon 65 and codon 180 (see Figure 6.15); the Total Wait Time is increased to 2012 cycles compared to 1696 cycles in the wildtype. Because of high wait-time clustering, bottlenecks exist at around codon 59 and codon 180, as shown in Figure 6.16. The maximum bottleneck (BNI of 296) is at codon 59.
**Figure 6.14.** Ribosome displacement plot of \(\text{gst}\) variant 16. This plot shows that the ribosome displacement remains close to zero during translation of variant 16.

**Figure 6.15.** Ribosome wait time plot of \(\text{gst}\) variant 16. The Total Wait Time for translation of variant 16 is 2012 cycles.

**Figure 6.16.** Translation bottleneck plot of \(\text{gst}\) variant 16. The BNI for \(\text{gst}\) variant 16 is 296 at codon 59.
6.3.1.5 Variant 23 Graphical Plots

gst variant 23 was also designed using a random mRNA sequence generator. Variant 23 was chosen because it has a low CAI and interesting polar plot behavior at zero degrees (see Figure 6.17). Variant 23 showed an average phase angle of -18 degrees, as shown in Figure 6.17.

![Polar plot of gst variant 23.](image)

Figure 6.17. Polar plot of gst variant 23. The average phase angle of variant 23 is -18 degrees, and this plot shows the strange activity at 0 degrees.

Ribosome displacement stayed minimized throughout elongation with the exception of codons 80 to 110 (see Figure 6.18), in which ribosome displacement decreases to around -0.70. The ribosome wait time plot shows clustering of high wait times around codon 60, codons 90 to 110, and codon 180 (see Figure 6.19); the Total Wait Time is increased to 1788 cycles compared to 1696 cycles in the wildtype. Because of high wait-time clustering, bottlenecks exist around codon 60, codon 90, and codon 185, as shown in Figure 6.20. The maximum bottleneck (BNI of 261) is at codon 185.
Figure 6.18. Ribosome displacement plot of \textit{gst} variant 23. Ribosome displacement decreases to -0.7 around codons 80 to 110.

Figure 6.19. Ribosome wait time plot of \textit{gst} variant 23. The Total Wait Time during translation of variant 23 is 1788 cycles.

Figure 6.20. Translation bottleneck plot of \textit{gst} variant 23. The BNI for \textit{gst} variant 23 is 261 at codon 185.
6.3.1.6 Variant 80 Graphical Plots

gst variant 80 was codon bias-optimized using the “one amino acid one codon” algorithm (see Chapter 3.3.1). Variant 80 showed an average phase angle of -59 degrees, as shown in Figure 6.21.

![Polar plot of gst variant 80](image)

Figure 6.21. Polar plot of gst variant 80. The average phase angle of variant 80 is -59 degrees.

The ribosome begins being displaced from codon 40 to codon 175 (see Figure 6.22). The maximum displacement is +0.50 around codon 85 to 125, which corresponds to an A-site misalignment of 0.25 nucleotides toward the +1 reading frame. The ribosome wait time plot shows clustering of high wait times around codon 3 (see Figure 6.23); the Total Wait Time is decreased to 1148 cycles compared to 1696 cycles in the wildtype. Because of high wait-time clustering, bottlenecks exist around codon 2, as shown in Figure 6.24. The maximum bottleneck (BNI of 160) is at codon 2.
Figure 6.22. Ribosome displacement plot of \( \text{gst} \) variant 80. The ribosome displacement begins increasing around codon 40 and remains high until codon 175.

Figure 6.23. Ribosome wait time plot of \( \text{gst} \) variant 80. The Total Wait Time during translation of variant 80 is 1148 cycles.

Figure 6.24. Translation bottleneck plot of \( \text{gst} \) variant 80. The BNI for \( \text{gst} \) variant 80 is 160 at codon 2.
6.3.1.7 Variant 82 Graphical Plots

*gst* variant 82, like variant 80, was chosen because of its high CAI; however, the phase angle of variant 82 was closer to the species angle of *E. coli*, and its displacement was closer to zero throughout the enter gene. One would expect this variant to produce higher protein yield than variant 80. Variant 82 showed an average phase angle of -51 degrees, as shown in Figure 6.25.

![Polar plot of gst variant 82](image)

**Figure 6.25. Polar plot of *gst* variant 82.** The average phase angle of variant 82 is -51 degrees.

The ribosome showed minimal displacement throughout elongation (see Figure 6.26). The ribosome wait time plot shows clustering of high wait times around codon 3 (see Figure 6.27); the Total Wait Time is decreased to 1148 cycles compared to 1696 cycles in the wildtype. Because of high wait-time clustering, bottlenecks exist at around codon 2, as shown in Figure 6.28. The maximum bottleneck (BNI of 160) is at codon 2.
Figure 6.26. Ribosome displacement plot of \textit{gst} variant 82. Variant 82 showed minimal displacement throughout the entire gene.

Figure 6.27. Ribosome wait time plot of \textit{gst} variant 82. The Total Wait Time for variant 82 is 1148 cycles.

Figure 6.28. Translation bottleneck plot of \textit{gst} variant 82. The BNI of \textit{gst} variant 82 is 160 at codon 2.
6.3.1.8 Variant 90D Graphical Plots

Using the model, \( \text{gst} \) variant 90D was designed to have a phase angle at around 90 degrees, as shown in Figure 6.29. Variant 90D showed an average phase angle of -87 degrees.

**Figure 6.29. Polar plot of \( \text{gst} \) variant 90D.** The average phase angle of variant 90D is -87 degrees.

Because the average phase angle of variant 90D is out of the “normal” phase angle range for \( E. \text{coli} \) (see Appendix C), the ribosome is displaced from around codon 40 to codon 224 (see Figure 6.30). The ribosome wait time plot shows clustering of very high wait times around codon 110 (see Figure 6.31); the Total Wait Time is increased to 1925 cycles compared to 1696 cycles in the wildtype. Because of high wait-time clustering, bottlenecks exist around codon 100, as shown in Figure 6.32. The maximum bottleneck (BNI of 288) is at codon 100.
Figure 6.30. Ribosome displacement plot of \textit{gst} variant 90D. The ribosome displacement continues to increase throughout translation of variant 90D, largely due to the phase angle differing from the species angle of \textit{E. coli}.

Figure 6.31. Ribosome wait time plot of \textit{gst} variant 90D. The Total Wait Time is 1925 cycles for translation of variant 90D.

Figure 6.32. Translation bottleneck plot of \textit{gst} variant 90D. The BNI of \textit{gst} variant 90D is 288 at codon 100.
6.3.1.9 Variant 0D Graphical Plots

Using the model, \( \text{gst} \) variant 0D was designed to have a phase angle at around 0 degrees. Variant 0D showed an exact average phase angle of -3 degrees, as shown in Figure 6.33. This phase angle is on the boundary of the “normal” phase angle range of \( E. \ coli \) (see Appendix C).

![Figure 6.33. Polar plot of \( \text{gst} \) variant 0D. Average phase for \( \text{gst} \) variant 0D is -3 degrees.]

Even though the phase angle is “on the border,” the ribosome displacement stays close to zero and deviates about 0.5 displacement units (see Figure 6.34). The ribosome wait time plot shows clustering of high wait times around codon 180 and 200 (see Figure 6.35); the Total Wait Time is decreased to 1527 cycles compared to 1696 cycles in the wildtype. Bottlenecks exist around codon 185, as shown in Figure 6.36. The maximum bottleneck (BNI of 205) is at codon 185.
Figure 6.34. Ribosome displacement plot of \textit{gst} variant 0D. The ribosome displacement of variant 0D deviates less than 0.5 units throughout the entire gene.

Figure 6.35. Ribosome wait time plot of \textit{gst} variant 0D. The Total Wait Time for variant 0D translation is 1527 cycles.

Figure 6.36. Translation bottleneck plot of \textit{gst} variant 0D. The BNI of \textit{gst} variant 0D is 205 at codon 185.
6.3.2 Experimental Procedures and Laboratory Results

All *gst* variants were synthesized and cloned into pBAD/Myc-His inducible plasmid available from Thermo Fisher Scientific, Inc. *gst* genes were synthesized and cloned by Genewiz, Inc. or GeneScript, Inc. (see Appendix E.1). The plasmids containing gene insert were transformed into chemically competent TOP10 *E. coli* cells (available from Thermo Fisher Scientific, Inc.) and expressed at 0.02% w/v (0.2 mg/ml) arabinose for 3 hours (see Appendix E.2 and E.3). The P_{bad} inducible promoter [11] was used to normalize the transcript levels. One milliliter of induced cells were harvested, pelleted, and lysed with BugBuster and Lysonase (both available from EMD Millipore, Inc.). GST activity was quantified using GST assay kit (E.C. 2.5.1.18) available from Sigma Aldrich, Inc. (see Appendix E.4). One unit of GST is defined as the amount of GST enzyme that produces 1 umol of GS-DNB conjugate per minute. GST specific activity was normalized to total protein activity using the Pierce™ BCA Protein Assay Kit available from Thermo Fisher Scientific, Inc. Total protein normalization was performed to standardize for deviations in the number of cells collected. Units for normalized GST activity were defined as units of GST per A_{562nm} of total protein where one A_{562nm} equals 18 mg of total protein. The calculations for units of normalized GST activity can be found in Appendix E.4. Two independent inductions were conducted to test for replicability. Three samples were collected from each induction for a total of six samples assayed. Detailed experimental procedures can be found in Appendices E.1 through E.4. Normalized GST activity of each *gst* variant and its protein yield prediction indices are shown in Table 6.1.
6.3.3 Correlating GST Activity with Predictive Indices

Normalized GST activity levels were correlated with the following predictive indices for protein yield: CAI using *E. coli* genomic coding sequences, CAI using highly expressed genes (HEG), Total Wait Time, and BNI. For comparison purposes, normalized GST activity and values for the various predictive indices for the corresponding *gst* variants can be found in Table 6.1. Correlations were computed by fitting a best-fit line using least-square regression and calculating the resulting r (Pearson correlation coefficient) and r-squared (coefficient of determination) values. BNI had the best correlation followed by Total Wait Time, CAI, and then CAI (HEG). BNI correlated with an r = -0.96 and r-squared = 0.92 (see Figure 6.37); Total Wait Time correlated with an r = -0.89 and r-squared = 0.80 (see Figure 6.38); CAI correlated with an r = 0.75 and r-squared = 0.57 (see Figure 6.39); and CAI (HEG) correlated with an r = 0.68 and r-squared = 0.47 (see Figure 6.40).

Table 6.1. Normalized GST activity and protein yield predictive indices for *gst* variants. Normalized GST activity, CAI, CAI (HEG), Total Wait Time, and BNI values for each *gst* variant are listed. CAI is the codon adaptation index using *E. coli* genomic coding sequences [9] as reference genes. CAI (HEG) is the codon adaptation index using highly expressed genes from [6] as reference genes. Total Wait Time and bottleneck index (BNI) are protein yield prediction indices from the model. *gst* variants are sorted from lowest to highest normalized GST activity.

<table>
<thead>
<tr>
<th><em>gst</em> Variants</th>
<th>Normalized GST Activity</th>
<th>CAI</th>
<th>CAI (HEG)</th>
<th>Total Wait Time (Cycles)</th>
<th>BNI</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>0.413 ± 0.023</td>
<td>0.41</td>
<td>0.13</td>
<td>2012</td>
<td>296</td>
</tr>
<tr>
<td>90D</td>
<td>0.422 ± 0.038</td>
<td>0.54</td>
<td>0.31</td>
<td>1925</td>
<td>288</td>
</tr>
<tr>
<td>23</td>
<td>0.493 ± 0.029</td>
<td>0.44</td>
<td>0.20</td>
<td>1788</td>
<td>261</td>
</tr>
<tr>
<td>Wildtype</td>
<td>0.540 ± 0.033</td>
<td>0.58</td>
<td>0.34</td>
<td>1694</td>
<td>228</td>
</tr>
<tr>
<td>0D</td>
<td>0.555 ± 0.027</td>
<td>0.56</td>
<td>0.33</td>
<td>1527</td>
<td>205</td>
</tr>
<tr>
<td>80</td>
<td>0.583 ± 0.032</td>
<td>0.75</td>
<td>0.79</td>
<td>1148</td>
<td>160²</td>
</tr>
<tr>
<td>VM</td>
<td>0.590 ± 0.014</td>
<td>0.61</td>
<td>0.37</td>
<td>1494</td>
<td>218</td>
</tr>
<tr>
<td>M</td>
<td>0.594 ± 0.024</td>
<td>0.62</td>
<td>0.39</td>
<td>1492</td>
<td>218</td>
</tr>
<tr>
<td>82</td>
<td>0.607 ± 0.037</td>
<td>0.78</td>
<td>0.80</td>
<td>1148</td>
<td>160²</td>
</tr>
</tbody>
</table>

± indicates one standard deviation. Normalized GST activity is measured as units of GST per A₅₆₂nm of total protein where one A₅₆₂nm equals 18 mg of total protein; one unit of GST is defined as the amount of GST enzyme producing 1 umol of GS-DNB conjugate per minute. BNI values before adding wait time value to initiation. BNI values after adding wait time value to initiation.
In order for BNI to correlate with r-squared = 0.92, a ribosome wait time value was
adjusted to the beginning of all gene variants by adding 50 wait cycles to the beginning of
each gene. This increased ribosome wait time simulates the bottleneck at translation
initiation. After adding the wait time values to the beginning of each gene, the BNI of
variants 80 and 82 changed from 160 to 202 at the beginning of the gene. The BNI of other
variants were not affected. This is because the BNIs of other genes were already greater than
202 and were located at the middle or the end rather than the beginning of the gene. On the
other hand, variants 80 and 82 did not have bottlenecks in the middle or end of the gene. In
essence, the model over-predicted the protein yield for variants 80 and 82 because
bottlenecking at initiation was not taken into account. Adding ribosome wait time value at
the beginning of each gene takes into account bottlenecking at initiation and, thus, improves
protein yield prediction.

The ordering of normalized GST activity from lowest to highest corresponds better
with BNI and Total Wait Time than with CAI and CAI (HEG) indices. *gst* variant 90D and
variant 0D was designed to have the similar CAI (0.54 and 0.56 respectively) and CAI
(HEG) (0.31 and 0.33 respectively) but different ribosome displacements. Variant 90D was
designed to have higher ribosome displacement while variant 0D has lower displacement.
This resulted in a higher BNI and Total Wait Time for variant 90D when compared to 0D
(see Table 6.1). BNI and Total Wait Time predicted that variant 0D will have higher yield
compared to 90D. According to CAI and CAI (HEG), the two variants should have similar
yield; however, *gst* variant 0D showed 23% higher yield than variant 90D. This indicated that
BNI and Total Wait Time more accurately predict the activity levels of 90D and 0D
compared to CAI and CAI (HEG). Variant 23 showed a 17% increase in activity level compared to variant 90D. CAI and CAI (HEG) predicted that the yield of variant 23 is lower than that of variant 90D. Contrastingly, BNI and Total Wait Time predicted that variant 23 produces a higher yield than variant 90D. BNI and Total Wait Time predicted GST activity more accurately. According to CAI and CAI (HEG), variants 80 and 82 should produce a lot more yield compared to variant M and VM. However, measured activity levels showed that variants 80 and 82 produced similar levels to variants M and VM. BNI more accurately predicted the protein yield of variants 80 and 82 compared to variants M and VM given the inclusion of ribosome wait time value at initiation. The Total Wait Time index did not accurately predict the protein yield of variants 80 and 82 versus variants M and VM because it does not take bottlenecks into account.

