

Utah State University

DigitalCommons@USU

---

All Graduate Theses and Dissertations

Graduate Studies

---

5-2014

## Metabolic Modeling of Spider Silk Production in *E. coli*

Sarah Allred  
*Utah State University*

Follow this and additional works at: <https://digitalcommons.usu.edu/etd>



Part of the [Biological Engineering Commons](#)

---

### Recommended Citation

Allred, Sarah, "Metabolic Modeling of Spider Silk Production in *E. coli*" (2014). *All Graduate Theses and Dissertations*. 3892.

<https://digitalcommons.usu.edu/etd/3892>

This Thesis is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Theses and Dissertations by an authorized administrator of DigitalCommons@USU. For more information, please contact [digitalcommons@usu.edu](mailto:digitalcommons@usu.edu).



METABOLIC MODELING OF SPIDER SILK PRODUCTION IN *E. COLI*

by

Sarah Allred

A thesis submitted in partial fulfillment  
of the requirements for the degree of

MASTER OF SCIENCE

in

Biological Engineering

Approved:

---

H. Scott Hinton  
Major Professor

---

Dr. Ronald C. Sims  
Committee Member

---

Dr. Randy Lewis  
Committee Member

---

Dr. Charles Miller  
Committee Member

---

Dr. Mark McLellan  
Vice President for Research and  
Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY  
Logan, Utah

2014

Copyright © Sarah Allred 2014

All Rights Reserved

## ABSTRACT

Metabolic Modeling of Spider Silk in *E. coli*

by

Sarah Allred, Master of Science

Utah State University, 2014

Major Professor: H. Scott Hinton  
Department: Biological Engineering

Spider silk has drawn much attention in recent years because of its many unique and remarkable mechanical properties. Great elasticity coupled with high tensile strength, toughness, biocompatibility, and biodegradability are all contributing factors to the interest in spider silk for medical applications and as a biomaterial. It is not feasible to harvest natural spider silk from spiders because of their aggressive and territorial behavior, so recombinant spider silk proteins are created in host systems, such as *Escherichia coli* (*E. coli*), and the resulting proteins are then spun into artificial fibers. However, spider silk production in *E. coli* is not efficient enough for large-scale manufacturing. Metabolic modeling can help assist metabolic engineering methods to increase the production of spider silk. A metabolic modeling tool known as dynamic FBA correctly predicted the need to increase the ammonium concentration in the cell medium. As a result of increasing the ammonium concentration in laboratory fermentors, the spider silk production increased significantly.

(93 pages)

## PUBLIC ABSTRACT

Metabolic Modeling of Spider Silk in *E. coli*

Sarah Allred

Spider silk has the potential to be a useful biomaterial due to its high tensile strength and elasticity. It is also biocompatible and biodegradable, making it useful for wound dressings and sutures, tissue and bone scaffolds, vessels for drug delivery, and ligament and tendon replacements. In some studies where spider silk has been used to grow cells, the silk has promoted more cell growth than the control. However, it is difficult to obtain the high volume of silk needed for these undertakings on a large scale. Spiders are territorial and cannibalistic, so they cannot be easily farmed. Therefore, spider silk proteins are frequently produced in other organisms. *E. coli* is often used for spider silk production due to the relative ease of gene manipulation and the cost effectiveness of large-scale fermentation. However, due to the large protein size of the spider silk and the repeating amino acid motifs, there are some challenges with production in *E. coli*.

Metabolic modeling is a way to model the metabolism of an organism and can help overcome some of the difficulties of spider silk production in *E. coli* by predicting metabolic engineering strategies. In this study, a metabolic modeling tool known as dynamic FBA predicted that ammonium is depleted during cell growth. Laboratory results confirmed that by adding additional ammonium to the medium, the *E. coli* cells experienced more cell growth and were able to produce more spider silk protein.

## CONTENTS

	Page
ABSTRACT .....	iii
PUBLIC ABSTRACT .....	iv
ACKNOWLEDGMENTS .....	vii
LIST OF TABLES .....	viii
LIST OF FIGURES .....	ix
INTRODUCTION .....	1
LITERATURE REVIEW .....	3
Spider silk .....	3
Mechanical properties .....	3
Chemical structure .....	4
Applications .....	6
Production process .....	7
Problems associated with the production process .....	8
Proposed solutions .....	9
Metabolic reconstructions .....	10
Flux balance analysis .....	14
Dynamic FBA .....	17
MATERIALS AND METHODS .....	19
Metabolic modeling .....	19
Medium simulation .....	19
FIYS3 reaction .....	23
FBA and dynamic FBA .....	24
Laboratory experiments .....	25
Cells and cell proliferation .....	25
Ammonium analysis .....	26
ELISA .....	28
Western blot .....	28

RESULTS.....	30
Metabolic modeling.....	30
Laboratory experiments.....	34
DISCUSSION.....	37
Interpretation of results.....	37
Recommended research.....	38
REFERENCES.....	39
APPENDICES.....	43
Appendix A: Matlab codes.....	44
Dynamic FBA.....	44
Optimize Cb model (FBA).....	46
Dynamic growth simulation.....	52
Appendix B: Recipes and protocols.....	54
Yeast-Hy ES II medium.....	54
K12 medium and K12 trace metal solution.....	54
Glucose feeding solution.....	55
Q-Plex array: anti poly-histidine kit.....	56
Western blot.....	68
Appendix C: Sequence information.....	74
FIYS3 sequence.....	74
Plasmid Modification.....	78
Plasmid Sequence.....	82

## ACKNOWLEDGMENTS

I would like to thank my major professor, Professor H. Scott Hinton, for his support and assistance throughout my project, as well as my committee members, Dr. Ron Sims, Dr. Randy Lewis, and Dr. Charles Miller. I would also like to thank Dr. Dong Chen and Justin Jones for the use of lab equipment and indispensable help. I am also grateful for the assistance given by Matthew Sims, Sreevidhra Krishnaji, Ashik Satish, Michael Hinman, and Joan McLean. I also thank Tessa Guy for performing lab analyses for me. I wish to thank USTAR for funding my project.

My special thanks are extended to my previous professors and advisors in the Biological Engineering department, especially to Dr. Timothy Taylor and Anne Martin who have been very helpful to me during my time at Utah State University. Finally, I would like to thank my husband and my family for their continual support and encouragement.

Sarah Allred

## LIST OF TABLES

Table	Page
1 Mechanical properties of silks and common materials.....	4
2 Amino acid composition of silks from <i>A. diadematus</i> .....	5
3 K12 medium and K12 trace metal solution.....	20
4 Initial glucose/acetate study .....	20
5 Concentration and lower bounds of metabolites in media.....	22
6 ReliOn reading standard curve .....	26
7 ELISA results .....	36
8 Statistical analysis of final ODs for NaOH and NH <sub>4</sub> OH groups, P value = 0.0032 .....	37
9 Statistical analysis of protein concentrations for NaOH and NH <sub>4</sub> OH groups, P value = 0.0096 .....	39

## LIST OF FIGURES

Figure	Page
1 Schematic diagram of the flagelliform gene structure .....	6
2 Amino acid sequence in flagelliform spider silk .....	6
3 Portion of reaction database for iECD_1391 model.....	11
4 Portion of metabolite database for iECD_1391 model.....	12
5 Stoichiometric representation of metabolic networks .....	13
6 The biomass function for the iJO1366 E. coli metabolic reconstruction .....	15
7 Typical amino acid profile (in mg/g) for HY-YEST 444.....	21
8 Previous fermentation experiment with E. coli B121(DE3) and pET19K-FIYS3 .....	24
9 Standard Curve for ReliOn Readings.....	26
10 Finding the background reading for the NH <sub>4</sub> electrode.....	28
11 Dynamic FBA output for all metabolites in K12 medium .....	31
12 Dynamic FBA output for all metabolites in K12 medium with unlimited ammonium.....	32
13 Dynamic FBA output for all metabolites in K12 medium with unlimited ammonium, zoomed in to view iron(III) depletion .....	33
14 Dynamic FBA output for all metabolites in K12 medium, with unlimited ammonium and unlimited iron(III) .....	34
15 The OD at 600 nm over time for E. coli cells grown using 20% NH <sub>4</sub> OH or 10% NaOH as the pH control.....	35
16 The ammonium concentration over time for E. coli cells grown using 20% NH <sub>4</sub> OH or 10% NaOH as the pH control.....	36
17 A western blot from the last time point of each fermentation experiment.....	37
18 Vector map of pET-19b.....	79
19 Vector map of pET-26b.....	80
20 Vector map of pET-19k.....	81
21 Vector map of pET19-SX.....	82

## INTRODUCTION

Spider silk has extraordinary mechanical properties that make it ideal as a biomaterial for many applications. Because of the unique combination of high tensile strength and great elasticity, spider silk is stronger than steel and can be several times tougher than Kevlar (Griffiths & Salanitri, 1980; Agnarsson *et al*, 2010). It is also biocompatible, biodegradable, and has good thermal stability to almost 200°C (Wong Po Foo & Kaplan, 2002; Chung *et al*, 2012). Studies have found recombinant spider silk to be effective as wound dressings, tissue scaffolds, and as scaffolds for bone regeneration (Gomes *et al*, 2011; Widhe *et al*, 2012). Many other applications for spider silk are being considered as well, including parachute cords, composite materials in aircrafts, protective clothing against high-velocity projectiles, vessels for drug delivery, and coatings for biomedical implants (Xia *et al*, 2010; Chung *et al*, 2012).

Unfortunately, natural spider silk cannot be easily obtained by farming spiders, both because of the relatively low levels of silk production and because spiders are highly territorial and aggressive (Xia *et al*, 2010). Instead, recombinant spider silk proteins are created in host systems, and the resulting proteins are then spun into artificial fibers. While several organisms have been investigated for host systems, including yeast (Fahnestock & Bedzyk, 1997), transgenic silkworms (Wen *et al*, 2010), insect and mammalian cell lines (Lazaris *et al*, 2002; Service, 2002; Zhang *et al*, 2008), and transgenic plants and animals (Scheller *et al*, 2001; Williams, 2003; Menassa *et al*, 2004; Xu *et al*, 2007), one of the most widely employed systems for recombinant spider silk protein production is the bacteria *Escherichia coli* (*E. coli*) (Chung *et al*, 2012). This is in part because the production of heterologous proteins has already been developed for large-scale fermentation in *E. coli*, making the process less prone to error and more cost effective. However, due to the large size of the spider silk proteins, it is difficult to produce enough spider silk to be useful.

One way to increase the production of spider silk in *E. coli* is to optimize the cell media. However, if modifications to cell media are done without consideration of the consequences to the organism, unwanted changes in cellular metabolism can occur. Metabolic modeling can predict how factors like media supplements affect the whole metabolic network. Metabolic modeling has been shown to be successful in predicting growth rates using a constraint-based reconstruction and analysis (COBRA) approach (Price *et al*, 2003). It can also accurately predict what metabolic changes need to be made to optimize the production of bioproducts in organisms (Lee *et al*, 2005; Feist & Palsson, 2008; Kim *et al*, 2008a, 2008b; Park *et al*, 2008). Metabolic modeling can assist in overcoming the challenges of producing recombinant spider silk proteins in *E. coli*, thus increasing the yield and quality of the bioproduct.

## LITERATURE REVIEW

### **Spider silk**

Spider silk has been a product of interest from antiquity because of its unique mechanical properties. These properties have evolved so spiders can build webs to successfully stop rapidly flying insects almost instantly, entangling and trapping their prey (Lewis, 2006). The webs of orb-weaving spiders are designed to absorb the energy of incoming insects without breaking and without bouncing the insect away. The efficiency of spider silk allows orb-weaving spiders to use the minimum amount of silk necessary in their webs to catch their prey (Lewis, 2006).

Spiders are able to produce as many as six different kinds of silk that have different tensile strengths and elasticity (Lewis, 2006). The mechanical properties of these silks are tailored to their different functions, such as creating different parts of the web, capturing prey with adhesive properties, creating the inner egg sac, aiding in reproduction, and acting as a dragline for the spider (Wong Po Foo & Kaplan, 2002). Flagelliform silk, formed in the flagelliform gland, forms the capture spiral of the web. It is one of the most well studied spider silks because of its exceptional mechanical properties (Hayashi & Lewis, 1998, 2001; Higgins *et al*, 2007).

#### *Mechanical properties*

Spider silk is an impressive biomaterial because it combines high tensile strength with great elasticity. The tensile strength of flagelliform silk is not as high as that of dragline silk (one of the strongest known biological materials), but the elongation is approximately 270% (Gosline *et al*, 1999). The combination of strength and high elasticity give flagelliform silk a greater toughness than Kevlar, rubber, silkworm silk, tendon, and bone, as shown in Table 1, which summarizes the mechanical properties of different types of spider silk compared with other materials.

**Table 1. Mechanical properties of silks and common materials<sup>a</sup>**

Material	Strength (MPa)	Strain (%)	Toughness (KJ/kg)
Dragline silk	4000	35	400
Minor Ampullate silk	1000	5	30
Flagelliform	1000	>200	400
Tubuliform silk	1000	20	100
<i>Bombyx mori</i> silk	600	20	60
Kevlar 49	3600	5	30
Rubber	50	850	80
Tendon	150	5	5
Bone	160	3	3

<sup>a</sup>Data from Gosline *et al*, 1999; Lewis, 2006; Altman *et al*, 2003

Spider silks are also insoluble in most solvents, including water, and dilute acids and alkaline materials. They are resistant to digestion from most proteolytic enzymes, which relates to the environmental stability of these proteins in fiber form (Wong Po Foo & Kaplan, 2002). Spider silks also have good thermal stability to almost 200°C and are biocompatible and biodegradable (Wong Po Foo & Kaplan, 2002; Chung *et al*, 2012). These remarkable properties have sparked interest in spider silk as a biomaterial and for medical applications.

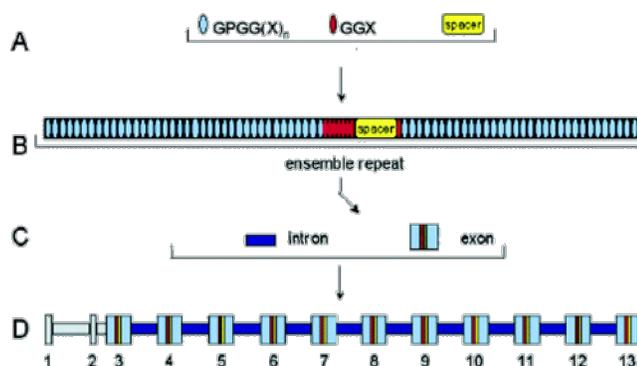
#### *Chemical structure*

The reasons behind the unique mechanical properties of spider silk have been elucidated in the past several years by studying the chemistry and structure of spider silk. In particular, the proteins that comprise the silk and their sequences provide information that directly relates to these properties (Lewis, 2006). Spider silks are made of proteins. Except for the sticky material deposited on some parts of the web, less than 0.1% of any other compound, including sugars, minerals, and lipids, is covalently linked to the silk proteins (Lewis, 2006). The major components of flagelliform silk are glycine, proline, alanine, and valine. For more details about the amino acid composition of spider silk, see Table 2.

**Table 2. Amino acid composition (%) of silks from *A. diadematus* (Andersen, 1970)**

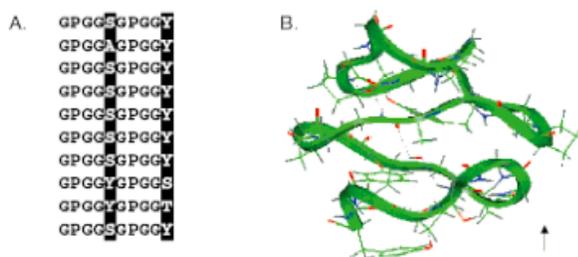
amino acid	major ampullate	minor ampullate	flagelliform	aciniform	tubuliform
Asp	1.04	1.91	2.68	8.04	6.26
Thr	0.91	1.35	2.48	8.66	3.44
Ser	7.41	5.08	3.08	15.03	27.61
Glu	11.49	1.59	2.89	7.22	8.22
Pro	15.77	trace amounts	20.54	2.99	0.59
Gly	37.24	42.77	44.16	13.93	8.63
Ala	17.60	36.75	8.29	11.30	24.44
Val	1.15	1.73	6.68	7.37	5.97
Ile	0.63	0.67	1.01	4.27	1.69
Leu	1.27	0.96	1.40	10.10	5.73
Tyr	3.92	4.71	2.56	1.99	0.95
Phe	0.45	0.41	1.08	2.79	3.22
Lys	0.54	0.39	1.35	1.90	1.76
His	trace amounts	trace amounts	0.68	0.31	trace amounts
Arg	0.57	1.69	1.13	4.09	1.49

Flagelliform silk is composed of repeating sequences of three different amino acid motifs, flanked on either side by a non-repetitive amino-terminal and a carboxy-terminal region, as shown in Fig 1. The three repeating units are GPGGX, GGX, and the highly conserved 34 amino acid spacer region TITEDLDITIDGADGPITISEELTISGA (Lewis, 2006). Most of the flagelliform protein sequence is made up of the pentapeptide GPGGX, where X is one of a small set of amino acids, usually alanine, valine, serine, or tyrosine. This motif is repeated 40 to 65 times and most likely forms type II  $\beta$ -turns (Hayashi & Lewis, 2001; Jenkins *et al*, 2010). The shorter subrepeat GGX is then repeated about 8 times, followed by a single occurrence of the spacer sequence. The flagelliform gene also contains a pattern of repeating introns that are highly conserved across different species. Tandem arrays of the previously mentioned  $\beta$ -turns most likely create  $\beta$ -spirals (see Fig 2), which enables flagelliform silk to stretch and recoil (Hayashi & Lewis, 2001), while the spacer motifs contribute to the strength of the fiber (Adrianos *et al*, 2013).



**Figure 1. Schematic diagram of the flagelliform gene structure. The organization is as follows:**

- A Sequence of individual protein components.
- B Protein repeat units.
- C Intron and exon units of the gene.
- D Complete gene structure, which is about 30 kb (Lewis, 2006).



**Figure 2. Amino acid sequence in flagelliform spider silk, made up of tandem repeats of the GPGGX unit, where X is usually alanine, valine, serine, or tyrosine.**

- A Here is shown a portion of the translated flagelliform cDNA (Genbank accession AF027973).
- B This molecular model of a GPGGSGPGGY peptide shows how the successive  $\beta$ -turns formed from the GPGGX motif likely form  $\beta$ -spirals, creating elasticity in the flagelliform fiber (putative fiber axis shown by the arrow) (Hayashi & Lewis, 2001).