![Figure 6.37. Correlation of normalized GST activity with BNI. This correlation scored an r of -0.96 and r-squared of 0.93.](image)
Figure 6.38. Correlation of normalized GST activity with Total Wait Time. This correlation scored an $r$ of -0.90 and $r^2$ of 0.80.

Figure 6.39. Correlation of normalized GST activity and CAI. This correlation scored an $r$ of 0.75 and $r^2$ of 0.57.
6.3.4 Sensitivity Analysis of Window Sizes

A sliding window size of 20 codons was used to calculate the BNI. This window size was chosen because it corresponds to the length of the ribosome footprint on the mRNA [3]; that is, the ribosome complex from the 30S head domain to the tail end of the 30S near the E-site covers a distance of approximately 20 codons on the mRNA. Sensitivity analysis was conducted to show that changes in window size made little difference in the prediction of protein yield. Window sizes used for the sensitivity analysis were 10, 15, 20, and 25 codons. The BNI was calculated using different window sizes for \( \text{gstr} \) wildtype as well as variants 16, 23, 80, 82, 90D, 0D, VM, and M (see Table 6.2).
Table 6.2. BNI for different window sizes and \textit{gst} variants. The bottleneck indices (BNIs) for variant windows sizes are tabulated for comparison.

<table>
<thead>
<tr>
<th>\textit{gst} Variants</th>
<th>BNI with Window Size 25</th>
<th>BNI with Window Size 20</th>
<th>BNI with Window Size 15</th>
<th>BNI with Window Size 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>349</td>
<td>296</td>
<td>246</td>
<td>205</td>
</tr>
<tr>
<td>90D</td>
<td>352</td>
<td>288</td>
<td>236</td>
<td>203</td>
</tr>
<tr>
<td>23</td>
<td>297</td>
<td>261</td>
<td>217</td>
<td>155</td>
</tr>
<tr>
<td>Wildtype</td>
<td>274</td>
<td>228</td>
<td>191</td>
<td>150</td>
</tr>
<tr>
<td>0D</td>
<td>251</td>
<td>205</td>
<td>172</td>
<td>128</td>
</tr>
<tr>
<td>80</td>
<td>229</td>
<td>202</td>
<td>172</td>
<td>128</td>
</tr>
<tr>
<td>VM</td>
<td>241</td>
<td>218</td>
<td>182</td>
<td>147</td>
</tr>
<tr>
<td>M</td>
<td>241</td>
<td>218</td>
<td>182</td>
<td>147</td>
</tr>
<tr>
<td>82</td>
<td>229</td>
<td>202</td>
<td>172</td>
<td>128</td>
</tr>
</tbody>
</table>

Because the BNI is used to predict relative protein yield, the change in ratio of the BNI for each \textit{gst} variants relative to the BNI for \textit{gst} wildtype for all window sizes were used to illustrate sensitivity. The BNI ratios for each window size were calculated as follows:

\[
\left( \frac{\text{BNI}_{\text{\textit{gst} variants}}}{\text{BNI}_{\text{\textit{gst} wildtype}}} \right)_{\text{Window Size } N}
\]  

(6.1)

where N is a window size of 10, 15, 20, or 25 codons. For example, the BNI ratio for \textit{gst} variant 16 for the window size of 25 is 349 (BNI of \textit{gst} variant 16) divided by 274 (BNI of \textit{gst} wildtype), which is 1.27. These BNI ratios are shown in Table 6.3 for different window sizes.
Table 6.3. BNI ratios for different window sizes and \textit{gst} variants. The ratios for BNI of \textit{gst} variants relative to BNI of \textit{gst} wildtype using different window sizes are tabulated.

<table>
<thead>
<tr>
<th>\textit{gst} Variants</th>
<th>BNI Ratio Window Size 25</th>
<th>BNI Ratio Window Size 20</th>
<th>BNI Ratio Window Size 15</th>
<th>BNI Ratio Window Size 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>1.27</td>
<td>1.3</td>
<td>1.29</td>
<td>1.37</td>
</tr>
<tr>
<td>90D</td>
<td>1.28</td>
<td>1.26</td>
<td>1.24</td>
<td>1.35</td>
</tr>
<tr>
<td>23</td>
<td>1.08</td>
<td>1.14</td>
<td>1.14</td>
<td>1.03</td>
</tr>
<tr>
<td>Wildtype</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0D</td>
<td>0.92</td>
<td>0.90</td>
<td>0.90</td>
<td>0.85</td>
</tr>
<tr>
<td>80</td>
<td>0.84</td>
<td>0.89</td>
<td>0.90</td>
<td>0.85</td>
</tr>
<tr>
<td>VM</td>
<td>0.88</td>
<td>0.96</td>
<td>0.95</td>
<td>0.98</td>
</tr>
<tr>
<td>M</td>
<td>0.88</td>
<td>0.96</td>
<td>0.95</td>
<td>0.98</td>
</tr>
<tr>
<td>82</td>
<td>0.84</td>
<td>0.89</td>
<td>0.90</td>
<td>0.85</td>
</tr>
</tbody>
</table>

To illustrate the change in BNI ratios for different window sizes, the percent error was used; the percent errors of the BNI ratios were measured relative to the BNI ratios for the window size of 20. The percent error of BNI ratios for each \textit{gst} variant was calculated as follows:

\[
\text{Percent Error} = \frac{|(\text{BNI ratio})_{\text{window size } N} - (\text{BNI ratio})_{\text{window size } 20}|}{(\text{BNI ratio})_{\text{window size } 20}} \times 100\% \quad (6.2)
\]

where \(N\) is the window size of 10, 15, 20, or 25 codons. For example, the percent error for the BNI ratio of \textit{gst} variant 16 is the absolute value of 1.27 subtracted by 1.3 multiplied by 100; this results in a 3\% error. The percent errors are shown in Table 6.4.
Table 6.4. Percent errors for BNI ratios using different window sizes. The percent errors for BNI ratios under different window size are tabulated.

<table>
<thead>
<tr>
<th>gst Variants</th>
<th>Percent Error Window Size 25</th>
<th>Percent Error Window Size 20</th>
<th>Percent Error Window Size 15</th>
<th>Percent Error Window Size 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>90D</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>23</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>Wildtype</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0D</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>80</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>VM</td>
<td>8</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>M</td>
<td>8</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>82</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Average</td>
<td>4.33 ± 2.78</td>
<td>0</td>
<td>0.78 ± 0.67</td>
<td>4.89 ± 3.55</td>
</tr>
</tbody>
</table>

± indicates one standard deviation.

Percent error calculations showed that for a change in window size of 10 from a window size of 20, there is on average only a 4.89% change in predicting protein (see Table 6.4). The percent error ranges from a 2-11% change. For a change in window size of 25, there is on average only a 4.33% change in predicted protein with the percent error ranging from 2-8% change. The percent error calculation indicated that predicting protein yield using BNI was not extremely sensitive to changes in window size. When correlating to GST activity, the BNIs using a window size of 25 had an r of -0.99 and r-squared of 0.98. BNIs using a window size of 20 correlated with an r of -0.96 and r-squared of 0.92. When the window size of 15 was used, the BNIs correlated with an r of -0.96 and r-squared of 0.92. For a window size of 10, the BNIs correlated with an r of -0.92 and r-squared of 0.84.

6.4 Comparison of Gene Optimizations

Five protein candidates were chosen and expressed in *E. coli* to assess the model’s optimization capabilities and were compared to codon bias optimization. The five candidates were GST, a glutathione S-transferase; PF0132, an alpha-glucosidase; CLJU_c11880, an
alcohol dehydrogenase; BIF1, a beta-galactosidase; and Rt8_0542, an endoglucanase and exoglucanase. Model optimizations of \textit{gst} and \textit{bif1} were performed as described in \textbf{Chapters 4.4 and 5.6}. Model optimization of \textit{gst} were compared to that of two codon bias optimization algorithms: 1) the “one amino acid, one codon” method, and 2) the “codon table matching, random guided” method as described in \textbf{Chapter 3.3.1}. Optimization of \textit{clju_c11880} and \textit{pf0132} was performed by minimizing displacement only; this ensures the codon bias remains approximately the same in both wildtype and model-optimized variants. The codon adaptation index (CAI) [1] was used as a measurement of codon bias. The front ends of all variants were unchanged to standardize for deviation in protein yield caused by initiation. The first 51 bases of \textit{gst} and \textit{bif1} were optimized using Salis’s RBS calculator [5]. Changing the front end of a gene using Salis’s RBS calculator [5] optimizes translation initiation such that there should not be a bottleneck at initiation. The resulting optimized front end generated by Salis’s RBS calculator was used in all \textit{gst} and \textit{bif1} variants; whereas, the first 90 bases of optimized \textit{clju_c11880} and \textit{pf0132} used that of their respective wildtypes.

\section*{6.4.1 GST, Glutathione-S-transferase}

\subsection*{6.4.1.1 Gene Optimization and Graphical Analysis}

\textit{gst} was optimized by choosing the codons that code for the most abundant tRNA while minimizing force and displacement (see \textbf{Chapters 4.4 and 5.6}). This kept the average phase angle of the gene close to the species angle of -23 degrees. \textit{gst} was also optimized using two codon bias optimization algorithms. “Codon bias-optimized 1” was optimized using the “one amino acid, one codon method” by Optimizer [6]. “Codon bias-optimized 2” was optimized using the “guided random, codon table matching” method by IDTDNA, Inc.’s
CodonOpt software. The codon bias optimization algorithms used highly expressed genes as the reference genes. The first 51 bases of $gst$ were optimized using Salis’s RBS calculator [5]. Optimizing the front end using Salis’s RBS calculator [5] optimizes translation initiation so that there should not be a bottleneck at initiation. The resulting optimized front end generated by Salis’s RBS calculator was used in all $gst$ variants.

Polar plot analysis showed that the average phase angle of “codon bias-optimized 2” $gst$ was shifted to -25 degrees from the wildtype phase angle of -61 degrees (see Figure 6.41). The average phase angle of “codon bias-optimized 1” $gst$ was shifted to -36 degrees from the wildtype phase angle of -61 degrees. The model-optimized variant was optimized close to the species angle, so the phase angle stayed close to -21 degrees.

**Figure 6.41. Polar plot of wildtype and optimized $gst$ variants.** Wildtype (purple) average phase angle is -61 degrees, model-optimized (red) average phase angle is -21 degrees, “codon bias-optimized 1” (green) angle average phase is -36 degrees, and “codon bias-optimized 2” (blue) average phase angle is -25 degrees.
For wildtype \textit{gst}, the ribosome was displaced from codon 50 to the stop codon, as shown in Figure 6.42. The ribosome displacement plot showed that both codon bias-optimized and model-optimized variants reduce displacement (see Figure 6.42). The model-optimized variant reduced displacement the most.

[Image of Figure 6.42: Displacement plot of wildtype and optimized \textit{gst} variants.]

Displacement plots for wildtype (purple), model-optimized (red), “codon bias-optimized 1” (green), “codon bias-optimized 2” (blue) are shown. All optimized variants minimized displacement. Model-optimized variants minimized displacement most.

Wildtype \textit{gst} showed clustering of high ribosome wait time at around codon 65 and codon 200 (see Figure 6.43) and bottlenecks at around codon 65 and codon 198 (see Figure 6.44). Ribosome wait time plots showed that both codon bias and model-optimized variants lowered wait times throughout the coding region (see Figures 6.45, 6.47, 6.49) as compared to wildtype (Figure 6.43). Translation bottleneck plots showed that all optimization methods lowered bottlenecks throughout the coding region (see Figures 6.46, 6.48, 6.50) as compared to wildtype (Figure 6.44). Normalized GST activity and protein yield prediction indices are shown in Table 6.5.
**Figure 6.43. Ribosome wait time plot of *gst* wildtype.** Total Wait Time is 1696 cycles. There are clustering of high ribosome wait times at around codon 65 and codon 200.

**Figure 6.44. Translation bottleneck plot of *gst* wildtype.** The BNI for wildtype *gst* is 233 at codon 198. Bottlenecks are identified at around codon 65 and codon 198.
Figure 6.45. Ribosome wait time plot of \textit{gst} model-optimized. Total Wait Time is 1019 cycles. Wait times were reduced throughout the codon region after optimization.

Figure 6.46. Translation bottleneck plot of \textit{gst} model-optimized. BNI is 123 at codon 138. Bottlenecks were reduced throughout the coding region after optimization.
Figure 6.47. Ribosome wait time plot of \( \text{gst} \) “codon bias-optimized 1”. Total Wait Time is 1103 cycles. Wait times were reduced throughout the codon region after optimization.

Figure 6.48. Translation bottleneck plot of \( \text{gst} \) “codon bias-optimized 1”. BNI is 139 at codon 138. Bottlenecks were reduced throughout the coding region after optimization.
Figure 6.49. Ribosome wait time plot of *gst* “codon bias-optimized 2”. Total Wait Time is 1527 cycles. Wait times were reduced throughout the codon region after optimization.

Figure 6.50. Translation bottleneck plot of *gst* “codon bias-optimized 2”. BNI is 161 at codon 187. Bottlenecks were reduced throughout the coding region after optimization.
6.4.1.2 Experimental Procedures and Results

Wildtype and optimized $gst$ variants were synthesized and cloned into pBAD/Myc-His inducible plasmid available from Thermo Fisher Scientific, Inc. $gst$ genes were synthesized and cloned by Genewiz, Inc. or GeneScript, Inc. (see Appendix E.1). The plasmids containing the gene insert of interest were transformed into chemically competent TOP10 $E. coli$ cells (available from Thermo Fisher Scientific, Inc.) and expressed at 0.02% w/v (0.2 mg/ml) arabinose for 3 hours (see Appendix E.2 and E.3). The $P_{bad}$ inducible promoter [11] was used to normalize for transcript levels. One milliliter of induced cells was harvested, pelleted, and lysed with BugBuster and Lysonase (both available from EMD Millipore, Inc.). GST activity was quantified using GST assay kit (E.C. 2.5.1.18) available from Sigma Aldrich, Inc. (see Appendix E.4). Because of increased GST activity, cell lysates were diluted twofold before assaying. One unit of GST is defined as the amount of GST enzyme producing 1 umol of GS-DNB conjugate per minute. GST specific activity was normalized to total protein using Pierce™ BCA Protein Assay Kit available from Thermo Fisher Scientific, Inc. Total protein normalization was performed to standardize for deviations in the number of cells collected. Units for normalized GST activity levels were defined as units of GST per $A_{562nm}$ of total protein where one $A_{562nm}$ equals 18 mg of total protein. Calculations for units of normalized GST activity can be found in Appendix E.4. Three independent inductions were conducted to test for replicability. From each induction, three samples were collected for a total of nine samples assayed. Detailed experimental procedures can be found in Appendices E.1 through E.4.
Normalized GST activity and protein yield prediction indices for each $gst$ variant can be found in Table 6.5. Model-optimized variants showed an increase of 53% over wildtype, “codon bias-optimized 1” showed a 13% increase over wildtype, and “codon bias-optimized 2” showed a 34% increase over wildtype. All optimizations decreased Total Wait Time and BNI and changed total GC content closer to 50%. CAI and CAI (HEG) indices under-predicted the protein yield of the model-optimized $gst$ variant and over-predicted the protein yield of “codon bias-optimized 1” variant. BNI and Total Wait Time over-predicted the protein yield of “codon bias-optimized 1” variant.