### *Applications*

Because of the biocompatibility and biodegradability of spider silk, there are many possible applications in the biomedical field (Rising *et al*, 2011). Artificial nerve constructs partially constructed from dragline spider silk have been able to promote the regeneration of peripheral nerves with high functionality in rats (Allmeling *et al*, 2008). Studies have found recombinant spider silk to be effective as wound dressings, tissue scaffolds, and as scaffolds for bone regeneration (Gomes *et al*, 2011; Widhe *et al*, 2012). In many of these studies, silk has promoted cell growth more than the control. In addition to increased cell growth, one study

showed that human bone marrow derived mesenchymal stem cells (hMSCs) grown on spider silk showed increased calcification, which measures mature bone-related outcomes from the hMSCs (Bini *et al*, 2006). Spider silk can also be modified to contain cell binding sites, such as the amino acid sequence RGD, which can further increase cell adherence and proliferation (Wohlrab *et al*, 2012). Other proposed biomedical applications are bandages to stop bleeding and promote wound healing, sutures for wounds, vessels for drug delivery, coatings for biomedical implants, and ligament and tendon replacements (Xia *et al*, 2010; Chung *et al*, 2012).

Many other applications for spider silk are being considered as well because of the astonishing and unique mechanical properties. Some of these proposed ideas include parachute cords, composite materials in aircrafts, protective clothing against high-velocity projectiles, and construction materials (Xia *et al*, 2010; Chung *et al*, 2012).

#### *Production process*

There has been an increased effort to produce spider silk because of the various industrial applications. Unfortunately, natural spider silk cannot be easily obtained by farming spiders, both because of the relatively low levels of silk production and because the spiders are highly territorial and aggressive (Xia *et al*, 2010). In recent years, researchers have instead used genetic and metabolic engineering to construct, clone, and express native or synthetic genes encoding recombinant spider dragline silk proteins (Wong Po Foo & Kaplan, 2002). These proteins are then spun into artificial fibers.

Several host systems have been investigated as platforms for producing recombinant spider silk proteins. The production of spider silk in unicellular organisms, particularly in bacteria and yeasts, has been studied due to ease and cost effectiveness (Chung *et al*, 2012). Mammalian and insect cell lines have also been studied as potential platforms (Lazaris *et al*, 2002; Huemmerich *et al*, 2004; Zhang *et al*, 2008). Transgenic silkworms have been used, as well as transgenic plants, such as potato and tobacco plants, and transgenic animals, such as mice and

goats (Scheller *et al*, 2001; Nexia Biotechnologies, unpublished, 2002; Xu *et al*, 2007). Currently, spider silk production in bacteria and yeast seems most viable because the production of heterologous proteins has already been developed for large-scale fermentation in these two host systems, making the process less prone to error and more cost effective (Wong Po Foo & Kaplan, 2002). One of the most widely employed systems for recombinant spider silk protein production is the bacteria *E. coli* (Chung *et al*, 2012).

It has been found that the molecular weight of the spider dragline silk proteins is proportional to the quality of the mechanical properties of the fiber (Xia *et al*, 2010). Most synthetic silk proteins created are much smaller than natural spider silk proteins, which are between 250 and over 360 kDa (Chung *et al*, 2012). However, it has been observed that as the size of the recombinant proteins nears the size of the natural silk proteins, the mechanical properties of the recombinant spider silk proteins are improved (Xia *et al*, 2010). It is reasonable to assume that the same holds true for flagelliform silk, which is approximately 360 kDa in size (Hayashi & Lewis, 1998, 2001). The size of the silk protein affects more than just mechanical properties of the final product; one study found that larger sizes of the repetitive core domain of flagelliform silk notably increased the solubility in the presence of kosmotropic salt or slightly acidic pH, which naturally triggers silk assembly (Heim *et al*, 2010).

#### *Problems associated with the production process*

Since the gene sequences of spider flagelliform silk proteins are highly repetitive and are composed of a relatively small set of amino acids, they are difficult to express in *E. coli*. While translating the mRNA, if there are not enough of the needed amino acids or tRNAs, the ribosome will stall, and this can result in a prematurely terminated polypeptide chain. Since spider silk proteins of a larger molecular weight display improved mechanical properties, this is not ideal.

The amino acid that poses the largest challenge is glycine. *E. coli* cells do not normally produce copious amounts of glycine; however, glycine is the most commonly occurring amino

acid in spider silk proteins. It has also been suggested that because of the highly repetitive amino acid sequence of the spider silk gene, DNA deletion in the gene as well as transcriptional and translational errors can occur (Chung *et al*, 2012). Therefore, the spider silk genes are difficult to express in *E. coli*.

### *Proposed solutions*

Metabolic engineering techniques allow several solutions to be proposed to aid with these challenges. There are several ways to potentially increase the amount glycine available to the *E. coli* cells, some of which have been suggested and tested recently by Xia *et al*. (2010). One way is to increase the number of tRNAs for glycine by introducing extra tRNA-producing genes into the cell. Measures could also be taken to overexpress the serine hydroxymethyltransferase enzyme so more serine can be converted into glycine. These methods were shown to be effective in producing dragline spider silk proteins with a higher molecular weight, although results of this study have yet to be duplicated (Xia *et al*, 2010). Methods that were not successful included the addition of glycine to the culture medium, the inactivation of the glycine cleavage system, and the overexpression of glycyl-tRNA synthase, which aminoacylates or charges the amino acid (Xia *et al*, 2010).

Although it is critical to ensure enough glycine is available, increasing the amount of available proline, alanine, and valine could increase spider silk production as well. The metabolic engineering approach for increasing these other amino acids would be similar to that of glycine. The intracellular production of the amino acids could be increased by either overexpressing the metabolic reactions preceding the amino acid, or by systematically knocking out nonessential genes to redirect the flow of metabolites through the target amino acid producing reactions. The tRNA pools for the amino acids could then be increased by introducing the tRNA genes into *E. coli*, which will allow more amino acids to be made into proteins. During these steps however, care should be taken to ensure that the cell does not contain too many non-aminoacylated tRNAs

at a time because non-aminoacylated tRNAs activate the stringent response, which inhibits rRNA and tRNA synthesis. Activation of the stringent response can be avoided by not increasing the tRNA pool too rapidly and by making sure there are enough corresponding amino acids to charge the tRNAs.

Additional supplements and nutrients added to the cell media could also increase the amount of intracellular amino acids that make up spider silk. Metabolic modeling can be used to predict which method is best to produce optimal amounts of spider silk.

### **Metabolic reconstructions**

In the mid-1800s, Gregor Mendel carried out a series of experiments that established the existence of discrete inherited elements, now called genes, that determine organism function and form (Henig, 2000). The relationship between the genetic makeup of a cell (genotype) and the organism's observable characteristics or traits (phenotype) is now a fundamental concept in biology. In the decades that followed, it was observed that many traits found in organisms are monogenic, meaning they are caused by one gene. However, the vast majority of traits are polygenic, meaning they are caused by many genes. The complex nature of polygenic traits can make understanding and predicting different phenotypes difficult.

Whole genome sequences started to become available in 1995 (Fleischmann *et al*, 1995). In principle, this made it possible to identify and characterize all of the genetic elements of an organism, although in practice, the function of some genes is still difficult to determine. Using these new genetic tools along with biochemical and physiological information, the first genome-scale metabolic reconstruction of *E. coli* was created in 2000 (Edwards & Palsson, 2000). Since then, metabolic reconstructions have been built for over 100 organisms, including *Bacillus subtilis*, *Helicobacter pylori*, *Pseudomonas aeruginosa*, *Saccharomyces cerevisiae*, and *Homo sapiens* (Feist *et al*, 2009; Thiele & Palsson, 2010). As new information is gathered, these reconstructions are updated to reflect the current metabolic knowledge available on the organism.

Metabolic reconstructions use stoichiometric information about biochemical transformations taking place in a target organism and represent the totality of the metabolism encoded in the genome. There are two parts of a metabolic reconstruction: a database of reactions, and a database of metabolites. The reaction database contains a list of all known biochemical reactions that take place in the organism, including exchange reactions that transport metabolites in and out of the cell and a biomass reaction, representing cell growth. The stoichiometrically balanced chemical equations of each reaction are listed, along with each reaction's subsystem, reversibility, associated genes and proteins, and upper and lower bounds of allowable flux (in  $\text{mmol gDW}^{-1}\text{h}^{-1}$ ) through the reaction (see Fig 3). The metabolite database contains a list of all metabolites present in each reaction, along with the metabolite charged formula and the metabolite charge (see Fig 4). Metabolites present in the extracellular medium, periplasm, and cytoplasm are designated by [e], [p], and [c], respectively.

These reactions of a metabolic reconstruction can be visualized as a network map, as shown in Fig 5A. This network map resembles a metabolic map, but has a few key differences: every reaction shown is associated with a gene and a genomic location on the genome of the target organism, and these network maps are specific to a target genome/organism (Palsson, 2006, 33–37). Network maps can be represented mathematically using the stoichiometric coefficients in each reaction. These can create a connectivity matrix (known as a stoichiometric matrix in the case of a metabolic network) that defines the nodes and links in the network.

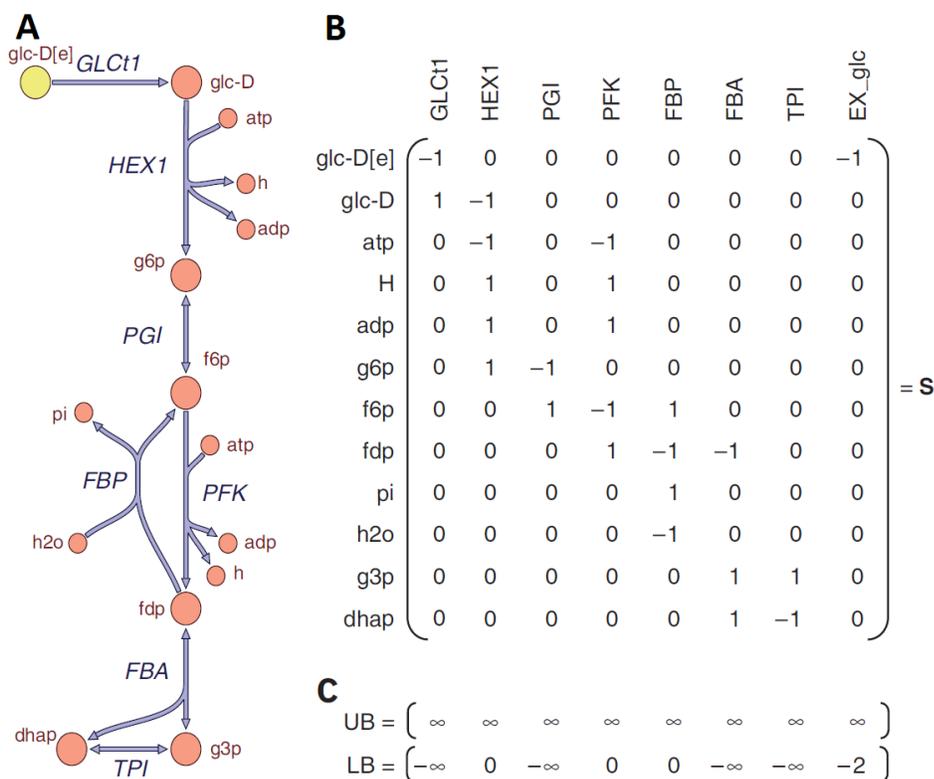
A stoichiometric matrix (S matrix) is a systematic way of representing all of the stoichiometric coefficients in the network. Each column corresponds to a particular reaction, each row corresponds to a particular metabolite, and each numerical element is the corresponding stoichiometric coefficient, as shown in Fig 5B (Becker *et al*, 2007). This mathematical representation of a metabolic network sets a mechanistic basis for the genotype-phenotype relationship and allows systematic tracing of the consequences of deleting or overexpressing a gene.

Rxn name	Rxn description	Formula	Gene-reaction association	Genes	Subsystem	Reversible	LB	UB
ACONTa	aconitase (half-reaction A, Citrate hydro-lyase)	cit[c] <=> acon-C[c] + h2o[c]	(ECD_00117 or ECD_01253)	ECD_00117 ECD_01253	Citric Acid Cycle	1	-1000	1000
ACONTb	aconitase (half-reaction B, Isocitrate hydro-lyase)	acon-C[c] + h2o[c] <=> icit[c]	(ECD_00117 or ECD_01253)	ECD_00117 ECD_01253	Citric Acid Cycle	1	-1000	1000
AKGDH	2-Oxoglutarate dehydrogenase	coa[c] + nad[c] + akg[c] -> co2[c] + nadh[c] + succoa[c]	((ECD_00115 and ECD_00685 and ECD_00686)	ECD_00115 ECD_00685 ECD_00686	Citric Acid Cycle	0	0	1000
CITL	Citrate lyase	cit[c] -> ac[c] + oaa[c]	((ECD_00583 and ECD_00584 and ECD_00585) and ECD_00582)	ECD_00582 ECD_00583 ECD_00584 ECD_00585	Citric Acid Cycle	0	0	1000
CS	citrate synthase	accoa[c] + oaa[c] + h2o[c] -> coa[c] + h[c] + cit[c]	ECD_00680	ECD_00680	Citric Acid Cycle	0	0	1000
FRD2	fumarate reductase	fum[c] + mql8[c] -> succ[c] + mqn8[c]	(ECD_04023 and ECD_04024 and ECD_04025 and ECD_04026)	ECD_04023 ECD_04024 ECD_04025 ECD_04026	Citric Acid Cycle	0	0	1000
FRD3	fumarate reductase	2dmmq8[c] + fum[c] -> succ[c] + 2dmmq8[c]	(ECD_04023 and ECD_04024 and ECD_04025 and ECD_04026)	ECD_04023 ECD_04024 ECD_04025 ECD_04026	Citric Acid Cycle	0	0	1000
FUM	fumarase	fum[c] + h2o[c] <=> mal-L[c]	(ECD_01581 or ECD_01581 or ECD_01580)	ECD_01580 ECD_01581	Citric Acid Cycle	1	-1000	1000
ICDHyr	isocitrate dehydrogenase (NADP)	nadp[c] + icit[c] <=> co2[c] + nadph[c] + akg[c]	ECD_01134	ECD_01134	Citric Acid Cycle	1	-1000	1000
MDH	malate dehydrogenase	mal-L[c] + nad[c] <=> h[c] + nadh[c] + oaa[c]	ECD_03096	ECD_03096	Citric Acid Cycle	1	-1000	1000
MDH2	Malate dehydrogenase (ubiquinone 8 as acceptor)	q8[c] + mal-L[c] -> oaa[c] + q8h2[c]	ECD_02137	ECD_02137	Citric Acid Cycle	0	0	1000
MDH3	Malate dehydrogenase (menaquinone 8 as acceptor)	mal-L[c] + mqn8[c] -> mql8[c] + oaa[c]	ECD_02137	ECD_02137	Citric Acid Cycle	0	0	1000
MOX	malate oxidase	mal-L[c] + o2[c] <=> h2o2[c] + oaa[c]			Citric Acid Cycle	1	-1000	1000
SUCOAS	succinyl-CoA synthetase (ADP-forming)	atp[c] + coa[c] + succ[c] <=> adp[c] + pi[c] + succoa[c]	(ECD_00687 and ECD_00688)	ECD_00687 ECD_00688	Citric Acid Cycle	1	-1000	1000

**Figure 3.** Portion of reaction database for iECD\_1391 model (Monk *et al*, 2013) representing the organism *Escherichia coli* BL21(DE3) CP001509. Shown are the reactions of the citric acid cycle.

Metabolite name	Metabolite description	Metabolite charged formula	Metabolite charge
2dmmq8[c]	2-Demethylmenaquinone 8	C50H70O2	0
2dmmql8[c]	2-Demethylmenaquinol 8	C50H72O2	0
ac[c]	Acetate	C2H3O2	-1
accoa[c]	Acetyl-CoA	C23H34N7O17P3S	-4
acon-C[c]	cis-Aconitate	C6H3O6	-3
adp[c]	ADP	C10H12N5O10P2	-3
akg[c]	2-Oxoglutarate	C5H4O5	-2
atp[c]	ATP	C10H12N5O13P3	-4
cit[c]	Citrate	C6H5O7	-3
co2[c]	CO2	CO2	0
coa[c]	Coenzyme A	C21H32N7O16P3S	-4
fum[c]	Fumarate	C4H2O4	-2

**Figure 4.** Portion of metabolite database for iECD\_1391 model (Monk *et al*, 2013) representing the organism *Escherichia coli* BL21(DE3) CP001509. Shown are some of the metabolites in the citric acid cycle.



**Figure 5. Stoichiometric representation of metabolic networks.**

- A** The first few reactions of glycolysis.
- B** The stoichiometric matrix (S) corresponding to the reactions in a. Each column corresponds to a particular reaction, each row corresponds to a particular metabolite, and the numerical element is the corresponding stoichiometric coefficient.
- C** The upper (UB) and lower (LB) bounds for each reaction in  $\text{mmol gDW}^{-1} \text{h}^{-1}$ . The four reversible reactions (GLCt1, PGI, FBA, TPI) have lower bounds of  $-\infty$ . The three irreversible reactions (HEX1, PFK, FBP) have lower bounds of zero, because they cannot proceed in the reverse direction. The exchange reaction for glucose (EX\_glc) has a lower bound of  $-2 \text{ mmol gDW}^{-1} \text{h}^{-1}$ , indicating a potential uptake rate for glucose into the cell. The upper bound for all reactions is  $\infty$ , meaning the reactions are unconstrained in the forward direction (Becker *et al.*, 2007).

The intracellular environment is crowded and interconnected, placing severe constraints on achievable physiological states (Palsson, 2006, 20–21). For example, an average *E. coli* cell is only 1 cubic micron in volume, and in it are about 2,500,000 proteins simultaneously at work (Lu *et al.*, 2007; Phillips *et al.*, 2008, 26). These constraints can be modeled in metabolic reconstructions using upper and lower bounds, as previously discussed and as shown in Fig 5C. These values represent the range of allowable flux (flow of metabolites in  $\text{mmol gDW}^{-1} \text{h}^{-1}$ )

through a reaction; the upper bounds represent the maximum flux allowed in a reaction, and the lower bounds represent the minimum amount of flux that can flow through a reaction.