<table>
<thead>
<tr>
<th>$gst$ Variants</th>
<th>Normalized GST Activity$^1$</th>
<th>CAI</th>
<th>CAI (HEG)</th>
<th>Total Wait Time (Cycles)</th>
<th>BNI</th>
<th>GC%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildtype</td>
<td>0.607 ± 0.041</td>
<td>0.60</td>
<td>0.37</td>
<td>1696</td>
<td>233</td>
<td>39</td>
</tr>
<tr>
<td>Model-optimized</td>
<td>0.927 ± 0.035</td>
<td>0.76</td>
<td>0.68</td>
<td>1019</td>
<td>123</td>
<td>45</td>
</tr>
<tr>
<td>Codon bias-optimized 1</td>
<td>0.684 ± 0.050</td>
<td>0.84</td>
<td>0.97</td>
<td>1103</td>
<td>139</td>
<td>51</td>
</tr>
<tr>
<td>Codon bias-optimized 2</td>
<td>0.814 ± 0.024</td>
<td>0.83</td>
<td>0.80</td>
<td>1171</td>
<td>161</td>
<td>49</td>
</tr>
</tbody>
</table>

$^1$ indicates one standard deviation. Normalized GST activity is measured as units of GST per $A_{562nm}$ of total protein where one $A_{562nm}$ equals 18 mg of total protein; one unit of GST is defined as the amount of GST enzyme producing 1 umol of GS-DNB conjugate per minute.

6.4.2 CLJU_c11880, Alcohol Dehydrogenase

6.4.2.1 Gene Optimization and Graphical Analysis

$clju_c11880$ is an alcohol dehydrogenase (ADH) and was optimized by minimizing displacement only (see Figure 6.52). This was done to ensure that codon bias remained similar between the two variants to better compare the effects of ribosome displacement.
alone. Minimizing displacement also shifts the average phase angle of the gene (-60 degrees) closer to the species angle of -25 degrees (see Figure 6.51). The bottleneck index (BNI) is lowered from 615 at codon 317 to 322 at codon 93, as shown in Figure 6.54 and Figure 6.56. Wildtype clju_c11880 showed very high wait times from codon 300 to codon 350 (see Figure 6.53). Total Wait Time is also lowered from 3905 to 3047 cycles after optimization (see Figure 6.53, Figure 6.55, and Table 6.6).

Figure 6.51. Polar plot of wildtype and optimized clju_c11880. Wildtype (red) average phase is -60 degrees, and model-optimized (blue) average phase angle is -25 degrees.

Figure 6.52. Displacement plot of wildtype and optimized clju_c11880. The displacement plots of wildtype (red) and model-optimized (blue) clju_c11880 are produced, showing that model-optimized clju_c11880 minimizes displacement effectively.
Figure 6.53. Ribosome wait time plot of wildtype clju_c11880. The Total Wait Time is 3905 cycles for the wildtype.

Figure 6.54. Translation bottleneck plot of wildtype clju_c11880. The BNI for wildtype clju_c11880 is 628 at codon 317.
Figure 6.55. Ribosome wait time plot of model-optimized \textit{clju\_c11880}. The Total Wait Time is 3047 cycles for the optimized variant.

Figure 6.56. Translation bottleneck plot model-optimized \textit{clju\_c11880}. The BNI for optimized \textit{clju\_c11880} is 322 at codon 93.
6.4.2.1 Experimental Procedures and Results

Both wildtype and model-optimized *cjlu_c11880* genes were synthesized and cloned into pBAD/Myc-His vectors by Genewiz Inc. (see Appendix E.1). The plasmids containing the gene insert of interest were transformed into chemically competent TOP10 *E. coli* cells (available from Thermo Fisher Scientific, Inc.) and induced at 0.2% w/v (2 mg/ml) arabinose for 4 hours (see Appendices E.2 and E.3). The P<sub>bad</sub> inducible promoter [11] was used to normalize for transcript levels. Two milliliters of induced cells were harvested, pelleted, and lysed with BugBuster and Lysonase (both available from EMD Millipore, Inc.). Alcohol dehydrogenase (ADH, E.C. 1.1.1.1) enzymatic assay (protocol available from Sigma Aldrich, Inc.) and SDS-PAGE were conducted on the cell lysates to quantify CLJU_C11880 protein yield (see Appendix E.6). One unit of ADH is defined as the amount of ADH enzyme producing 1 umol of NADH per minute. ADH specific activity was normalized to total protein using the Pierce™ BCA protein assay kit available from Thermo Fisher Scientific, Inc. Total protein normalization was performed to standardize for deviations in the number of cells collected. Units for normalized ADH activity were defined as units of ADH per A<sub>562nm</sub> of total protein where one A<sub>562nm</sub> equals 18 mg of total protein. Calculations for units of normalized ADH activity can be found in Appendix E.6. Two independent inductions were conducted to test for replicability. From each induction two samples were collected for a total of four samples assayed. Detailed protocols for these procedures are provided in Appendices E.1, E.2, E.3, and E.6.

Wildtype and optimized variants produced protein in both soluble (active) form and insoluble (inactive) inclusion bodies. Both fractions of proteins were quantified to compare
production between the wildtype and optimized \textit{cjluc11880}. The enzymatic assay results are shown in Table 6.6. A 44% increase in the activity of the optimized variant was measured.

<table>
<thead>
<tr>
<th>\textit{cjluc11880} Variants</th>
<th>Normalized ADH Activity$^1$</th>
<th>CAI</th>
<th>CAI (HEG)</th>
<th>Total Wait Time (Cycles)</th>
<th>BNI</th>
<th>GC%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildtype</td>
<td>0.025 ± 0.004</td>
<td>0.53</td>
<td>0.25</td>
<td>3905</td>
<td>615</td>
<td>30</td>
</tr>
<tr>
<td>Model-optimized</td>
<td>0.036 ± 0.006</td>
<td>0.57</td>
<td>0.29</td>
<td>3047</td>
<td>322</td>
<td>40</td>
</tr>
</tbody>
</table>

$^1$Normalized ADH activity is measured as units of ADH per A$_{622nm}$ of total protein where one A$_{622nm}$ equals 18 mg of total protein; one unit of ADH is defined as the amount of ADH enzyme producing 1 umol of NADH per minute.

CLJU_c11880 was also produced in insoluble inclusion body form. Inclusion bodies were purified and quantified using SDS-PAGE as described in Appendix E.6. A negative variant, a transformed plasmid which does not contain the \textit{cjluc11880} gene insert, was also expressed along with \textit{cjluc11880} variants and visualized on an SDS-PAGE gel. A gel comparing the production of CLJU_c11880 across variants is provided in Figure 6.57. Samples shown were from the whole cell lysate, soluble fraction of active enzyme, and insoluble fraction of inclusion bodies. From this comparison, there was a marked decrease in the production of inclusion bodies in the optimized \textit{cjluc11880} variant as indicated by black arrows. The lack of inclusion body production in the negative variant (the no-gene insert plasmid) and the estimated size of ~40 kDa for the protein bands (marked by red and black arrows) indicated that these bands are presumed to be CLJU_C11880, the protein of interest.
6.4.3 PF0132, Alpha-Glucosidase

6.4.3.1 Gene Optimization and Graphical Analysis

*pf0132* was also optimized based on minimizing displacement only; the displacements of both the wildtype and optimized variants are shown in Figure 6.59. This was done to verify whether minimizing displacement increases protein yield. Minimizing displacement changes the phase angle of the optimized variant to -44 degrees (see Figure 6.58). Optimizing *pf0132* also decreased Total Wait Time and lowered bottleneck effects (see Table 6.7 and Figures 6.60, 6.61, 6.62, and 6.63).
Figure 6.58. Polar plot of \textit{pf0132} variants. The wildtype (red) and model-optimized (blue) are both portrayed. Wildtype’s average phase is -85 degrees and that of the model-optimized variant is -44 degree.

Figure 6.59. Displacement plot of \textit{pf0132} variants. The wildtype variant’s displacement (green) and optimized variant’s displacement (blue) are both portrayed.
Figure 6.60. Ribosome wait time plot of wildtype pf0132. The wait time plot of the wildtype variant is shown where the Total Wait Time is 4044 cycles.

Figure 6.61. Translation bottleneck plot wildtype pf0132. The BNI for wildtype pf0132 is 282 at codon 25.
Figure 6.62. Ribosome wait time plot of model-optimized *pf0132*. The Total Wait Time for the optimized variant is 3159 cycles.

Figure 6.63. Translation bottleneck plot of model-optimized *pf0132*. The BNI for optimized *pf0132* is 222 at codon 25.
Table 6.7. \textit{pf0132} variants and their protein yield prediction indices. CAI, CAI (HEG), Total Wait Time, and BNI values for \textit{pf0132} variants are listed. CAI is the codon adaptation index using \textit{E. coli} genomic coding sequences [9] as reference genes. CAI (HEG) is the codon adaptation index using highly expressed genes from [6] as reference genes. Total Wait Time and bottleneck index (BNI) are protein yield prediction indices from the model. GC\% is percent of total GC content.

<table>
<thead>
<tr>
<th>\textit{pf0132} Variants</th>
<th>CAI</th>
<th>CAI (HEG)</th>
<th>Total Wait Time (Cycles)</th>
<th>BNI</th>
<th>GC%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildtype</td>
<td>0.48</td>
<td>0.25</td>
<td>4044</td>
<td>282</td>
<td>44</td>
</tr>
<tr>
<td>Model-optimized</td>
<td>0.634</td>
<td>0.3</td>
<td>3159</td>
<td>222</td>
<td>48</td>
</tr>
</tbody>
</table>

6.4.3.2 Experimental Procedures and Results

The wildtype \textit{pf0132} was cloned from the \textit{Pyrococcus furiosus} genome [2]. The optimized \textit{pf0132} gene was synthesized by Genscript, Inc. (see Appendix E.1). Both wildtype and optimized \textit{pf0132} were cloned into pBAD-Myc-His plasmids (available from Thermo Fisher Scientific, Inc.) for protein production in TOP10 \textit{E. coli} strain (available from Thermo Fisher Scientific, Inc.) with a final arabinose concentration of 0.2\% w/v (2 mg/ml) for 4 hours (see Appendix E.2 and E.3). The \textit{P}_{\text{bad}} inducible promoter [11] was used to normalize for transcript levels. One milliliter of induced cells was harvested, pelleted, and lysed with Bugbuster and Lysonase (both available from EMD Millipore, Inc.). The resulting lysate was used for assaying and SDS-PAGE (see Appendix E.5). Detailed protocols for these procedures are provided in Appendices E.1, E.2, E.3, and E.5.

\textit{pf0132} and its optimized variant produced insoluble inclusion bodies with no enzymatic activity. Therefore, the quantity of (inactive) protein produced was measured with SDS-PAGE gel, as described in Appendix E.5. Densitometer analysis of SDS-PAGE showed a 210\% increase in density of the bands (at 55 kD and 40 kD) in optimized PF0132 versus the bands (55 kD, 32 kD) in wildtype PF0132 (see Figure 6.64). This implies a
substantial improvement in the production of PF0132 protein compared to the wildtype, although the optimized gene was produced as inclusion body form. Protein bands were identified as PF0132 using MALDI-TOF/TOF and LC-ESI-MS/MS mass spectrometry.

Figure 6.64. SDS-PAGE of wildtype and model-optimized pf0132. Lane L) ladder, SW) water soluble fraction of wildtype, IW) inclusion body purification of wildtype, SO) water soluble fraction of model-optimized, IO) inclusion body purification of model-optimized. The red arrow points to wildtype inclusion body bands, and the black arrow points to bands of model-optimized inclusion bodies.
6.4.4 BIF1, Beta-Galactosidase

6.4.4.1 Gene Optimization and Graphical Analysis

*bif1* (GenBank: AJ272131) is a beta-galactosidase from *Bifidobacterium bifidum DSM20215* [3]. *bif1* was optimized using the model based on the concepts described in Chapter 4.4. Wildtype and model-optimized variants were also compared to codon bias-optimized variants. Codon bias optimization was performed by the “guided random, codon table matching” method (see Chapter 3.3.1) using IDTDNA, Inc.’s CodonOpt software. The codon bias optimization used highly express genes as the reference genes. The first 51 bases of *bif1* were optimized using Salis’s RBS calculator [5]. Optimizing the front end using Salis’s RBS calculator [5] optimizes translation initiation, so there should not be a bottleneck at initiation. The resulting optimized front end generated by Salis’s RBS calculator was used in all *bif1* variants.

The average phase angle of wildtype *bif1* was 5 degrees and neared the species angle of *E. coli* when optimized (see Figure 6.65). The codon bias-optimized variant has an average phase angle of -43 degrees, which is further away from the species angle than that of the model-optimized variant. The model-optimized variant has an average phase angle of -26 degrees, which is closer to the species angle.
Figure 6.65. Polar plot of wildtype and optimized \textit{bif1} variants. The polar plots of wildtype (red), model-optimized (blue), and codon bias-optimized (green) variants are shown. Wildtype average phase is 5 degrees, model-optimized average phase is -26 degrees, and codon bias-optimized average phase is -43 degrees.

The difference in phase angle that arises in ribosome displacement is much greater in longer genes, as shown in Figure 6.66. The model-optimized variant minimizes displacement more so than wildtype or codon-optimized variants.

Figure 6.66. Ribosome displacement plot of wildtype and optimized \textit{bif1} variants. The displacement plot of wildtype (red), model-optimized (blue), and codon bias-optimized (green) variants are depicted. Model-optimized displacement stays closer to zero than wildtype and codon bias-optimized variants.
The global GC content of the wildtype variant (see Table 6.8) is 63%, and local GC content hovers around 60% throughout the gene (see Figure 6.69). However, optimization using codon bias and the model changes both global GC content (see Table 6.8) and local GC content (Figures 6.72 and 6.75) to approximately 50%. Model optimization lowered both Total Wait Time and bottlenecks. Codon bias optimization did not substantially decrease Total Wait Time or bottlenecks. In fact, the bottleneck effect increased around codon 100 in the codon bias-optimized variant (see Figure 6.74) when compared to the wildtype (see Figure 6.68). Ribosome wait time plots are shown in Figures 6.67, 6.70, and 6.73. Translation bottleneck plots are shown in Figures 6.68, 6.71, and 6.74. Local GC content plots are shown in Figures 6.69, 6.72, and 6.75. Local GC content plots were computed using [14] with a window size of 50. A summary of all the protein yield prediction indices and beta-galactosidase activities for all three variants are tabulated in Table 6.8.
Figure 6.67. Ribosome wait time plot of wildtype bif1. The Total Wait Time is 6397 cycles.

Figure 6.68. Translation plot bottleneck plot of wildtype bif1. The BNI for wildtype bif1 is 207 at codon 843.

Figure 6.69. Local GC content along the coding region of wildtype bif1. GC content is centered at approximately 60%.
Figure 6.70. Ribosome wait time plot of model-optimized *bif1*. The Total Wait Time for model-optimized *bif1* is 4875 cycles.

Figure 6.71. Translation plot bottleneck plot of model-optimized *bif1*. The BNI for model-optimized *bif1* is 147 at codon 123.

Figure 6.72. Local GC content along the coding region of model-optimized *bif1*. The GC content is centered at approximately 50% throughout the gene.
Figure 6.73. Ribosome wait time plot of codon bias-optimized *bif1*. The total wait time for codon-bias optimized *bif1* is 6289 cycles.

Figure 6.74. Translation plot neck plot of codon bias-optimized *bif1*. The BNI for codon bias-optimized *bif1* is 197 at codon 123.

Figure 6.75. Local GC content along the coding region of codon bias-optimized *bif1*. The GC content is centered at approximately 50% throughout the gene.
6.4.4.2 Experimental Procedures and Results

*bif1* wildtype and optimized genes were synthesized by Genewiz, Inc. (see Appendix E.1) Both wildtype and optimized genes were cloned into pBAD-Myc-His plasmids (available from Thermo Fisher Scientific, Inc.) for protein production in the TOP10 *E. coli* strain (also available from Thermo Fisher Scientific, Inc.) using a final arabinose concentration of 0.2% w/v (2 mg/ml) for 4 hours (see Appendices E.2 and E.3). The P<sub>bad</sub> inducible promoter [11] was used to normalize transcript levels. Two milliliters of induced cells were harvested, pelleted, and lysed with BugBuster and Lysonase Bioprocessing Reagent (both available from EMD Millipore, Inc.). β-galactosidase (E.C. 3.2.1.23) enzymatic assay kit (available from Sigma Aldrich, Inc.) and SDS-PAGE were conducted on the cell lysates to quantify *bif1* protein yield (see Appendix E.7). One unit of β-galactosidase is defined as the amount of β-galactosidase enzyme producing 1 umol of o-nitrophenol (ONP) per minute. β-galactosidase specific activity was normalized to total protein yield using the Pierce™ BCA protein assay kit available from Thermo Fisher Scientific, Inc. Total protein normalization was performed to standardize for deviations in the number of cells collected. Units for normalized β-galactosidase activity were defined as units of β-galactosidase per A<sub>562nm</sub> of total protein, where one A<sub>562nm</sub> equals 18 mg of total protein. Calculations for units of normalized β-galactosidase activity can be found in Appendix E.7. One induction was conducted with two samples collected from that induction. One sample was used for assay, and the other sample was used for SDS-PAGE. Detailed protocols for these procedures are provided in Appendices E.1, E.2, E.3, and E.7.
Wildtype *bif1* and its optimized variants were mainly synthesized as inclusion body form; however, small quantities of active enzymes were measured using the β-galactosidase activity assay kit and SDS-PAGE. A negative variant (does not contain the *bif1* gene insert) was also expressed with *bif1* variants and visualized on an SDS-PAGE gel. Inclusion body fraction, water soluble fractions, and whole cell extracts were visualized on an SDS-PAGE gel to compare differences between the wildtype, optimized variants, and negative variant. Preliminary results showed a 6% increase in activity of the model-optimized versus the codon bias-optimized variants. Activity assay showed that both optimized variants expressed lower than the wildtype gene. However, these measurements recorded were at the lower end of the linear range for the assay.

<table>
<thead>
<tr>
<th><em>bif1</em> Variants</th>
<th>Normalized β-galactosidase Activity x 10^{-3}</th>
<th>CAI</th>
<th>CAI (HEG)</th>
<th>Total Wait Time (Cycles)</th>
<th>BNI</th>
<th>GC%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildtype</td>
<td>4.94 ± 0.15</td>
<td>0.73</td>
<td>0.57</td>
<td>6397</td>
<td>207</td>
<td>63%</td>
</tr>
<tr>
<td>Model-optimized</td>
<td>4.85 ± 0.22</td>
<td>0.68</td>
<td>0.56</td>
<td>4875</td>
<td>147</td>
<td>52%</td>
</tr>
<tr>
<td>Codon bias-optimized</td>
<td>4.57 ± 0.10</td>
<td>0.73</td>
<td>0.55</td>
<td>6289</td>
<td>197</td>
<td>53%</td>
</tr>
</tbody>
</table>

± indicates one standard deviation. Normalized β-galactosidase activity is measured as units of β-galactosidase per A_{562nm} of total protein where one A_{562nm} equals 18 mg of total protein; one unit of β-galactosidase is defined as the amount of β-galactosidase enzyme producing 1 umol of o-nitrophenol per minute.

Visualization of BIF1 on SDS-PAGE provided more conclusive measurements to compare gene variants. The SDS-PAGE results are presented in **Figure 6.76**. The inclusion body, water soluble, and whole cell extract fractions showed a band (indicated by red arrows between 130 kDa and 170 kDa) that is presumed to be BIF1; this band does not exist in the
negative inclusion body fraction. The band appears to be equally dense in wildtype, the codon bias-optimized, and model-optimized variants; however, it is less dense in water soluble fraction than in the inclusion body fraction. It is also interesting to note that this band is greater than the predicted 112 kDa molecular weight of BIF1 as published in [3]. By comparing samples from the water soluble fraction versus that of inclusion body fraction, this gel provides evidence for the production of water soluble BIF1 despite low activity on the results of the enzymatic assay. Another unknown band (indicated by black arrows around 70 kDa) was also produced in the water soluble fraction of bif1 variants but not in the negative variant.

Figure 6.76. SDS-PAGE of wildtype and optimized bif1, a β-galactosidase. This gel shows lanes for the inclusion body fractions, soluble fractions, and whole cell lysates. Inclusion body lanes: IW) wildtype, IC) codon bias-optimized, IO) model-optimized, and IN) negative. Soluble fraction lanes: SW) wildtype, SC) codon bias-optimized, SO) model-optimized, and SN) negative. Whole cell lysate lanes: WCW) wildtype, WCC) codon bias-optimized, WCO) model-optimized, and WCN) negative. L is PageRuler Prestain ladder. The negative lane contains the plasmid but no gene insert. Red arrows indicate bands corresponding to BIF1 produced as inclusion bodies. Black arrow indicates water soluble unknown band from expression of BIF1 variants. This unknown band does not appear in the negative.
6.4.5 Rt8_0542

6.4.5.1 Gene Optimization and Graphical Analysis

Wildtype and codon bias-optimized rt8_0542 was analyzed using the model to determine the dynamics of codon bias optimization when compared to the wildtype gene. Codon bias optimization was performed using the “guided random, codon table matching” method (Chapter 3.3.1) by IDTDNA, Inc.’s CodonOpt software. Wildtype rt8_0542 has not been successfully produced in E. coli while codon bias-optimized rt8_0542 was successfully produced as both water soluble protein and inclusion bodies. Model-optimized rt8_0542 is also presented as a reference for comparison between wildtype and codon bias optimization; however, in vivo expression of this variant has not been performed. Both global and local GC contents were computed for all variants. Local GC content plots were computed using [14] with a window size of 50. A summary of all protein yield prediction indices for the three variants are tabulated in Table 6.9.

The phase angle of wildtype is -90 degrees in the beginning of the gene and gradually shifts to -10 degrees in the latter half of the sequence around codon 600, as shown in Figure 6.77. A shift in local GC content of the wildtype gene from 40 to 50% GC content also occurs at a similar spot at nucleotide 2000 (Figure 6.81). Because the phase angle begins at the “wrong” angle (-90 degrees), the ribosome was displaced 1.5 displacement units downstream (corresponding to a misalignment of ¾ nucleotide towards the +1 reading frame) at around codon 400 (see Figure 6.78) when translating the wildtype gene. The ribosome wait time plot of the wildtype rt8_0542 showed clustering of high ribosome wait times (70 to 129 cycles) around codons 400 through 500 (see Figure 6.79). Normal ribosome wait times
for endogenous *E. coli* genes are around 5 to 20 cycles per codon. This leads to bottlenecks at around codon 400 through 500 (see Figure 6.80). The maximum ribosome wait time is 129 cycles at codon 476. Codon 476 codes for the isoleucine rare codon ATA. According to the model, the ribosomes are unlikely to translate past codon 476 due to high ribosome displacement and very high ribosome wait time. Wildtype variant showed a BNI of 362 and Total Wait Time of 15,053 cycles, which is higher than both model-optimized and codon bias-optimized variants (see Table 6.9).

![Image of polar plot](image)

**Figure 6.77. Polar plot of wildtype and optimized rt8.0542 variants.** Wildtype (red) average phase angle is -50 degrees, model-optimized (blue) average phase angle is -25 degrees, and codon bias-optimized (red) average phase is -24 degrees.

The codon bias-optimized variant showed an average phase angle of -24 degrees, which is very close to the species angle (see Figure 6.77). This codon bias optimization was fortuitous because the “guided random, codon table matching” optimization method results in random average phase angles which could be further away from the species angle. Because
the phase angle of the gene is near the species angle throughout the coding region, the local
GC content also remained around 50% (see Figure 6.87); whereas, the global GC content is
51%. The displacement plot of the codon bias-optimized variant (see Figure 6.78) showed
reduced displacement when compared to the wildtype variant. Ribosome wait times were
lowered to less than 20 cycles per codon, and bottlenecks were lowered throughout the entire
coding region compared to wildtype (see Figure 6.85 and 6.86). Codon bias-optimized
variant showed a BNI of 218 and Total Wait Time of 11,511 cycles, which is substantially
lower than the wildtype variant (see Table 6.9).

![Figure 6.78. Ribosome displacement plot of wildtype and optimized rt8_0542 variants.](image)
The codon bias-optimized variant (red) stays closer to zero displacement than wildtype
(blue). The model-optimized variant (blue) stays the closest to zero displacement.

The model-optimized variant showed an average phase angle of -25 degrees, which is
very close to the species angle (see Figure 6.77). Both global GC content (see Table 6.9)
and local GC content (see Figure 6.84) remained near 50%. The displacement plot of the
model-optimized variant showed minimal displacement that is lower than both wildtype and
codon bias-optimized variants (see Figure 6.78). Ribosome wait times were lowered to less
than 20 cycles per codon, and bottlenecks were lowered throughout the entire coding region compared to the wildtype (see Figure 6.82 and 6.83). The model-optimized variant showed a BNI of 152 and Total Wait Time of 9161 cycles, which is lower than both codon bias-optimized and wildtype variants (see Table 6.9).
Figure 6.79. Ribosome wait time plot of wildtype *rt8_0542*. The Total Wait Time is 15,053 cycles.

Figure 6.80. Translation plot neck plot of wildtype *rt8_0542*. The BNI of wildtype *rt8_0542* is 362 at codon 375.

Figure 6.81. GC content distribution along the gene for wildtype *rt8_0542*. GC content is centered at approximately 40% until the 2000\textsuperscript{th} nucleotide, after which the GC converges around 50%. This “shift” in GC content is also captured on the phase plot and displacement plot.
Figure 6.82. Ribosome wait time plot of model-optimized *rt8_0542*. The Total Wait Time for model-optimized *rt8_0542* is 9161 cycles.

Figure 6.83. Translation bottleneck plot of model-optimized *rt8_0542*. The BNI for model-optimized *rt8_0542* is 152 at codon 373.

Figure 6.84. GC content distribution along the gene for model-optimized *rt8_0542*. The GC content is centered on 50% throughout the entire gene.
Figure 6.85. Ribosome wait time plot of codon bias-optimized *rt8_0542*. The Total Wait Time for codon-bias optimized *rt8_0542* is 11,511 cycles.

Figure 6.86. Translation plot neck plot of codon bias-optimized *rt8_0542*. The BNI for codon bias-optimized *rt8_0542* is 218 at codon 846.

Figure 6.87. GC content along the gene using for codon bias-optimized *rt8_0542*. The GC content is centered on 50% throughout the entire gene.
Table 6.9. *rt8_0542* variants and their protein yield prediction indices. CAI, CAI (HEG), Total Wait Time, and BNI values for *rt8_0542* variants are listed. CAI is the codon adaptation index using *E. coli* genomic coding sequences [9] as reference genes. CAI (HEG) is the codon adaptation index using highly expressed genes from [6] as reference genes. Total Wait Time and bottleneck index (BNI) are protein yield prediction indices from the model. GC% is percent of total GC content.

<table>
<thead>
<tr>
<th><em>rt8_0542</em> Variants</th>
<th>CAI</th>
<th>CAI (HEG)</th>
<th>Total Wait Time (Cycles)</th>
<th>BNI</th>
<th>GC%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildtype</td>
<td>0.54</td>
<td>0.29</td>
<td>15053</td>
<td>362</td>
<td>47%</td>
</tr>
<tr>
<td>Model-optimized</td>
<td>0.69</td>
<td>0.58</td>
<td>9161</td>
<td>152</td>
<td>50%</td>
</tr>
<tr>
<td>Codon bias-optimized</td>
<td>0.71</td>
<td>0.51</td>
<td>11511</td>
<td>218</td>
<td>51%</td>
</tr>
</tbody>
</table>

6.4.5.2 Experimental Procedures and Results

Laboratory experiments were conducted by Jonathan Conway in Dr. Robert Kelly’s lab and are included in this dissertation with permission. Wildtype *rt8_0542* has not been successfully expressed in *E. coli*; however, the codon bias-optimized variant was successfully expressed as soluble and insoluble fractions. Codon bias-optimized *rt8_0542* was overexpressed using the T7 promoter (pET45 plasmid) in Arctic Express DE3 RIL *E. coli* strain at 13°C. Arctic Express DE3 RIL cells contain chaperones to assist in protein folding and to prevent or reduce inclusion body formation. The soluble portion of codon bias-optimized *rt8_0542* was purified using his-tagged purification (Ni-NTA column) and visualized on an SDS-PAGE gel, as shown in Figure 6.88.
Figure 6.88. SDS-PAGE gel of codon bias-optimized \( \text{rt8}_0542 \). Lane 1) whole cell extract, lane 2) Ni-NTA column purified \( \text{rt8}_0542 \), lane 3) cell extract from different construct (negative), and lane 4) protein ladder. Red arrows point to bands that correspond to \( \text{rt8}_0542 \). SDS-PAGE was conducted by Jonathan Conway from Dr. Robert Kelly’s lab. Image included in dissertation with permission.
6.5 Discussions

6.5.1 Comparison of Protein Yield Prediction Indices - BNI versus CAI

In order to compare the effectiveness of the standard codon adaptation index (CAI) versus the new bottleneck index (BNI) created with the model, the correlation between the indices and normalized GST activities were compared. BNI scored the best correlation with normalized GST activity levels with an r of -0.96 and r-squared of 0.92. Total Wait Time scored the second best correlation with an r of -0.89 and r-squared of 0.80. BNI and Total Wait Time correlated better with normalized GST activity levels than codon bias indices CAI and CAI (HEG). CAI correlated with an r of 0.75 and r-squared of 0.57 while CAI (HEG) scored an r of 0.68 and r-squared of 0.47. For reference, an r above 0.5 indicates that there is a correlation, and an r above 0.8 indicates a very high correlation.

Correlation between BNI and CAI yielded an r of 0.59. This indicates that BNI correlates with CAI because both intrinsically measure tRNA abundance usage. BNI uses actual tRNA abundance measurements from Dong et al. [4] while CAI uses genomic codon frequency as references. The two indices use two different reference sources to predict protein yield, yet both showed correlation with each other and with protein yield. This indicates that genomic codon frequency intrinsically measures tRNA abundance usage. However, unlike CAI and CAI (HEG), which only measure tRNA abundance usage, BNI also accounts for ribosome displacement, force, and clustering of high ribosome wait times (bottlenecks). BNI does not take bottlenecks at initiation or termination into account; however, ribosome wait time values can be added to increase bottlenecks at either location.
Total Wait Time did not score nearly as well as BNI because two genes can have the same Total Wait Time but different “slow” translation clustering sites which result in bottlenecks. The mRNA is concurrently translated into protein by multiple ribosomes; therefore, multiple “slow” translation regions clustered in local proximity are more detrimental to yield than impeding regions distributed across the gene. This was explained in Chapter 4.3. Total Wait Time does not consider bottlenecks within the codon region, initiation, or termination.