In addition to metabolic reactions, other reactions can be added to a metabolic reconstruction to represent phenomena such as metabolite exchange/transport and biomass formation. Exchange reactions allow the simulation of metabolites being added or removed from the intracellular environment; exchange reactions with a negative flux value indicate removal from the extracellular environment (substrate uptake), while positive flux values indicate the addition of metabolites to the extracellular environment (secretion). Changing the lower bounds of exchange reactions controls how much of a metabolite can enter the cell. For example, a cell in aerobic conditions on a glucose substrate could be modeled by changing both the lower bound of the oxygen exchange reaction and the lower bound of the glucose exchange reaction to  $-20 \text{ mmol gDW}^{-1} \text{ h}^{-1}$ , an uptake value close to that observed in aerobic cultures (Feist *et al*, 2010).

The growth rate of a cell can be simulated using a biomass reaction. A biomass reaction accounts for all known biomass constituents and their fractional contributions to the overall cellular biomass (Thiele & Palsson, 2010). This is obtained by gathering detailed information about the chemical composition of the cells, including proteins, RNA, DNA, lipids, lipopolysaccharides, peptidoglycan, glycogen, etc. Once the biomass composition is found, a stoichiometric reaction is created and scaled so that the flux through it is equal to the exponential growth rate ( $\mu$ ) of the organism (Orth *et al*, 2010). An example of a typical biomass reaction is shown in Fig 6.

### *Flux balance analysis*

Flux balance analysis (FBA) is an important tool for harnessing the knowledge encoded in these metabolic reconstructions. FBA is a widely used mathematical approach for analyzing the flow of metabolites through a metabolic network based on constraints imposed on the model (Orth *et al*, 2010). These constraints, in the form of stoichiometric coefficients and the upper and

$$\begin{aligned}
&0.000223 \text{ 10fthf}[c] + 0.000223 \text{ 2dmmql8}[c] + 2.5e-05 \text{ 2fe2s}[c] + 0.000248 \text{ 4fe4s}[c] + 0.000223 \text{ 5mthf}[c] + \\
&0.000279 \text{ accoa}[c] + 0.000223 \text{ adocbl}[c] + 0.49915 \text{ ala-L}[c] + 0.000223 \text{ amet}[c] + 0.28742 \text{ arg-L}[c] + \\
&0.23423 \text{ asn-L}[c] + 0.23423 \text{ asp-L}[c] + 54.12 \text{ atp}[c] + 0.000116 \text{ bmocogdp}[c] + 2e-06 \text{ btn}[c] + 0.004952 \\
&\text{ ca2}[c] + 0.000223 \text{ chor}[c] + 0.004952 \text{ cl}[c] + 0.000168 \text{ coa}[c] + 2.4e-05 \text{ cobalt2}[c] + 0.1298 \text{ ctp}[c] + \\
&0.000674 \text{ cu2}[c] + 0.088988 \text{ cys-L}[c] + 0.024805 \text{ datp}[c] + 0.025612 \text{ dctp}[c] + 0.025612 \text{ dgtp}[c] + \\
&0.024805 \text{ dttp}[c] + 0.000223 \text{ enter}[c] + 0.000223 \text{ fad}[c] + 0.006388 \text{ fe2}[c] + 0.007428 \text{ fe3}[c] + 0.25571 \\
&\text{ gln-L}[c] + 0.25571 \text{ glu-L}[c] + 0.5953 \text{ gly}[c] + 0.15419 \text{ glycogen}[c] + 0.000223 \text{ gthrd}[c] + 0.20912 \text{ gtp}[c] + \\
&48.7529 \text{ h2o}[c] + 0.000223 \text{ hemeO}[c] + 0.092056 \text{ his-L}[c] + 0.28231 \text{ ile-L}[c] + 0.18569 \text{ k}[c] + 0.43778 \text{ leu-} \\
&\text{ L}[c] + 3e-06 \text{ lipopb}[c] + 0.33345 \text{ lys-L}[c] + 3.1e-05 \text{ malcoa}[c] + 0.14934 \text{ met-L}[c] + 0.008253 \text{ mg2}[c] + \\
&0.000223 \text{ mlthf}[c] + 0.000658 \text{ mn2}[c] + 7e-06 \text{ mobd}[c] + 7e-06 \text{ mococdp}[c] + 7e-06 \text{ mocogdp}[c] + \\
&0.000223 \text{ mql8}[c] + 0.001787 \text{ nad}[c] + 4.5e-05 \text{ nadh}[c] + 0.000112 \text{ nadp}[c] + 0.000335 \text{ nadph}[c] + \\
&0.012379 \text{ nh4}[c] + 0.000307 \text{ ni2}[c] + 0.012366 \text{ pe160}[c] + 0.009618 \text{ pe161}[c] + 0.004957 \text{ pe181}[c] + \\
&0.005707 \text{ pg160}[c] + 0.004439 \text{ pg161}[c] + 0.002288 \text{ pg181}[c] + 0.18002 \text{ phe-L}[c] + 0.000223 \text{ pheme}[c] + \\
&0.2148 \text{ pro-L}[c] + 0.03327 \text{ ptrc}[c] + 0.000223 \text{ pydx5p}[c] + 0.000223 \text{ q8h2}[c] + 0.000223 \text{ ribflv}[c] + \\
&0.20968 \text{ ser-L}[c] + 0.000223 \text{ sheme}[c] + 0.004126 \text{ so4}[c] + 0.006744 \text{ spmd}[c] + 9.8e-05 \text{ succoa}[c] + \\
&0.000223 \text{ thf}[c] + 0.000223 \text{ thmpp}[c] + 0.24651 \text{ thr-L}[c] + 0.055234 \text{ trp-L}[c] + 0.13399 \text{ tyr-L}[c] + 5.5e-05 \\
&\text{ udcpdp}[c] + 0.1401 \text{ utp}[c] + 0.41118 \text{ val-L}[c] + 0.000324 \text{ zn2}[c] + 0.008151 \text{ colipa}[e] + 0.002944 \\
&\text{ clpn160}[p] + 0.00229 \text{ clpn161}[p] + 0.00118 \text{ clpn181}[p] + 0.001345 \text{ murein3p3p}[p] + 0.000605 \\
&\text{ murein3px4p}[p] + 0.005381 \text{ murein4p4p}[p] + 0.005448 \text{ murein4px4p}[p] + 0.000673 \text{ murein4px4px4p}[p] \\
&+ 0.031798 \text{ pe160}[p] + 0.024732 \text{ pe161}[p] + 0.012747 \text{ pe181}[p] + 0.004892 \text{ pg160}[p] + 0.003805 \\
&\text{ pg161}[p] + 0.001961 \text{ pg181}[p] \rightarrow 53.95 \text{ adp}[c] + 53.95 \text{ h}[c] + 53.9459 \text{ pi}[c] + 0.74983 \text{ ppi}[c]
\end{aligned}$$

**Figure 6. The biomass function for the iJO1366 *E. coli* metabolic reconstruction, the most current *E. coli* reconstruction and the most complete metabolic reconstruction to date (Orth & Palsson, 2012).**

lower bounds on the flux through each reaction, define the space of allowable flux distribution of a system.

Once constraints are established, a biological objective is chosen from the reactions in a metabolic reconstruction. This objective can be either maximized or minimized. Often, the biomass reaction is chosen as the objective function and is maximized to simulate maximum cell growth. The mathematical representations of the metabolic reactions in the stoichiometric matrix and the biological objective function define a system of linear equations that are solved using linear programming. Since there are almost always more reactions (columns in the S matrix) than metabolites (rows in the S matrix), there is no unique solution for the system of equations. FBA uses constraints in the form of upper and lower bounds to narrow the solution space and optimize the objective function. The output for FBA is a list of reactions and their corresponding fluxes that optimize the objective function.

FBA does not require kinetic parameters and can compute results quickly even for relatively large networks (Orth *et al*, 2010). This feature makes it well suited for studying variations in growth conditions such as the use of different substrates, different oxygenation levels, and different genetic manipulations. However, FBA has limitations. Because it is a constraint-based approach and does not use kinetic parameters, FBA cannot predict metabolite concentrations. FBA also operates only under the assumption of steady state conditions, meaning the flow of metabolites and model conditions do not change over time, and, except in some modified forms, it does not account for the regulation of gene expression or for regulatory effects such as activation of enzymes by protein kinases (Orth *et al*, 2010).

There are many diverse applications for FBA. FBA used with genome-scale metabolic reconstructions can be used for physiological studies, gap-filling efforts, and genome-scale synthetic biology (Orth *et al*, 2010). By changing the lower bounds for exchange reactions for different metabolites, cell growth on different media and varying oxygenation levels can be modeled. Setting both the upper and lower bounds of a reaction to zero (thus eliminating the possibility for flux flowing through the reaction) can simulate a reaction knockout, and using genomic and genetic information contained in the metabolic reconstruction relating the different reactions to specific genes can simulate gene knockouts. Adding enzymatic reactions to the reconstruction can show the effects of gene additions, and constraints can be altered to model the overexpression of specific genes.

Metabolic modeling using FBA has been able to predict cell growth under different conditions and has been able to accurately predict what metabolic changes need to be made to optimize the production of bioproducts in organisms (Lee *et al*, 2005; Feist & Palsson, 2008; Kim *et al*, 2008a, 2008b; Park *et al*, 2008). However, all genome-scale metabolic reconstructions are incomplete and contain ‘knowledge gaps’ where reactions are missing (Orth *et al*, 2010). Therefore, modeling results need to be tested in the laboratory. When modeling results do not parallel *in vivo* results, the reconstruction is refined accordingly. Thus, building and using

metabolic reconstructions is an iterative process (Thiele & Palsson, 2010).