Both CAI and CAI (HEG) correlated similarly according to r and r-square values. However, the correlation was less than BNI and Total Wait Time. This is because CAI and CAI (HEG) only account for codon bias and do not consider the “ordering” of those codons within the gene. Two genes can have similar codon bias but different ribosome displacement and translation bottlenecks due to the ordering of codons. Unlike CAI and CAI (HEG), BNI and Total Wait Time consider the actual sequence of codons as it is translated in vivo to consider how that order of codons affects ribosome displacement, force, translation bottlenecks (only for BNI, not Total Wait Time), as well as aminoacyl-tRNA usage.

6.5.2 gst Optimization

All optimized variants shift the phase angle from -61 degrees (that of the wildtype gene) closer to E. coli’s species angle of -25 degrees (Figure 6.41), which minimizes displacement (Figure 6.42). All optimized variants shifted the GC content of the genes closer to 50%, which matches the GC content of the E. coli genome. Both codon bias-optimized and model-optimized variants minimize ribosome wait and bottlenecking effects; however, the
model-optimized variant minimizes bottlenecks better than codon bias-optimized variants (see Table 6.5).

All three optimized variants produced an increase in protein yield over the wildtype genes. Model-optimized variants produced the highest increase from the wildtype (53% improvement). The “codon bias-optimized 2” variant using the “guided random, codon table matching” method produced the next highest increase in yield (34% improvement). “Codon bias-optimized 1” variant using the “one amino acid, one codon” method produced the smallest increase in yield (13% improvement). Both codon bias optimization and model optimization did not show “substantial” gain (as observed in [20,22,23]) in protein yield compared to wildtype gst because wildtype gst is already highly expressed in E. coli. Optimization of gst for translation elongation has probably hit the upper limit for increasing protein yield.

A probable reason why the “one amino acid, one codon” method did not show a larger increase in yield is because of aminoacyl-tRNA depletion; this was explained in Chapter 3.3.1 and Chapter 3.3.2. The “one amino acid, one codon” method uses only one tRNA species per amino acid rather than spreading the tRNA usage as implemented in the “guided random, codon table matching” method. Optimization using the “one amino acid, one codon” method leads to competition between heterologous protein production and endogenous protein production for usage of available aminoacyl-tRNA. The competition depletes available aminoacyl-tRNA; therefore, protein yield suffers. These observations were explained in Welch et al. [17], Dittmar et al. [18], Elf et al. [19], and Menzella et al. [24].
bias-optimized 1” variant which used the “one amino acid, one codon” optimization method. This is because CAI, CAI (HEG), BNI, and Total Wait Time assume unlimited availability of aminoacyl-tRNA; however, availability of aminoacyl-tRNA is limited and can become depleted if both endogenous protein production and heterologous protein production compete for the same tRNA species. In addition, CAI and CAI (HEG) under-predicted the protein yield of model-optimized variant because CAI and CAI (HEG) do not take translation bottlenecks into account as explained in Chapter 6.5.1.

6.5.3 *clju_c11880* Optimization

*clju_c11880*, an alcohol dehydrogenase (ADH), was optimized by minimizing displacement only. This was done intentionally to assess whether minimizing displacement alone will increase protein yield. Optimizing by minimizing displacement also ensured that the codon bias was similar in both variant. Optimized *clju_c11880* showed increased enzymatic activity by 44% and decreased inclusion body production when compared to the wildtype.

The decrease in inclusion body production could be caused by the ribosome choosing the correct aminoacyl-tRNA due to reduction of displacement from codon 270 to 380. Picking the wrong aminoacyl-tRNA and incorporating the wrong amino acid can cause misfolding and production of inclusion bodies [25,26]. Wildtype *clju_c11880* also showed very high ribosome wait times from codon 300 to 350 (see Figure 6.53) leading to bottlenecks around the same position (see Figure 6.54). Optimizing *clju_c11880* reduced ribosome wait times and bottlenecks at the same positions (see Figures 6.55 and 6.56). The
ribosome may have stalled due to high ribosome wait times around codon 300 to 350. A stalled ribosome will dissociate and produce partially-completed CLJU_C11880 which may not fold correctly to form active enzymes. These partially produced polypeptides can form inclusion bodies [26] that are probably the source of the two smaller bands around 35 kDA in the inclusion body fractions on the SDS-PAGE (see Figure 6.57). The intensity of those two smaller bands is greatly reduced in the optimized variant, suggesting a reduction in inclusion body production. Reduction of ribosome wait times in the optimized variant may have reduced ribosome stalling and dissociations. Reduction in ribosome stalling and disassociation could have caused the reduction of inclusion body production as observed in the two smaller bands of the optimized variant. These aforementioned postulates and observations suggest that optimization of genes using the model’s wait time parameters may allow for improved protein folding and, therefore, reduce or eliminate inclusion body production; this postulate was also suggested by Pechmann et al. [12] and Li et al. [13] and was demonstrated by Siller et al. [11] in vivo by decreasing ribosome translation rates.

6.5.4 pf0132 Optimization

pf0132 was optimized by minimizing displacement only. This was done to see whether minimizing displacement would increase protein yield. pf0132 was only produced in inclusion body form. Optimized pf0132 showed increased inclusion body production when compared to the wildtype. pf0132 was cloned from an extremophile Pyrococcus furiosus that lives at 100°C. PF0132 was evolutionarily designed to fold at a much higher temperature than 37°C, the temperature at which E. coli expressed pf0132 in these experiments.
Thermophile proteins folded at mesophilic temperature tend to form inclusion bodies because of improper folding [27,28].

6.5.5 *bif1* Optimization

Wildtype, model-optimized, and codon bias-optimized variants of *bif1* were expressed in *E. coli*, and a comparison of their activity levels and inclusion body production was performed. All variants were primarily expressed as inclusion bodies. The presumed *bif1* band on the SDS-PAGE gel had a molecular weight between 130 kDa and 170 kDa (see Figure 6.76). This molecular weight is larger than the predicted molecular weight of 112 kDa as published in [3]. Water-soluble expression of BIF1 was detected through SDS-PAGE and β-galactosidase activity assays in low quantities. A water-soluble “unknown” band around 70 kDa was also observed through SDS-PAGE; this band was not observed in the negative-variant lane. Preliminary assay data suggested that the wildtype variant produced the most working enzyme followed by the model-optimized variant and codon bias-optimized variant. However, measurements recorded were at the lower end of the linear range for the assay. More experiments at different arabinose concentration are needed to be performed to quantify the precise activity level and inclusion body production of each variant and test for replicability. Mass-spectrometry should also be conducted on the BIF1 and the “unknown” protein bands to identify amino acid composition.

6.5.6 *rt8_0542* Optimization

Model analysis showed the potential cause of unsuccessful expression of wildtype *rt8_0542* in *E. coli*. An “incorrect” phase angle led to a displaced ribosome which increased
ribosome wait times and bottlenecks around codons 400 through 500. The maximum ribosome wait time is 129 cycles at codon 476. Codon 476 codes for the isoleucine rare codon ATA. According to the model, the ribosomes are unlikely to translate past codon 476 due to high ribosome displacement and very high ribosome wait time. This increase in ribosome wait time may be substantial enough (compared to endogenous genes and \textit{prfB}) for ribosomes to disassociate and stop translating. Alternatively, the ribosome could have potentially frameshifted due to high ribosome displacement of +1.5 units (misalignment of ¾ nucleotides towards the +1 reading frame) and encountered a stop codon in the new reading frame. For comparison purposes, endogenous genes showed ribosome wait times of 5-20 cycles while \textit{prfB} showed a ribosome wait time of 147 cycles at the frameshift site. Contrastingly, model analysis showed the underlying factors that lead to successful expression of codon bias-optimized variant in \textit{E. coli}. The phase angle of codon bias-optimized variant followed closely to the species angle throughout the coding region; this was “random luck” due to the fact that codon bias optimization (using the “guided random, codon table matching” method) is a stochastic process that does not always closely follow the species angle. Remaining close to the species angle will result in minimal compression or extension of the “spring”, which also minimizes displacement. Codon bias optimization reduced ribosome displacement, ribosome wait times, and bottlenecks. Model optimization of \textit{rt8 \_0542} was also included as a reference for \textit{in silico} comparison to codon bias optimization and wildtype variants. Optimizing \textit{rt8 \_0542} using the model (as described in \textbf{Chapter 4.4}) led to a reduction in ribosome displacement, ribosome wait time, and
translation bottlenecks. These reductions surpassed that of codon bias optimization as described in Chapter 6.4.5.

6.6 Conclusions

An index for determining yield was developed based on expression level of multiple mRNA variants of \textit{gst}. The index is named the bottleneck index (BNI), which measures the clustering of high ribosome wait times during the translation of genes. The development of this index was based on principle of “ribosome traffic jams” and the “ribosome footprint” as described in Chapter 4.3. Total ribosome wait time can also be used as measure of translation efficiency. In comparing these new metrics to measured protein yield, BNI correlated with an \textit{r}-square of 0.92 while Total Wait Time correlated with an \textit{r}-squared of 0.80. On the other hand, CAI and CAI (HEG) scored an \textit{r}-square of 0.57 and 0.47 respectively. This indicated that BNI and Total Wait Time may be a better index for predicting yield as explained in Chapter 6.3.3 and Chapter 6.5.1; however, BNI and Total Wait Time need to be further tested with a larger dataset to more extensively compare them to CAI’s prediction. BNI does not account for bottlenecks at initiation or termination, but values can manually be added at those locations to increase prediction accuracy. BNI may also over-predict protein yield, especially in the case of “one amino acid, one codon” codon bias optimization. This is because BNI assumes unlimited aminoacyl-tRNA availability (see Chapter 6.4.1).

Genes optimized using the model showed increased protein levels when compared to codon bias-optimized variants and wildtype variants. Model-optimized \textit{gst} showed a 53% increase over wildtype, “codon bias-optimized 2” showed a 34% increase of wildtype, and
“codon bias-optimized 1” showed a 13% increase over wildtype. Model-optimized $clju_c11880$ showed a 44% increase in activity and reduction in inclusion body formation compared to wildtype; this indicated that the model has the potential to reduce or eliminate inclusion bodies by changing ribosome wait time parameters as explained in Chapter 6.5.3. Model-optimized $pf0132$ showed an increase in protein production of 200% using densitometer analysis. Unfortunately, both wildtype and optimized variants were only produced as inclusion bodies. Wildtype, model-optimized, and codon bias-optimized variants of $bif1$ produced inclusion bodies and minor water soluble fractions. Differences in protein level between the variants were hard to discern from the preliminary data, and the results are in-conclusive. Further laboratory experiments are required. Analysis of wildtype $rt8_0542$ showed that the first 500 codons were detrimental to protein yield due to high ribosome wait times and bottlenecks. The model suggested that the ribosomes may have dissociated from the mRNA between codon 400 and 500; alternatively, the ribosomes could have potentially frameshifted due to high ribosome displacement of +1.5 units (displacement of $\frac{3}{4}$ nucleotides towards the +1 reading frame) and encountered a stop codon in the new reading frame. Conversely, codon bias optimization lowered ribosome wait times and bottlenecks which allowed the ribosomes to translate the gene more efficiently. The model-optimized variant showed greater reductions in ribosome displacement, Total Wait Time, and translation bottlenecks compared to codon bias-optimized variant.

As shown through polar plots, codon bias optimization changed the average phase angle of the wildtype genes into the “working” range for $E. coli$’s phase angle (see Appendix C). The “one amino acid, one codon” codon bias optimization method (using highly
expressed genes as reference) always changed the average phase angle of a gene near -45 degrees. This is because the average phase angle of highly expressed genes is near -45 degrees (see Appendix C). The “guided random, codon table matching” codon bias optimization method “randomly” changes the average phase angle of the gene with every optimization because it is a stochastic algorithm (see the phase angle of codon bias-optimized rt8_0542, bif1, and gst variants); however, the average phase angle still remains within the “working” range of E. coli’s phase angle. Both methods of codon bias optimization miss the species angle (although “codon table matching, guided random” method can by chance “hit” the species angle) resulting in slight displacement of the ribosome. This slight displacement accumulates for very long genes and increases ribosome wait time and bottlenecks, as shown in codon bias optimization of bif1. The model optimizes a gene’s phase angle using the species angle (see Chapter 5.6) which results in an average phase angle that is close to the species angle. Therefore, ribosome displacement is minimized more so than with codon bias optimization throughout the coding region. It should be noted that changing the phase angle of the gene (phase angle is related to the binding patterns of the 16S rRNA exposed tail and mRNA) also changes the global and local GC content of the gene. This observation is shown in the optimization examples of bif1 and rt8_0542. Therefore, the author contends that GC content is related to the nucleotide composition of the 16S rRNA 3’ exposed tail, as explained in Chapter 3.3.4.

The experiment results showed that reducing ribosome displacement, ribosome wait times, and translation bottlenecks usually increased total protein yield. In the case of the CLJU_c11880 optimization, lowering displacement, ribosome wait times, and bottlenecks
also reduced inclusion body production. The experimental results also showed that optimization using the model’s translation bottleneck principle showed greater improvements in protein yield than codon bias optimization. This is because codon bias optimization does not consider the “ordering” of codons which affects the location of 16S rRNA tail:mRNA bindings relative to the P-site. The distance of the 16S rRNA tail:mRNA bindings relative to the P-site is calculated using the “phase angle” as described in Chapter 5.4.2. An unfavorable binding (a phase angle that differs from the species angle) compresses or extends the spring which exerts a force on the ribosome. Therefore, this extra force could lead to a displaced ribosome. A displaced ribosome increases the ribosome wait times which can result in translation bottlenecks and, therefore, decrease protein yield.

Experimental results also showed that BNI was a better predictor of protein yield than CAI and CAI (HEG) as described in Chapter 6.5.1. This is because CAI and CAI (HEG) only consider codon bias rather than the “ordering” of those codons in the gene. Two genes can have similar codon bias but different ribosome displacement and translation bottlenecks due to the “ordering” of codons on the gene as explained in the previous paragraph. Unlike CAI and CAI (HEG), BNI takes aminoacyl-tRNA usage into account as well as the “ordering” of codons, which affects the force exerted on the ribosome, ribosome displacement, and translation bottlenecks.

Based on the results observed in this chapter and the literature published by Vu et al. [8], Li et al. [13], Welch et al. [17], Allert et al. [29], Kudla et al. [30], and Tinoco et al. [31] the author of this dissertation contends that the determinants of protein yield from translation elongation are not just codon bias, but, rather, the convolution between aminoacyl-tRNA
usage and additional effects involving the interaction between the 16S rRNA 3’ terminal end and the mRNA, which may lead to a displaced ribosome and, therefore, increased ribosome wait times, bottlenecks, or frameshifts.