### *Dynamic FBA*

Dynamic FBA is a program that utilizes FBA to predict the concentration of extracellular metabolites and the growth rate over a series of small time steps. Initial concentration values of different extracellular metabolites are specified, as are the number of time steps, and the size of the time steps. Beginning with the first time step, the following iterative algorithm predicts concentrations for each substrate and the growth rate for all consecutive time steps (Varma & Palsson, 1994).

- (i) The substrate concentration ( $S_c$ ) (mmol/L) is determined from the substrate concentration predicted for the previous step ( $S_{co}$ ) or from the initial substrate concentration if it is the first time step:

$$S_c = S_{co}$$

- (ii) The substrate concentration is scaled to define the amount of substrate available per unit of biomass per unit of time (mmol gDW<sup>-1</sup>h<sup>-1</sup>):

$$\text{Substrate available} = \frac{S_c}{X \times \Delta t}$$

where  $X$  is the cell density.

- (iii) FBA is then used to calculate the substrate uptake ( $S_u$ ) and the growth rate ( $\mu$ ).
- (iv) Concentrations for the next time step are calculated from the standard differential equations:

$$\frac{dX}{dt} = \mu X \rightarrow X = X_0 \times e^{\mu \Delta t}$$

$$\frac{dS_c}{dt} = -S_u \times X \rightarrow S_c = S_{co} + \frac{S_u}{\mu} X_0 (1 - e^{\mu \Delta t})$$

The output of dynamic FBA is two graphs: one showing the flux through the objective reaction over time, and one showing the flux through the exchange reactions for the selected metabolites over time. For the dynamic FBA code, see Appendix A.

One downside to dynamic FBA is that at this point, there is no way to account for substrates that enter the medium via fed batch mode, so it can only show what becomes of the initial concentrations. Another limitation is that dynamic FBA was created to optimize the biomass reaction, so there is currently no way to maximize reactions for protein production, or to maximize both the growth rate and protein production at the same time. Also, the predicted growth rate can sometimes reach values higher than possible because the calculated growth rate is constantly in the exponential phase. These aspects make the tool more useful for qualitative rather than quantitative study.

However, dynamic FBA is a useful tool to understand how substrates are used by the organism over time. In one study, dynamic FBA successfully predicted the time profiles of cell density, glucose, and fermentation byproduct concentrations (Varma & Palsson, 1994). In cases where a recombinant protein is being produced, if the lower bound of flux through the protein production is set to a reasonable value, dynamic FBA can be useful for determining which metabolites are limiting.

## MATERIALS AND METHODS

### **Metabolic modeling**

The iECD\_1391 model (Monk *et al*, 2013) representing the organism *Escherichia coli* BL21(DE3) CP001509 was used in this study. Modeling was done using the COBRA Toolbox, (version 2.0.5), with the solver Gurobi (version 5), using Matlab r2013a (version 8.1.0.604).

### *Medium simulation*

The lower bound of the oxygen exchange reaction was set to  $-20 \text{ mmol gDW}^{-1}\text{h}^{-1}$ . This has been shown to be the maximum oxygen uptake rate for *E. coli* cells growing aerobically on a variety of carbon sources (Varma *et al*, 1993).

Growth on K12 medium was simulated by adjusting lower bounds (in  $\text{mmol gDW}^{-1}\text{h}^{-1}$ ) of the exchange reactions. Exchange reactions with a negative flux value indicate removal from the extracellular environment (substrate uptake), while positive flux values indicate the addition of metabolites to the extracellular environment (secretion). The lower bound of glucose was the first to be adjusted. The initial glucose concentration in K12 medium is 25 g/L, or 138.89 mmol/L (see Table 3 for recipe). Glucose concentrations higher than this can result in the production of the fermentation product acetate, which can be toxic to the cells. The specific uptake rate for glucose in the media was not available, so an initial study was done to determine what the lower bound of the glucose exchange reaction should be. The study showed that when the oxygen exchange reaction is set to  $-20 \text{ mmol gDW}^{-1}\text{h}^{-1}$ , the maximum value the of the lower bound of the glucose exchange reaction without secreting acetate is  $-11 \text{ mmol gDW}^{-1}\text{h}^{-1}$  (see Table 4).

For each chemical compound in the media, the mass fraction of each constituent was found and converted into g/L. For example, there are 4 g/L of dipotassium phosphate. This chemical is not in the model, but both potassium and phosphate alone are. There are 1.8 g/L of potassium and 2.18 g/L of phosphate. These values were then converted into mmol/L. The only

**Table 3. K12 medium and K12 trace metal solution**

	Chemical	Concentration
K12 Medium:	KH <sub>2</sub> PO <sub>4</sub>	2 g/L
	K <sub>2</sub> HPO <sub>4</sub> .3H <sub>2</sub> O	4 g/L
	(NH <sub>4</sub> ) <sub>2</sub> HPO <sub>4</sub>	5 g/L
	Yeast Extract	5 g/L
	Glucose	25 g/L
	MgSO <sub>4</sub> .7H <sub>2</sub> O	0.5 g/L
	Thiamine	2.5 mg/L
	K12 trace metal	5 ml/L
K12 trace metal solution:	NaCl	5 g/L
	ZnSO <sub>4</sub> .7H <sub>2</sub> O	1 g/L
	MnCl <sub>2</sub> .4H <sub>2</sub> O	4 g/L
	FeCl <sub>3</sub> .6H <sub>2</sub> O	4.75 g/L
	CuSO <sub>4</sub> .5H <sub>2</sub> O	0.4 g/L
	H <sub>3</sub> BO <sub>3</sub>	0.575 g/L
	NaMoO <sub>4</sub> .2H <sub>2</sub> O	0.5 g/L
	6N H <sub>2</sub> SO <sub>4</sub>	12.5 ml/L

**Table 4. Initial glucose/acetate study**

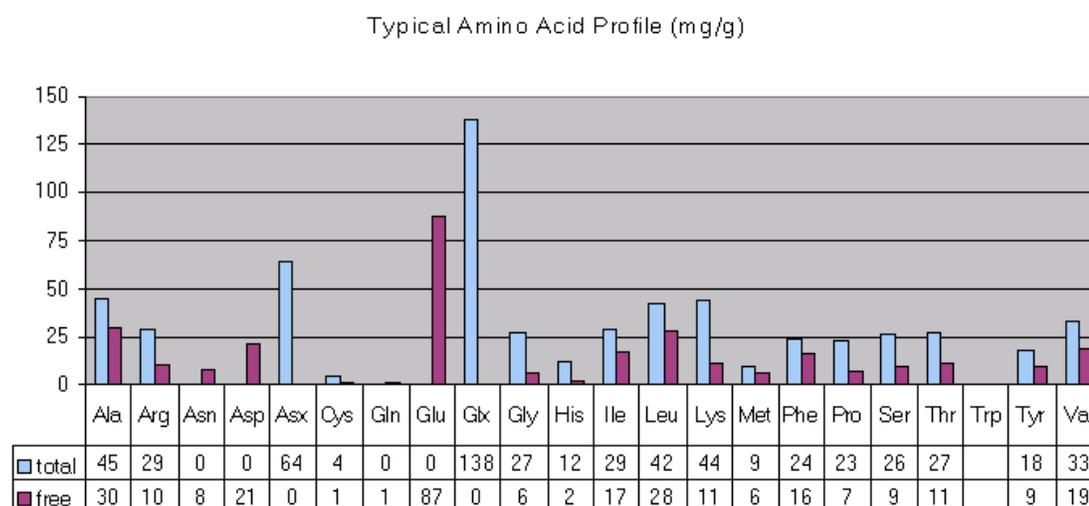
Flux in mmol gDW-1h-1		
Glucose Exchange	Acetate Exchange	Oxygen Exchange
-15	7.92215	-20
-14	5.82052	-20
-13	3.71888	-20
-12	1.61725	-20
-11	0	-19.6029
-10	0	-17.8707

chemical in the K12 medium that did not have an exchange reaction in the model was boric acid.

The metabolite borate (nor boron in any form) is not included in the iECD\_1391 model.

The yeast extract used in the K12 medium is HY-YEST 444 from Kerry (product code 5Z10313). Like most natural yeast extracts, the composition is undefined, but the product information sheet gives the typical amino acid profile (see Fig 7). Using the total amino acid concentration (in mg/g), the concentration of each amino acid in the medium in mmol/L was calculated. The non-abbreviation Asx means "asparagine or aspartic acid" and Glx means

"glutamic acid or glutamine." For these two cases, half of value was added to the total amino acid concentration for the two amino acids the abbreviation was associated with. The product information sheet for the HY-YEST 444 also listed that the NaCl concentration was 1.0% maximum, so an NaCl concentration of 1.0% was also incorporated into the model.