Improvements to the current experimental work should consider case studies including: 1) a comparing of codon bias optimization versus model optimization for very long genes, 2) designing a synthetic +1 or -1 frameshift using the model’s principles and verifying the frameshift \textit{in vivo}, 3) optimizing and expressing genes that showed poor translation even when codon bias-optimized [20,22,23], 4) conducting ribosome profiling experiments to determine the accuracy of the model’s predicted ribosome wait times and bottlenecks.
6.7 References


CHAPTER 7

Limitations of the Model and Future Research

7.1 Summary of the Biophysical Model

A new computational-based biophysical model for translation elongation has been developed that provides: a) new insights into the mechanisms behind translation, i.e. elucidating programmed frameshifts prfB and dnaX, and b) a new approach to gene optimization for heterologous protein production. The model introduces two important concepts: i) ribosome displacement and ii) the ribosome “spring” model. Ribosome displacement is the misalignment between the A-site and the normal “zero” reading frame as defined by the start codon. Misalignments are caused by a force exerted on the ribosome from an energetic “spring,” which is derived from the binding between the 16S rRNA 3’ terminal end (anti-Shine-Dalgarno region) and the mRNA. The amount of ribosome displacement is based on three factors: 1) how much the ribosome has already been displaced, 2) the amount of force exerted on the ribosome from an energetic spring, and 3) how long the ribosome has to wait for the delivery of the next aminoacyl-tRNA which is related to aminoacyl-tRNA abundance. These three factors are convolved to describe the ribosome spring model which produces new ribosome wait-time parameters. From these wait-time parameters, translation bottlenecks are identified and corrected to optimize genes for heterologous protein production. The model also includes the ability to manually add ribosome wait-time values at initiation and termination which could produce new bottlenecks at those locations. This ability can be used to improve protein yield predictions because the model does not take into account bottlenecks at initiation or termination. The model also has
the ability to include “noise” from entropy (see Chapter 4.2.3); however, this effect has not yet been explored.

The biophysical model offers many insights into translation elongation. First, minimizing ribosome displacement and translation bottlenecks tend to increase the total protein yield. The model properly demonstrates frameshifts. In addition, the model (BNI) reasonably predicts protein yield; however, predictions may not be perfect. Overall, the model usually provides insight into which genes will translate well and which genes will translate poorly. The model also provides the ability to redesign a given gene for a more optimal translation, and, therefore, increases total protein yield. A distinction between total protein yield and active protein yield is made: total protein yield is active protein yield plus inclusion bodies production. This distinction is made because optimization using the model does not currently guarantee a reduction in inclusion body production and an increase in active protein yield. However, the model has the potential to reduce or eliminate inclusion bodies (see Chapter 6.5.3) through optimizing protein folding by modifying ribosome wait time parameters [8,9,10].

The biophysical model as described in this dissertation and its algorithms and mathematics were incorporated in a fully implemented software package called RiboScan™; this software package will be available as a webserver [1]. Outputs of software include polar plots (see Appendix B), ribosome displacement plots, ribosome wait time plots, translation bottleneck plots, and simulations of single ribosome dynamics during translation elongation.
7.2 Limitations of the Model and Algorithms

The model assumes that unlimited resources are available for protein synthesis. That is, the model does not take into account ribosome abundance and ribosome recycling rates caused by ribosome stalling and dissociation at translation bottlenecks. The model also does not take into account initiation factor abundance or termination factor abundance. Abundance of these factors, or rather a lack of, can contribute to bottlenecks at initiation and termination. The model assumes that aminoacyl-tRNA abundance is static and not dynamic; i.e., the model assumes aminoacyl-tRNA abundance is always available at a certain quantity. Specific aminoacyl-tRNA abundance is dependent on its usage rates compared to its recharge rates. The model does not take into account aminoacyl-tRNA depletion because of amino acid starvation due to overexpression of the mRNA [2,3,4,5]; in reality, the tRNA recharge rate for different cognate tRNA species changes under this condition. However, the model does compensate for aminoacyl-tRNA depletion by using codons that code for multiple tRNA species when optimizing genes.

The model does not include other possible contributions to protein yield. For example, the model does not include extra force exerted on the ribosome from pseudoknots and hairpin structures (see Programmed Frameshifts in Chapter 2.7). The model does not include the mistranslations of amino acids due to ribosome displacement (see Chapter 4.2.2) which can cause misfolding and potentially decrease the production of active proteins. However, the model has the ability to include this effect and will be investigated as part of future research. Bottlenecks at initiation and termination are not automatically taken into account by the model; ribosome wait time values can be manually added at initiation and
termination which could produce new bottlenecks at those locations. Values for ribosome wait time at initiation can be determined by using Salis’s RBS calculator [6,7].

### 7.3 Future Research

Future research using the model includes: 1) analysis of endogenous *E. coli* genes and prediction of their protein synthesis levels, 2) optimization and expression of high value industrial and therapeutetic genes that showed poor translation even when codon bias-optimized [11,12], 3) modification of ribosome wait time parameters to optimize protein folding for decreased or elimination of inclusion body formation [8,9,10], 4) exploration of the effects of “noise” on the translation error rates during elongation, 5) incorporation of mistranslations due to ribosome displacement and exploration of its effects on predicting protein yield, 6) incorporation of biophysical interactions on the ribosome from pseudoknots and hairpins, 7) incorporation of a dynamic aminoacyl-tRNA abundance usage table, and 8) expansion of the biophysical model to different production organisms including eukaryotic expression systems.
7.4 References


Appendix A. Free Energy Calculation

A.1 Introduction

An energy analysis of the binding between two nucleic acid sequences (in Watson-Crick base pairing) to form a secondary structure produces a thermodynamic constant that describes either the amount of energy required for or released by any given reaction. This thermodynamic free energy, $\Delta G$, is measured in units of kcal/mol. Reactions that require free energy have a positive value, and reactions that release free energy have a negative value. Energy must be released to form a stable nucleic acid secondary structure. A more negative free energy corresponds to greater stability in the binding between two nucleic acid sequences. This appendix provides a method of calculating the free energy array from the binding between the 16S “exposed tail” (3’-AUUCCUCCACUAG-5’ for E. coli) and an mRNA sequence, which has been published in [1,2]. To calculate the free energy array, two inputs are needed: the 16S exposed tail and the mRNA sequence. All free energy calculations used in this dissertation came from the Freier free energy model [3]. Other free energy models like Vienna [4], mFold [5], and Aalberts [6] can also be used; preliminary analysis found no major differences between the three free energy models (unpublished). It should be noted that “forced binding” is used for the calculation; i.e., no sliding is allowed between the 13 bases of the 16S exposed tail and the 13 bases of the mRNA sequence. Starma et al. [7] published “Freescan”, a software that calculates the free energy array.
A.2 Calculating the Free Energy Array

The first free energy value ($\Delta G_1$) is calculated from the Watson-Crick binding between the 13 bases of the 16S exposed tail and the first 13 bases of the mRNA starting with the first nucleotide of each codon, as shown in Figure A.1. This is important when calculating the species angle and gene’s phase angle as described in Appendix B. The second free energy value ($\Delta G_2$) is calculated by aligning the 16S tail one base downstream on the mRNA and re-computing the free energy $\Delta G$ from the new binding. This is repeated until the last base of the mRNA is reached; i.e., this is repeated until the 5’ of the exposed tail reaches the 3’ of the mRNA. This creates an array of free energy values resulting from the Watson and Crick binding between the 16S exposed tail and the mRNA (see Figure A.1). This free energy array is used to compute the cumulative phase, cumulative magnitude, polar plot, and phase angle of a gene. Free energy is calculated by using the method published by Mishra et al. [1] or Starma’s “free scan” [7].
Figure A.1. Calculating free energy from exposed tail binding with mRNA. (a) Free energy calculations start at the first nucleotide of each codon as defined by the AUG. Free energy value ($\Delta G_0$) is calculated from the binding between the 13 nucleotides of the 16S rRNA exposed tail and the 13 nucleotides of the mRNA. (b) Free energy value ($\Delta G_1$) is calculated by sliding the 16S rRNA exposed tail one base along the mRNA. (c) Free energy value ($\Delta G_n$) is the last free energy calculation. $\Delta G_n$ is calculated from the binding of the third nucleotide of the last codon on the mRNA; this is usually the stop codon. The result is a free energy array $\{\Delta G_0, \Delta G_1, \ldots, \Delta G_n\}$. 
A.3 References


Appendix B. Polar Plot and Phase Angle

B.1 Polar Plot and Phase Angle

The free energy array for a single gene is intrinsically noisy, hence the need to average this free energy array among many genes as described in [1,2]. To overcome this noise, averaging the free energy over the length of the gene was calculated using cumulative phase and cumulative magnitude approaches [2]. This appendix will describe how to compute the polar plot of a gene, average phase angle of a gene, and corresponding species angle. The polar plot is a graph derived by calculating the free energy array resulting from “scanning” the tail along the mRNA. The methods and equations are published in [2]. $\Phi_k$ and $M_k$ (as computed in Equations (9) and (10) from [1]) are the cumulative phase and magnitude at every codon $k$. It is important to note that $\Phi_k$ and $M_k$ are computed from every first nucleotide of a codon rather than the second or third nucleotide. Computing phase and magnitude from the second or third nucleotide will change the “relative” phase angle, which needs to be “compensated” by a phase shift in order to compare it to $\Phi_k$. $\Phi_k$ and $M_k$ are plotted on a polar plot to show the relative binding position of the 16S rRNA exposed tail to the zero frame at every codon $k$. Drawing the polar plot of $\Phi_k$ and $M_k$ ends at the stop codon. Changes in phase indicate change in the relative binding position of the 16S rRNA tail with respect to reading frames. This is shown in Figure B.1. The average phase angle of the gene (denoted as $\overline{\Theta}$) is calculated by averaging $\Phi_k$ across all codon $k$ (see Equation (B.1)). 
\[ \overline{\Theta} = \frac{\sum_{k=1}^{n} \Phi_k}{n} \] (B.1)

where \( k \) represents the \( k^{th} \) codon, and \( n \) is the total number of codon in the gene.

Figure B.1. Polar plot of \textit{prfB}. \( \Phi_k \) changes 230 degrees at frameshift site, codon 26 [1].

Shown in Figure B.2 is the polar plot of \textit{E. coli’s lacZ} gene, which codes for a \( \beta \)-galactosidase. The polar plot shows normal “in-frame” translation of the \textit{lacZ} gene. The average phase angle of the gene (\( \overline{\Theta} \)) nears the species angle of -25 degrees (see Appendix B.2).
B.2 Species Angle

The species angle ($\theta_{sp}$) is calculated by taking the average phase angle of all non-hypothetical, non-putative, non-pseudo “long” genes (genes greater than 1000 nucleotides) [1]. The average phase angle of long genes was chosen as the species angle because long genes require almost perfect alignment to translate full-length mRNA without errors, i.e. errors can accumulate for very long genes. Therefore the ribosome “spring” needs to be close to “relaxed” throughout translation elongation. This was explained by Vu et al. [3]. The species angle for *E. coli* is -25 degrees.

Figure B.2. Polar plot of *lacZ*. A normal gene that is translated “in-frame.”
B.3 References


Appendix C. Model Analysis of *E. coli* K-12 Genes

C.1 Phase Angle Analysis

All non-pseudo, non-hypothetical, and non-putative genes from *E. coli* K-12 *substr. MG1655* genome (RefSeq: NC_000913.3) from GenBank [1] were plotted on the polar plot (see Figure C.1). The phase angle for those genes ranges from +20 degrees to -60 degrees. This is defined as the “working” range for the phase angle of *E. coli*. The average phase angle of the genes ($\bar{\Theta}$) was also plotted for increasing gene lengths using box plots. This plot is shown in Figure C.2. As the gene length increases, the average phase angle of the genes converges to -25 degrees, which is the species angle. Highly expressed genes of *E. coli* that were identified by Karlin et al. [2] were also plotted on the polar plot (see Figure C.3). The phase angle of these highly expressed genes converges to -45 degrees. The “correct” phase angle is the species angle which is explained in Appendix B.2 and Vu et al. [3]. Therefore, optimizing based on the codon bias of highly expressed genes uses the incorrect phase angle (-45 degrees) and could lead to a displaced ribosome.
Figure C.1. Polar plot of *E. coli* genes from GenBank. Genes were non-pseudo, non-hypothetical, and non-putative from GenBank [1]. The phase angle for those genes ranges from +20 degrees to -60 degrees. This is defined as the “working” range for the phase angle of *E. coli*.

Figure C.2. Box plot of genes’ average phase angle with respect to length. Genes were non-pseudo, non-hypothetical, and non-putative taken from GenBank [1]. As the gene length increases, the average phase angle, $\Theta$, of the genes converges to -25 degrees, which is the species angle.
C.2 Ribosome Displacement Analysis

All non-pseudo, non-hypothetical, and non-putative genes from *E. coli K-12 substr. MG1655* genome (RefSeq: NC_000913.3) from GenBank [1] were plotted on the displacement plot (see Figure C.4). All genes stayed within or very close to -1 to 1 displacement units.

![Figure C.4. Displacement plot of *E. coli* genes from GenBank. Genes were non-pseudo, non-hypothetical, and non-putative from GenBank [1]. All genes stayed within or very close to -1 to 1 displacement units.](image)
C.3 Highly Expressed *E. coli* Genes

Highly expressed genes showed smaller BNI than poorly expressed genes.

C.3.1 *tufA*

![Polar plot of tufA](image1.png)

**Figure C.5. Polar plot of tufA.** The average phase angle of *tufA* is ~45 degrees.

![Ribosome displacement plot of tufA](image2.png)

**Figure C.6. Ribosome displacement plot of tufA.** This plot shows the displacement of the ribosome during translation of the *tufA* gene.
Figure C.7. Ribosome wait time plot of \textit{tufA}. Total Wait Time for translation of \textit{tufA} is 1794 cycles.

Figure C.8. Translation bottleneck plot of \textit{tufA}. The BNI for \textit{tufA} is 142 at codon 67.
C.3.2 $ompA$

Figure C.9. Polar plot of $ompA$. The average phase angle of $ompA$ is -49 degrees.

Figure C.10. Ribosome displacement plot of $ompA$. This plot shows the displacement of the ribosome during translation of the $ompA$ gene.
Figure C.11. Ribosome wait time plot of \textit{ompA}. Total Wait Time for translation of \textit{ompA} is 1857 cycles.

Figure C.12. Translation bottleneck plot of \textit{ompA}. The BNI for \textit{ompA} is 150 at codon 197.
C.3.3 fusA

Figure C.13. Polar plot of fusA. The average phase angle of fusA is -45 degrees.

Figure C.14. Ribosome displacement plot of fusA. This plot shows the displacement of the ribosome during translation of the fusA gene.
Figure C.15. Ribosome wait time plot of *fusA*. Total Wait Time for translation of *fusA* is 3725 cycles.

Figure C.16. Translation bottleneck plot of *fusA*. The BNI for *fusA* is 147 at codon 682.
C.4 Poorly Expressed *E. coli* Genes

Poorly expressed genes showed greater BNI than highly expressed genes.

C.4.1 *galR*

![Polar plot of *galR*](image1.png)

*Figure C.17. Polar plot of *galR*. The average phase angle of *galR* is -29 degrees.*

![Ribosome displacement plot of *galR*](image2.png)

*Figure C.18. Ribosome displacement plot of *galR*. This plot shows the displacement of the ribosome during translation of the *galR* gene.*
Figure C.19. Ribosome wait time plot of *galR*. Total Wait Time for translation of *galR* is 2082 cycles.

Figure C.20. Translation bottleneck plot of *galR*. The BNI for *galR* is 180 at codon 120.
C.4.2 lacI

Figure C.21. Polar plot of lacI. The average phase angle of lacI is 8 degrees.

Figure C.22. Ribosome displacement plot of lacI. This plot shows the displacement of the ribosome during translation of the lacI gene.
Figure C.23. Ribosome wait time plot of *lacI*. Total Wait Time for translation of *lacI* is 2258 cycles.

Figure C.24. Translation bottleneck plot of *lacI*. The BNI for *lacI* is 213 at codon 213.
C.4.3 dnaE

Figure C.25. Polar plot of dnaE. The average phase angle of dnaE is -17 degrees.

Figure C.26. Ribosome displacement plot of dnaE. This plot shows the displacement of the ribosome during translation of the dnaE gene.
Figure C.27. Ribosome wait time plot of dnaE. Total Wait Time for translation of dnaE is 6634 cycles.

Figure C.28. Translation bottleneck plot of dnaE. The BNI for dnaE is 214 at codon 805.
C.5 References


Appendix D. DNA Sequences of Heterologous Genes

The DNA sequences of all genes and variants are provided here. These genes include 

\( \text{gst} \) (glutathione S-transferase), \( \text{pf0132} \) (alpha-glucosidase), \( \text{clju_c11880} \) (alcohol dehydrogenase), \( \text{bif1} \) (beta-galactosidase), \( \text{rt8_0542} \), and all of their respective variants.

D.1 \( \text{gst} \), Glutathione S-Transferase used for Prediction

\[
\text{\textit{gst Wildtype}}
\]

\[
\text{\textit{gst Variant 16}}
\]

\[
\text{\textit{gst Variant 23}}
\]

\[
\text{\textit{gst Variant 80}}
\]
> **gst Variant 82**

atgagccctatactaggttagttaggaataataaagggcggtgcttcgtcgtggaacctccacactcgactctttggaatatctgtaaggaagataaggcgtcgcgcgaagagaaggtgacgttaaactgacactcagtctatggcgatcatcaggtacatcgcggacaaacacaacatgctgggtggttgcccgaaagaacgtgcggaaatctctatgctggaaggtgcgttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaaacttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacctcaccacccggacttcatgctgtacgacgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggctaccttcggtggtggtgaccacccgccgaaatctgacggttctacctag

> **gst Variant 0D**

atgagccctatactaggttagttaggaataataaagggcggtgcttcgtcgtggaacctccacactcgactctttggaatatctgtaaggaagataaggcgtcgcgcgaagagaaggtgacgttaaactgacactcagtctatggcgatcatcaggtacatcgcggacaaacacaacatgctgggtggttgcccgaaagaacgtgcggaaatctctatgctggaaggtgcgttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaaacttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacctcaccacccggacttcatgctgtacgacgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggctaccttcggtggtggtgaccacccgccgaaatctgacggttctacctag

> **gst Variant 90D**

atgagccctatactaggttagttaggaataataaagggcggtgcttcgtcgtggaacctccacactcgactctttggaatatctgtaaggaagataaggcgtcgcgcgaagagaaggtgacgttaaactgacactcagtctatggcgatcatcaggtacatcgcggacaaacacaacatgctgggtggttgcccgaaagaacgtgcggaaatctctatgctggaaggtgcgttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaaacttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacctcaccacccggacttcatgctgtacgacgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggctaccttcggtggtggtgaccacccgccgaaatctgacggttctacctag

> **gst Variant M**

atgagccctatactaggttagttaggaataataaagggcggtgcttcgtcgtggaacctccacactcgactctttggaatatctgtaaggaagataaggcgtcgcgcgaagagaaggtgacgttaaactgacactcagtctatggcgatcatcaggtacatcgcggacaaacacaacatgctgggtggttgcccgaaagaacgtgcggaaatctctatgctggaaggtgcgttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaaacttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacctcaccacccggacttcatgctgtacgacgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggctaccttcggtggtggtgaccacccgccgaaatctgacggttctacctag
> **gst Variant VM** (Same as M, but same first 90 bases same as wildtype)

attgaagctattccacaaattgataagtacttgaaaagcagcaagtatatagcatggcctttgcaggggtggcaagccacgtttggtggtggcgatcatccgccgaatcggatggttcaacctag

gst

Wildtype

atgtctccgattctgggttattgggaagatcaaaggtctcgttcagccgacgCGActtcttttggaatatcttgaagaaaaatatgaagagcatttgtatgagcgcgatgaaggtgataaatggcgtaacaaaaaattcgaactgggcctggaatttcgaacctgcgtactacatcgacggtgacgttaacactgacccaatctatggcgatcatccgctatatcgcagataacacaacatgctgggtggttgcccgaaagaacgcgctgaaatctctatgctggaaggtgcggttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaagttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacccacccgacttcatgctgtacgcgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggcgaccttcggtggtggtgaccacccgccgaaatctgacggttctacctaa

gst

Model-optimized

atgtctccgattctgggttattgggaagatcaaaggtctcgttcagccgacgCGCctgctgctggaatacctggaagaaaaatacgaagaacacctgtacgaacgcgatgaaggtgacaaatggcgtaacaagaaattcgaactgggcctggaatttccgaacctgccgtattacatcgacggtgacgttaacactgacccaatctatggcgatcatccgctatatcgcagataacacaacatgctgggtggttgcccgaaagaacgcgctgaaatctctatgctggaaggtgcggttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaagttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacccacccgacttcatgctgtacgcgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggcgaccttcggtggtggtgaccacccgccgaaatctgacggttctacctaa

> **gst Codon bias-optimized 1**

atgtctccgattctgggttattgggaagatcaaaggtctcgttcagccgacgCGTctgctgctggaatacctggaagaaaaatacgaagaacacctgtacgaacgcgatgaaggtgacaaatggcgtaacaagaaattcgaactgggcctggaatttccgaacctgccgtattacatcgacggtgacgttaacactgacccaatctatggcgatcatccgctatatcgcagataacacaacatgctgggtggttgcccgaaagaacgcgctgaaatctctatgctggaaggtgcggttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaagttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacccacccgacttcatgctgtacgcgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggcgaccttcggtggtggtgaccacccgccgaaatctgacggttctacctaa

> **gst Codon bias-optimized 2**

atgtctccgattctgggttattgggaagatcaaaggtctcgttcagccgacgCGCctgctgctggaatacctggaagaaaaatacgaagaacacctgtacgaacgcgatgaaggtgacaaatggcgtaacaagaaattcgaactgggcctggaatttccgaacctgccgtattacatcgacggtgacgttaacactgacccaatctatggcgatcatccgctatatcgcagataacacaacatgctgggtggttgcccgaaagaacgcgctgaaatctctatgctggaaggtgcggttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaagttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacccacccgacttcatgctgtacgcgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggcgaccttcggtggtggtgaccacccgccgaaatctgacggttctacctaa

D.2 **gst**, Glutathione S-Transferase used for Optimization

> **gst** Variant VM (Same as M, but same first 90 bases same as wildtype)

atgagccctatactaggttattggaaaattaagggccttgtgcaacccactcgacttcttttggaatatcttgaagaaaaatatgaagagcatttgtatgagcgcgatgaaggtgataaatggcgtaacaaaaaattcgaactgggcctggaatttccgaacctgccgtattacatcgacggtgacgttaacactgacccaatctatggcgatcatccgctatatcgcagataacacaacatgctgggtggttgcccgaaagaacgcgctgaaatctctatgctggaaggtgcggttctggacatccgttacggtgtttctcgtatcgcgtactctaaagacttcgaaaccctgaaagttgacttcctgtctaaactgccggaaatgctgaaaatgttcgaagaccgtctgtgccacaaaacctacctgaacggtgaccacgttacccacccgacttcatgctgtacgcgcgctggacgttgttctgtacatggacccgatgtgcctggacgcgttcccgaaactggtttgcttcaaaaaacgtatcgaagcgatcccgcagatcgacaaaatacctgaaatcttctaaatacatcgcgtggccgctgcagggttggcaggcgaccttcggtggtggtgaccacccgccgaaatctgacggttctacctaa
D.3 clju_c11880, Alcohol Dehydrogenase

> clju_c11880 Wildtype

D.3 clju_c11880, Alcohol Dehydrogenase

> clju_c11880 Model-optimized
D.4 pf0132, Alpha-Glucosidase

> pf0132 Wildtype

gttgaaaaagattccagatatcccccaagatggttccagagatggggagttgaggtttatctgccacaaatggaacgtcatataatggaattcctgaaatgagttcttcttaaaagagttcagttgtaatcgatcattctgataagttcagttgtaatggttcgcggcatggtttccatgattttccttttcccagcggatggtttcggagtttcgagtttttcttttcgagttttttcttttctttttcttttctttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt
**D.5 *bif1*, Beta-Galactosidase**

> **Wildtype**

\[\text{atgtttatcccccgctattacgaatctctcggtcacctacacgtcggtACGcagccgaaccgcgcgtattacgtgcccgcgtccacg}
\[\text{ccgatggacacggtcggcgagaatcgcgtgaattccgatcgtttcatgctgctgaacggcgactgggatttcaagtattatgcgagcat}
\[\text{ctacgacttggacgacagtcgagctgcgtctgtgtgcgagctggtggttcgctgctatgtgcgttgcgtaatctggtcgtgtttcat}
\[\text{gtgcgatgtgccgacatcgcgactgttggcgaaaaccgcgttaacagcgaccgatttatgctgctcaacggtgattgggattttaagtactatgcttcgatctatgat}

> **Model-optimized**

\[\text{atgtttatcccccgctattacgaatctctcggtcacctacacgtcggtacgcagcctaatcgcgcatactatgttccggcaagcactccta}
\[\text{tggacactgttggcagcctgatcgcgccgtgcgatgacgatggagcgaacgatgccgcggctacgacgacgtcgtgctggcagacccgctgaccgtcatgcagaccgatgcctcgatcaccgtggaaggctcgacgttccggtatgt}
\[\text{gctcgacccgtaccggccgttctcgtcgatgtcgttcgcaaccgatcgctgcttaacaggccgatggagctgaacgtgtggcgcgct}
\[\text{cccaccgacaacgaccagtacactcaaggccgattggattcgcgcccagtataa}
bif1 Codon bias-optimized

agtgtaccccccgttattacgcataactctcgtcgacccacacgtctctcattttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt
D.6  *rt8* _0542, Endoglucanase and Exoglucanase

> *rt8* _0542_ Wildtype

The text contains a detailed genetic sequence related to the *rt8* _0542_ gene. It appears to be the result of a genetic experiment involving the *rt8* _0542_ gene, which is related to endoglucanase and exoglucanase activities. The text is dense and technical, indicating a focus on the genetic manipulation and expression of these enzymes.
aagctttgagtttgagcacaattgcccaggaagagcatacctggtcattgctcattgtcgccttttgcagacattttgaaatcgttgtttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt...
gtcagacgcaggaaggaatctctgtgtgcttcttacacatcagagatctactgctgcaagcgttactctt
cgttatagttgatcagagcaggaagtctctgtgctttcactacgtacttatgttctcctcgttaatgtt
ttctcctgctggctgtattgtgtctcttgttttcgtttgtggtttgtggttttgttttgttgggggt
ggttggtggttgttttttggttttctcttgcgaggaacagtctctgtggtttgtgtttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt
gtttactcagactggtcaggctgcgccagtaaaaaagatacatatcgtaaattcggcgaatcccaaatcgattatgccctgggttca
gctggtcggtcattcgtcgtcggtttcgtgaccaacccaccgaaacgcccagactctgacctctgacctctgacctctgacctctgac
gtactacccatcagactggtcaggctgcgccagtaaaaaagatacatatcgtaaattcggcgaatcccaaatcgattatgccctgggttca
Appendix E. Laboratory Experimental Procedures

E.1 Cloning Genes into pBAD-Myc-His Plasmid

Genes from Appendix D, including the start and stop codon, were synthesized and cloned into the pBAD/Myc-His C plasmid (available from Thermo Fisher Scientific, Inc., Catalog Number: V440-01) using GATEWAY cloning technique by either Genewiz, Inc. or GenScript, Inc. Insertion of the gene sequence into the plasmid directly replaced the ATG at the NcoI site, as shown in Figure E.1.

![Figure E.1. Insertion of gene into pBAD/Myc-His C plasmid.](image)


E.2 Transformation into Chemically Competent TOP10 Cells

After gene synthesis and cloning into pBAD/Myc-His C by either GenScript, Inc. or Genewiz, Inc., the plasmid (containing the gene insert) was chemically transformed into One Shot® TOP10 Chemically Competent E. coli cells (available from Thermo Fisher Scientific,
Inc., Catalog Number: C4040-03) according to the manufacturer’s instructions. 1 ul of DNA plasmid at a concentration of 10 ng/ul were pipetted into a pre-aliquoted vial of competent cells followed by incubation on ice for 10 minutes and subsequent heat-shock at 42°C for 30 seconds. Vials were then flicked for mixing and placed on ice for 5 minutes. 250 uL of Super optimal catabolite (S.O.C) broth was added to each vial. 25 uL of transformed cells in S.O.C. were added to LB-amp agar plates (50 ug/ml of ampicillin; see Appendix E.9). Plates were incubated at 37°C for 16 hours and subsequently stored at 4°C. Cells were frozen at -80°C in a 75% LB, 25% glycerol mixture with 50 ug/ml of ampicillin for long term storage (glycerol freezer stocks).

E.3 Inducing pBAD/Myc-His with Gene Insert

TOP10 E. coli cells from glycerol freezer stocks (containing the pBAD/Myc-His plasmid with the gene insert) were T-streaked on LB-amp agar plates (containing 50 ug/ml of ampicillin; see Appendix E.9). Isolated colonies were selected and incubated overnight in LB-amp medium (containing 50 ug/ml of ampicillin; see Appendix E.8) in conical flasks no smaller than 5X larger than medium volume used. Samples were incubated in a water bath shaker (Model G76D from New Brunswick™ Digital) for 16 hours at 37°C at 210 rpm.

Sufficient overnight culture was then added to fresh LB-amp medium (containing 50 ug/ml of ampicillin; see Appendix E.8) to attain optical density (OD) reading of 0.10 at 600 nm in 1 cm cuvette using a Biowave Cell Density Meter CO8000 from Biochrom. Samples were then incubated in the water bath shaker at the same conditions until OD reading was 0.50. 100 ul of D-arabinose was added from stock solutions ranging from 2% to 20% w/v to create a final concentration ranging from 0.02% w/v to 0.2% w/v (0.2 mg/ml to 20 mg/ml),
and samples were incubated in the water bath shaker at the aforementioned conditions for 3 or 4 hours to induce expression of pBAD vector. Cell pellets were collected by centrifugation (Z 216 MK Refrigerated Micro-centrifuge from HERMLE Labortechnik) at 10,000G at 4°C for 10 minutes, and supernatant was removed from either 1 or 2 ml aliquots of sample volume. Samples were subsequently stored at -20°C before protein quantification.

E.4 GST Protein Quantification

Activity Assay

Cell pellets from induction were re-suspended in 1 ml of EMD Millipore’s BugBuster™ Protein Extraction Reagent (Catalog Number: 70584). 3 ul of EMD Millipore’s Lysonase™ Bioprocessing Reagent (Catalog Number: 71230) per 1 ml of BugBuster™ was added to each cell re-suspension; Samples were placed on a shaker (55S Single Platform Shaker from Reliable Scientific, Inc.) at medium speed at room temperature for 20 min and subsequently centrifuged at 16,000G at 4°C for 20 min.