**Figure 7. Typical amino acid profile (in mg/g) for HY-YEST 444 (Kerry, Product Code 5Z10313).**

Specific uptake rates for each metabolite in the medium were not available, so the values for the lower bounds were obtained using the conversion factor  $138.89 \text{ mmol/L} = 11 \text{ mmol gDW}^{-1} \text{h}^{-1}$  ( $12.63 \text{ mmol/L} = 1 \text{ mmol gDW}^{-1} \text{h}^{-1}$ ), found from converting the concentration of glucose to the glucose uptake rate. Although calcium, cobalt, and nickel are not included in the media recipe, the model did not produce biomass unless a small amount of flux was allowed to go through those exchange reactions. Therefore, the lower bounds of calcium, cobalt, and nickel were set to  $-2.37 \times 10^{-03}$ ,  $-1.14 \times 10^{-05}$ , and  $-1.47 \times 10^{-04} \text{ mmol gDW}^{-1} \text{h}^{-1}$  respectively. Table 5 shows all metabolites in the media that are accounted for in the model, the concentration in g/L, the concentration in mmol/L, and the calculated uptake rate in  $\text{mmol gDW}^{-1} \text{h}^{-1}$ . Additional nutrients are gradually added by the glucose feed, but since the contribution of these nutrients is minimal

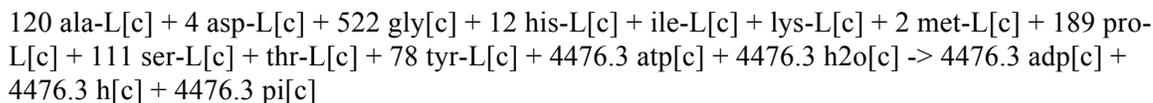
and not accurately known, only the initial concentrations of nutrients in the K12 medium were incorporated into the model.

**Table 5. Concentration and lower bounds of metabolites in media**

Metabolite	MW (g/mol)	g/L in media	mmol/L in media	Lower bound (mmol gDW <sup>-1</sup> h <sup>-1</sup> )
Glucose	180.16	25	138.7655417	11
Ammonium	18.03851	1.365829484	75.71742258	6.002150375
Phosphate	94.9714	6.655736264	70.08147994	5.555386948
Potassium	39.0983	1.945099309	49.74894838	3.943619038
Sulfate	96.07	0.203323529	2.11641021	0.167768684
Chloride	35.453	0.062050364	1.750214773	0.138740225
Copper	63.546	0.000509009	0.008010093	0.000634963
Iron (III)	55.845	0.00490684	0.087865335	0.00696512
Magnesium	24.305	0.049304203	2.028562155	0.160804933
Manganese	54.938044	0.005551956	0.101058487	0.008010947
Molybdate	95.95	0.001826052	0.019031286	0.001508618
Sodium	22.98976928	0.029775544	1.295164976	0.102668246
Thiamine	265.35	0.0025	0.009421519	0.000746848
Zinc	65.38	0.001136847	0.017388304	0.001378378
Alanine	89.09	0.225	2.525535975	0.200200247
Arginine	174.2	0.145	0.832376579	0.065982824
Asparagine	132.12	0.16	1.211020285	0.095998062
Aspartic acid	133.1	0.16	1.202103681	0.09529124
Cysteine	121.16	0.02	0.165070981	0.013085243
Glutamine	146.14	0.345	2.360749966	0.187137594
Glutamic Acid	147.13	0.345	2.344865085	0.185878393
Glycine	75.07	0.135	1.798321567	0.14255367
Histidine	155.15	0.06	0.386722527	0.030655649
Isoleucine	131.17	0.145	1.105435694	0.08762833
Leucine	131.17	0.21	1.600975833	0.126909995
Lysine	146.19	0.22	1.504890895	0.119293303
Methionine	149.21	0.045	0.301588365	0.02390703
Phenylalanine	165.19	0.12	0.726436225	0.05758489
Proline	115.13	0.115	0.998870842	0.079180891
Serine	105.09	0.13	1.237034922	0.098060253
Threonine	119.12	0.135	1.133310947	0.089838012
Tyrosine	181.19	0.09	0.496716154	0.039374888
Valine	117.15	0.165	1.408450704	0.111648451

### *FIYS3 reaction*

The following reaction simulating the production of FIYS3, a protein made of 3 repeating fragments of flagelliform silk, was added to the model:



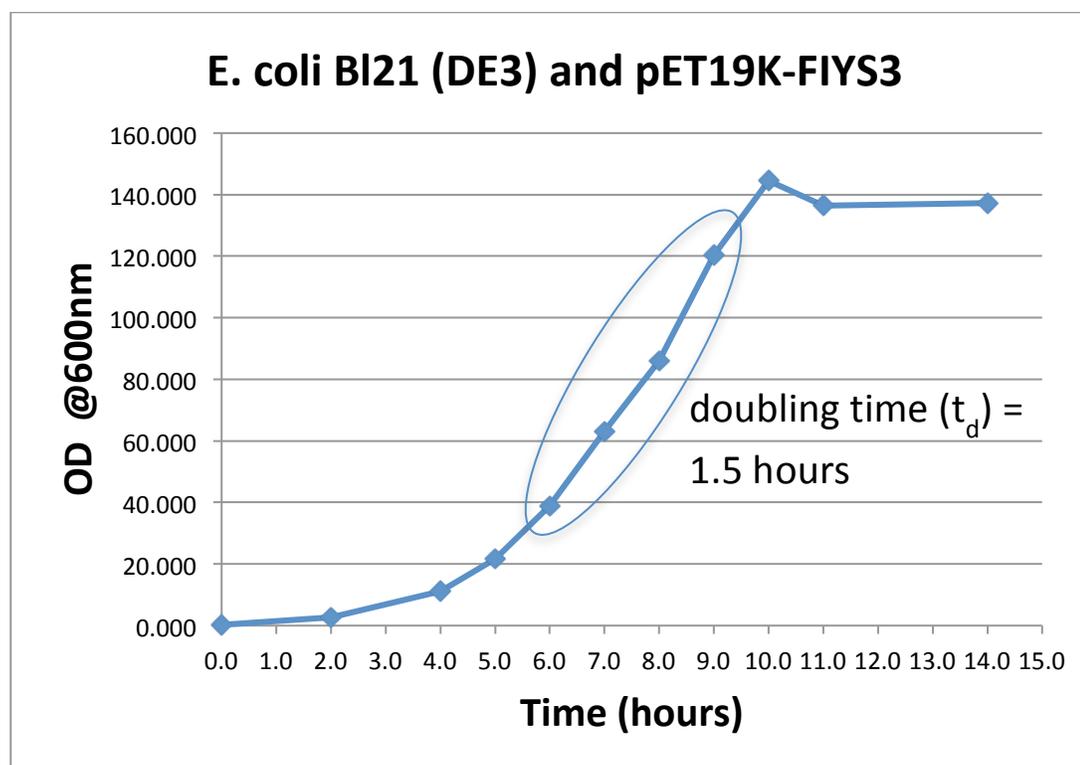
This reaction was created by finding the amino acid composition (obtained from Life Technologies, see Appendix C), and adding 4.3 ATP per amino acid (with the corresponding ADP, phosphate, hydrogen and water) to account for the energy cost of protein synthesis (Stephanopoulos *et al*, 1998, 69).

FIYS3 production was simulated by finding the maximum flux through the FIYS3 reaction under typical conditions, and setting that value as the lower bound. A growth curve from a previous successful fermentation experiment with *E. coli* BL21(DE3) CP001509 cells containing the pET19k-SX-FIYS3 plasmid was obtained (see Fig 8), and the exponential growth rate was found using the following equation:

$$\mu = \frac{\ln(2)}{t_d}$$

where  $\mu$  = exponential growth rate ( $\text{h}^{-1}$ ) and  $t_d$  = doubling time (h).

The growth rate was found to be  $0.456 \text{ g biomass produced gDW}^{-1}\text{h}^{-1}$ . Using FBA, the media conditions were set, the lower bound of the biomass reaction was set to  $0.456 \text{ g biomass produced gDW}^{-1}\text{h}^{-1}$ , and the flux through FIYS3 was optimized. The lower bound through iron(III) was not enough to support cell growth, so it was changed from  $-0.00696512$  to  $-0.00732381 \text{ mmol gDW}^{-1}\text{h}^{-1}$ , the minimum amount of flux required to support growth. The maximum flux through FIYS3 was found to be  $0.00336705 \text{ mmol gDW}^{-1}\text{h}^{-1}$ , and this value was used as the lower bound for the FIYS3 reaction during dynamic FBA simulations.



**Figure 8.** Previous fermentation experiment with *E. coli* BI21(DE3) and pET19K-FIYS3. The growth rate was found to be 0.456 g biomass produced  $\text{gDW}^{-1}\text{h}^{-1}$ .

#### *FBA and dynamic FBA*

FBA was used for preliminary modeling to determine how much glucose could be added without acetate being secreted, which exchange reactions needed flux through them to enable the model to run, and the maximum flux through the FIYS3 reaction. See Appendix A for more information.

Dynamic FBA was the primary tool used to analyze the growth rate and production of FIYS3. All lower bounds were set as previously described, and the biomass reaction was set as the objective function. The initial biomass field (initBiomass) was set to 0.01, the time step size (timeStep) was set to 0.5, and the number of time steps (nSteps) was set to 100. All metabolites in the media were analyzed.

Although a method has been developed to model the metabolic burden of carrying and replicating plasmids to the cell (Ow *et al*, 2009), the specific plasmid synthesis, vital to the method, was not known, so this procedure were not used in this study.

## **Laboratory experiments**

### *Cells and cell proliferation*

*E. coli* BL21(DE3) CP001509 cells containing the pET19k-SX-FIYS3 plasmid were used in this study. The pET19k-SX-FIYS3 plasmid, a modified form of the pET-19b plasmid, confers kanamycin resistance to the cells and contains the sequences for serine hydroxymethyltransferase (SHMT), tRNAs for glycine and two forms of proline (GlyT, ProL, and ProM), and FIYS3, an insert made up of 3 multiples of a 1000 bp fragment of flagelliform silk (for more information, see Appendix C).