Glutathione S-transferase activity (E.C. 2.5.1.18) was quantified in 96-well microplates using GST assay kit from Sigma Aldrich, Inc. (Catalog Number: CS0410). 10 uL of each cell lysate was aliquoted in three wells as triplicates. 190 uL of assay mixture containing [9.8:1:1] mixture of [Dulbecco’s Phosphate Buffered Saline:100 mM 1-chloro-2,4-dinitrobenzene (CDNB):200 mM L-glutathione reduced] (all from Sigma-Aldrich, Inc.) was added to each well. GST produces GS-DNB using L-Glutathione and CDNB as substrates. The GST assay kit measures the production of GS-DNB conjugate, which absorbs light at 340 nm. Ten absorbance measurements at 340 nm are made at 30 second intervals using a POLARstar Galaxy microplate reader from BMG LABTECH, Inc. The first
measurement was made exactly 1 minute after mixing assay mixture and cell lysate. GST activity is directly proportional to the rate of increase in absorbance at 340 nm per minute (ΔA₃₄₀nm/minute). Therefore, GST activity was measured as the slope of the ten absorbance measurements with respect to time (in units of ΔA₃₄₀nm/minute). GST specific activity was calculated using the follow formula:

\[
\frac{\Delta A_{340\text{nm}}}{\text{min}} \cdot V(\text{ml}) \cdot \text{dil} \cdot \epsilon_{mM} \cdot V_{\text{enz}}(\text{ml}) = \frac{\text{umol}}{\text{ml} \cdot \text{min}} = \frac{\text{Units of GST}}{\text{ml}} \tag{E.1}
\]

One unit of GST is defined as the amount of GST enzyme producing 1 umol of GS-DNB conjugate per minute at 25°C. \(\text{dil}\) is the dilution factor of original sample; \(\epsilon_{mM}\) is the millimolar extinction coefficient for GS-DNB conjugate at 340 nm; \(V\) is the reaction volume; and \(V_{\text{enz}}\) is the volume of the sample tested. The milli-molar extinction coefficient \((\epsilon_{mM})\) for GS-DNB conjugate at 340 nm is 9.6 mM⁻¹ for a path length of 1 cm. Under the aforementioned assay conditions, \(\text{dil} = 1\), \(\epsilon_{mM} = 6.04\), \(V (\text{ml}) = 0.2\), and \(V_{\text{enz}} (\text{ml}) = 0.01\). \(\frac{\text{Units of GST}}{\text{ml}}\) were multiplied by 0.2 ml (cell pellet resuspension volume) to compute the units of GST per cell pellet.

GST specific activity was normalized to total protein using Pierce™ BCA Protein Assay Kit (available from Thermo Fisher Scientific, Inc., Catalog Number: 23225). Total protein normalization was performed to standardize for deviations in the number of cells collected. Total protein content was measured by absorbance of light at 562nm (A₅₆₂nm) according to the manufacturer’s instructions for the microplate procedure. The same microplate reader (POLARstar Galaxy microplate reader) was used, and A₅₆₂nm
measurements were made after 1 hour of incubation at room temperature. Normalized GST activity was computed by dividing GST specific activity with total protein measurements ($A_{562nm}$) for the same cell pellet. Units for normalized GST activity are defined as units of GST per $A_{562nm}$ of total protein where one unit of $A_{562nm}$ equals 18 mg of total protein using the BSA standard curve. All assays were conducted in triplicates and measured within the linear range of the assay.

**E.5 PF0132 Protein Quantification**

**SDS-PAGE**

Gene *pf0132* and variants expressed as insoluble inclusion bodies. Soluble fraction and inclusion body purification were performed using EMD Millipore’s BugBuster™ Protein Extraction Reagent (Catalog Number: 70584) and EMD Millipore’s Lysonase™ Bioprocessing Reagent (Catalog Number: 71230) according to the manufacturer's instructions following a 4 hour induction at a 0.2% w/v (2 mg/ml) arabinose concentration. Wash and resuspension volumes ranging from 200 uL to 1 ml were used to vary purified protein concentration.

Inclusion body and water-soluble fraction of wildtype and model-optimized *pf0132* were visualized on SDS-PAGE gels. Samples were mixed with equal volume of 2x Laemmlig Sample Buffer (available from Bio-Rad, Inc., Catalog Number: 1610737) prepared with 5% 2-mercaptoethanol. Samples were incubated in boiling water for 5 minutes. 30ul of samples were loaded and run on a NuSep’s Tris-Glycine NB 4-20% precast SDS-PAGE gel (Catalog Number: NB10-008) in a Mini-PROTEAN® Tetra Electrophoresis System by Bio-Rad, Inc. PageRuler Prestained Protein Ladder (available from Thermo Fisher Scientific, Inc., Catalog
Number: 26616) was used as a protein size standard. The gels were run at 125 V for approximately 2 hours. The completed gel was placed on a shaker at low speed in NuBlu Express Stain according to manufacturer’s instruction to visualize protein bands.

E.6 CLJU_C11880 Protein Quantification Activity Assay

Cell pellets from induction were re-suspended in 200 ul of EMD Millipore’s BugBuster™ Protein Extraction Reagent (Catalog Number: 70584). 3 ul of EMD Millipore’s Lysonase™ Bioprocessing Reagent (Catalog Number: 71230) per 1 ml of BugBuster™ were added to each cell resuspension. Samples were placed on a shaker at medium speed at room temperature for 20 minutes and subsequently centrifuged at 16,000G at 4°C for 20 minutes.

CLJU_c11880 (alcohol dehydrogenase, ADH) activity levels were quantified using the alcohol dehydrogenase (EC 1.1.1.1) enzymatic assay protocol available from Sigma Aldrich, Inc. [1]. All assay conditions were conducted per manufacturer’s instructions with contents reduced tenfold for use on 96-well microplate. ADH converts alcohol to aldehyde, with the concomitant reduction of NAD⁺ to NADH. The ADH assay measures the reduction NAD⁺ to NADH; NADH absorbs light at 340 nm. Ten absorbance measurements at 340 nm were made at 30-second intervals using a microplate reader. The first measurement was made exactly 1 minute after combining assay mixture and cell lysate. ADH activity is directly proportional to the rate of increase in absorbance at 340 nm per minute (ΔA₃₄₀nm/minute). Therefore, ADH activity was measured as the slope of the ten absorbance measurements with respect to time (in units of ΔA₃₄₀nm/minute). ADH specific activity was calculated using the follow formula:
\[
\frac{\Delta A_{340\text{nm}} \cdot V(\text{ml}) \cdot \text{dil}}{\varepsilon_{mM} \cdot V_{\text{enz}}(\text{ml})} = \frac{\text{umol}}{\text{ml} \cdot \text{min}} = \frac{\text{Units of ADH}}{\text{ml}}
\] (E.2)

One unit of ADH is defined as the amount of ADH enzyme producing 1 umol of NADH per minute at 25°C. \(\text{dil}\) is the dilution factor of original sample; \(\varepsilon_{mM}\) is the milli-molar extinction coefficient for NADH at 340 nm; \(V\) is the reaction volume; and \(V_{\text{enz}}\) is the volume of the sample tested. The milli-molar extinction coefficient \((\varepsilon_{mM})\) for NADH at 340 nm is 6.22 mM\(^{-1}\) for a path length of 1 cm. Under the aforementioned assay conditions, \(\text{dil} = 1\), \(\varepsilon_{mM} = 5.47\), \(V(\text{ml}) = 0.3\), and \(V_{\text{enz}}(\text{ml}) = 0.01\). \(\frac{\text{Units of ADH}}{\text{ml}}\) were multiplied by 0.2 ml (cell pellet resuspension volume) to compute the units of ADH per cell pellet.

ADH specific activity was normalized to total protein using Pierce™ BCA Protein Assay Kit (available from Thermo Fisher Scientific, Inc., Catalog Number: 23225). Total protein normalization was performed to standardize for deviations in the number of cells collected. Total protein content was measured by absorbance of light at 562 nm \((A_{562\text{nm}})\) according to the manufacturer’s instructions for the microplate procedure; the same microplate reader (POLARstar Galaxy microplate reader) was used. \(A_{562\text{nm}}\) measurements were made after 1 hour of incubation at room temperature. Normalized ADH activity was computed by dividing ADH specific activity with total protein measurements \((A_{562\text{nm}})\) for the same cell pellet. Units of normalized ADH activity are defined as units of ADH per \(A_{562\text{nm}}\) of total protein where one unit of \(A_{562\text{nm}}\) equals 18 mg of total protein using the BSA standard curve. All assays were conducted in triplicates and measured within linear range of the assay.
Both wildtype and optimized clju_c11880 also expressed as insoluble inclusion bodies. Water-soluble fractions and inclusion body purification were performed using BugBuster™ Protein Extraction Reagent and Lysonase™ Bioprocessing Reagent according to the manufacturer's instructions following a 4 hour induction at a 0.2% w/v (2 mg/ml) arabinose concentration. Wash and resuspension volumes ranging from 200 uL to 400 ul were used to vary purified protein concentration for visualization on SDS-PAGE gel.

Inclusion body, whole cell lysate, and water-soluble fraction of wildtype and model-optimized clju_c11880 were visualized on SDS-PAGE gels. Samples were mixed with equal volume of with 2x Laemmli Sample Buffer (available from Bio-Rad, Inc., Catalog Number: 1610737) prepared with 5% 2-mercaptoethanol. Samples were incubated in boiling water for 5 minutes. 30ul of samples were loaded and ran on a NuSep’s Tris-Glycine NB 4-20% precast SDS-PAGE gel (Catalog Number: NB10-008) in a Mini-PROTEAN® Tetra Electrophoresis System by Bio-Rad, Inc. PageRuler Prestained Protein Ladder (available from Thermo Fisher Scientific, Inc., Catalog Number: 26616) was used as a protein size standard. The gels were run at 175 V for approximately 1 hour. The completed gel was placed on a shaker at low speed in NuBlu Express Stain to visualize protein bands.
E.7 BIFI Protein Quantification

Activity Assay

Cell pellets from induction were re-suspended in 300 uL of EMD Milipore Inc.’s BugBuster™ Protein Extraction Reagent (Catalog Number: 70584) with 0.9 uL of EMD Millipore Inc.’s Lysonase™ Bioprocessing Reagent (Catalog Number: 71230). Samples were placed on a shaker at medium speed at room temperature for 20 minutes and subsequently centrifuged at 16,000G at 4°C for 20 minutes.

BIFI (β-galactosidase) activity was quantified using the β-galactosidase reporter gene activity detection kit from Sigma-Aldrich, Inc. (Catalog Number: GALA) per manufacturer’s instructions for use on a 96-well microplate. β-galactosidase hydrolyzes β-galactosides into monosaccharides. The β-galactosidase assay kit uses o-nitrophenyl-β-D-galactopyranoside (ONPG) as the substrate to quantify β-galactosidase activity. ONPG, a colorless substrate, is hydrolyzed by β-galactosidase to produce yellow colored o-nitrophenol that absorbs light at 420 nm. β-galactosidase activity was quantified using the absorbance of light at 420 nm after 30 minute incubation at 37°C. Activity measurements were performed using a POLARstar Galaxy microplate reader from BMG LABTECH, Inc. β-galactosidase (β-Gal) specific activity was calculated using the follow formula:

\[
\frac{A_{420nm} \cdot V (ml) \cdot dil}{
\frac{\epsilon_{mM} \cdot V_{enz}(ml)}{30 \text{ min}}} = \frac{\text{umol}}{ml \cdot min} = \frac{\text{Units of } \beta-\text{Gal}}{ml}
\]  

(E.3)

One unit of β-galactosidase is defined as the amount of β-galactosidase enzyme producing 1 umol of o-nitrophenol per minute at 37°C. dil is the dilution factor of original sample; \(\epsilon_{mM}\) is
the milli-molar extinction coefficient of o-nitrophenol at 420 nm; \( V \) is the reaction volume; and \( V_{\text{enz}} \) is the volume of the sample tested. The milli-molar extinction coefficient (\( \varepsilon_{\text{mM}} \)) for o-nitrophenol at 420 nm is 4.6 mM\(^{-1}\) for a path length of 1 cm. Under the conditions the assays were conducted, \( \text{dil} = 1, \varepsilon_{\text{mM}} = 3.47, V (ml) = 0.25, \) and \( V_{\text{enz}} (ml) = 0.05 \). Units of \( \beta\text{-Gal} \) was multiplied by 0.3 ml (cell pellet resuspension volume) to compute the units of \( \beta \)-galactosidase per cell pellet.

\( \beta \)-galactosidase specific activity was normalized to total protein using Pierce™ BCA Protein Assay Kit (available from Thermo Fisher Scientific, Inc., Catalog Number: 23225). Total protein normalization was performed to standardize for deviations in the number of cells collected. Total protein content was measured by absorbance of light at 562 nm (\( A_{562\text{nm}} \)) according to the manufacturer’s instructions for the microplate procedure; the same microplate reader (POLARstar Galaxy microplate reader) was used. \( A_{562\text{nm}} \) measurements were made after 1 hour of incubation at room temperature. Normalized \( \beta \)-galactosidase activity was computed by dividing \( \beta \)-galactosidase specific activity with total protein measurements (\( A_{562\text{nm}} \)) for the same cell pellet. Units of normalized \( \beta \)-galactosidase activity are defined as units of \( \beta \)-galactosidase per \( A_{562\text{nm}} \) of total protein where one unit of \( A_{562\text{nm}} \) equals 18 mg of total protein using the BSA standard curve. All assays were conducted in triplicates and measured within the linear range of the assay.

**SDS-PAGE**

Gene \( bif1 \) and variants also expressed as insoluble inclusion bodies. Soluble fraction and inclusion body purification were performed using EMD Millipore’s BugBuster™ Protein
Extraction Reagent according to the manufacturer's instructions following a 4 hr induction at a 0.2% w/v (2 mg/ml) arabinose concentration. Wash and resuspension volumes ranging from 200 uL to 400 uL were used to vary purified protein concentration.

Inclusion body, whole cell lysate, and soluble fraction samples of BIFI and variants were used with SDS-PAGE. Samples were mixed with equal volume of with 2x Laemmli Sample Buffer (available from Bio-Rad, Inc., Catalog Number: 1610737) prepared with 5% 2-mercaptoethanol. Samples were incubated in boiling water for 5 minutes. 30ul of samples were loaded and ran on a NuSep’s Tris-Glycine NB 8% precast SDS-PAGE gel (Catalog Number: NB10-008) in a Mini-PROTEAN® Tetra Electrophoresis System by Bio-Rad, Inc. PageRuler Prestained Protein Ladder (available from Thermo Fisher Scientific, Inc. Catalog Number: 26616) was used as a protein size standard. The gels were run at 175 V for approximately 1 hour. The completed gel was placed on a shaker at low speed in Generon’s Quick Coomassie Stain (Catalog Number: GEN-QC-STAIN) overnight to visualize protein bands.

E.8 Lysogeny Broth (LB) Medium

LB was prepared with 1% tryptone, 0.5% yeast extract, and 0.5% NaCl. Mixture pH was adjusted to 7.0 with 5M NaOH and autoclaved for 20 minutes. Ampicillin was added to final concentration of 50 ug/ml before use. Final broth was henceforth called LB-amp medium.
E.9 Lysogeny Broth with Ampicillin (LB-amp) Agar Plates

Lysogeny broth with ampicillin (LB-amp) agar plates was prepared from tablets (Catalog Number: L7025, from Sigma-Aldrich, Inc.) and de-ionized water according to the manufacturer’s instructions with the addition of ampicillin to a final concentration of 50 ug/ml after cooling from autoclave sterilization and before pouring plates.
E.10 References