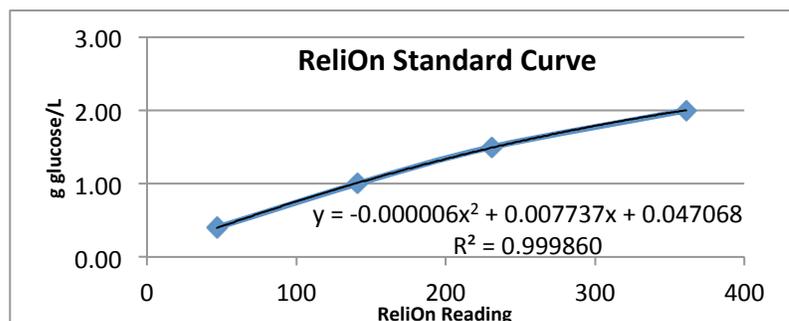
Unless otherwise noted, just prior to inoculation of all media, kanamycin was added so that the final kanamycin concentration was 75 µg/ml. A seed culture of cells was grown by inoculating 5 ml of Yeast-Hy ES II medium (see Appendix B for recipe) with around 5 cell colonies from an agar plate. The cells were grown in a 15 ml tube at 37°C and 220 rpm for 7.5 hours. The 5 ml seed culture was then added to 100 ml of Yeast-Hy ES II medium and grown in a covered 250 ml baffled flask at 30°C and 220 rpm for 9 hours.

The 100 ml seed was then transferred to 1.5 liters of K12 medium (see Appendix B for recipe) in a 5 liter BioFlo 310 Benchtop Fermentor/Bioreactor. The bioreactor was operated according to the user's guide provided by New Brunswick (New Brunswick, 2010). The dissolved oxygen was set to 45%, the pH was kept between 6.9 and 7.0, and the temperature was set to 37°C. The pH was controlled by 10 % NaOH (w/v) in some replicates and by 20% NH<sub>4</sub>OH (v/v) in others. Once the OD measured at a wavelength of 600 nm surpassed 4.0, the glucose feed was activated. The glucose feeding solution (see Appendix B for recipe) contained 100 µg/L of kanamycin. Glucose levels were monitored using a ReliOn Prime Blood Glucose Monitoring

System and were kept between 10 and 25 g/L. Samples of known glucose concentration were diluted 1:20 and were measured by the ReliOn Prime Blood Glucose Monitoring System. A standard curve was created to convert ReliOn readings to glucose concentration in g/L (see Table 6 and Fig 9).

**Table 6. ReliOn reading standard curve**

Glucose %	g glucose/L	ReliOn Reading	g/L from best fit equation
0.04	0.4	47	0.395646
0.1	1	141	1.013414
0.15	1.5	231	1.505534
0.2	2	361	2.044774



**Figure 9. Standard Curve for ReliOn Readings.**

Once the OD at 600 nm reached approximately 20, the cells were induced with 1 mM IPTG, 0.15 ml of 50 mg/ml kanamycin (7.5 mg in all) was added, and the temperature was lowered to 30°C to reduce the occurrence of protein aggregation. Two hours after IPTG induction, 0.15 ml of 50 mg/ml kanamycin (7.5 mg in all) was again added. Cells were grown until 4 hours after induction. At each time point, supernatant was obtained for various analyses by centrifuging samples at 15,000 rpm for 7 minutes, collecting the supernatant, and discarding the cell pellet.

#### *Ammonium analysis*

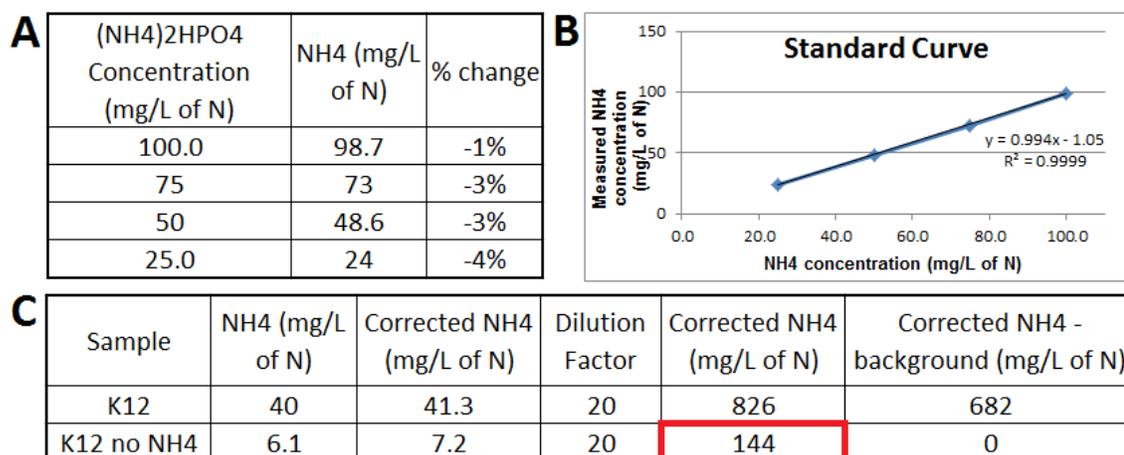
The ammonium concentration of the supernatants of the samples at every time point was measured using an Ammonium Ion-Selective Electrode from Vernier, which gives readings in

units of mg of nitrogen as ammonium per liter. For example, a solution with a concentration of 100 mg of ammonium per liter would have a concentration (and subsequent reading) of 77.65 mg of nitrogen as ammonium per liter. All standards with known concentrations were made using the unit mg of nitrogen as ammonium per liter. According to the user's manual, the suggested pH range for the electrode is between 4 and 7.5. Because of this, all standards were adjusted so that the pH was close to 4.0. All readings were recorded after the electrode had been submerged in the sample for 3 minutes.

The electrode was calibrated using two standards with a concentration of 100 and 1 mg of nitrogen as ammonium phosphate per liter. A standard curve was created by measuring the ammonium concentration in four standards (100, 75, 50, and 25 mg of nitrogen as ammonium phosphate per liter) and plotting those readings against the known concentrations each time before measuring the ammonium concentration of samples. Subsequent readings of samples were then corrected using the best fit line equation for the generated standard curve.

According to the user's manual, potassium ions interfere with the electrode's readings. Because K12 medium contains potassium, an ammonium reading of unaltered K12 medium with the standard amount of ammonium phosphate was compared to that of K12 medium with no ammonium phosphate to determine the background readings from the potassium in the medium. All samples were measured at a 1:20 dilution, so the media were also measured at a 1:20 dilution, the pH adjusted to 4.0. The results showed that the background reading for potassium was 144 mg/L (See Fig 10).

The supernatants of the samples were diluted 1:20 with acidic water (pH 3.2). The resulting samples had a pH of about 4.0. Once readings were recorded, they were corrected according to the best fit line equation for the standards measured prior to sample analysis, the value was multiplied by its dilution factor, and the background reading for the potassium (144 mg/L) was subtracted.



**Figure 10. Finding the background reading for the NH<sub>4</sub> electrode.**

- A Measured NH<sub>4</sub> concentrations of 4 (NH<sub>4</sub>)<sub>2</sub>HPO<sub>4</sub> standards.  
 B Standard curve generated from NH<sub>4</sub> electrode readings of standards.  
 C The NH<sub>4</sub> concentration of K12 media with and without NH<sub>4</sub> were measured to determine the background reading from potassium in the medium.

### *ELISA*

To quantify the protein, an ELISA test was performed on the samples from the last time point of each replicate. Prior to the test, all samples were diluted to an OD of 10 with 1X lysis buffer. The samples were then sonicated at 150 watts for 20 seconds. After sonication, samples were centrifuged at 14,000 rpm for 10 minutes. The supernatant was collected for analysis, and the cell pellets were discarded. A set of standards (200, 2000, 10,000, 20,000, 30,000, and 50,000 ng FLYS3 protein/ml) was also measured, and all samples and standards were analyzed in duplicates. A Q-Plex Array: Anti Poly-Histidine kit from Quansys Biosciences was used for the ELISA test, along with the accompanying Q-View™ imager and Q-View™ software to analyze the plate. The test was performed according to the user's manual (see Appendix B). MasterPlex 2010 was used to interpret the data.

### *Western blot*

SDS-PAGE and western blot were performed on the samples from the last time point of each replicate for protein verification. Prior to the procedure, the samples were prepared exactly as they were for the ELISA test: they were diluted to an OD of 10 with 1X lysis buffer, sonicated,

and centrifuged so the supernatant could be collected. In 1.5 ml tubes, 25  $\mu$ l of each sample was mixed with 25  $\mu$ l of 2X SaBU (sample buffer with urea) dye, and the tubes were boiled in a water bath for 10 minutes and quickly centrifuged to remove the sample from the lids of the tubes.

Precise™ Protein Gels from Thermo Scientific were used for SDS-PAGE. 5  $\mu$ l of Precision Plus Protein™ Dual Color Standard from Bio-Rad were loaded into the first lane, and 20  $\mu$ l of each sample were loaded into each subsequent lane. Sample loading and running was performed according to the Precise™ Protein Gels protocol (see Appendix B) at 100 volts for 60 minutes.

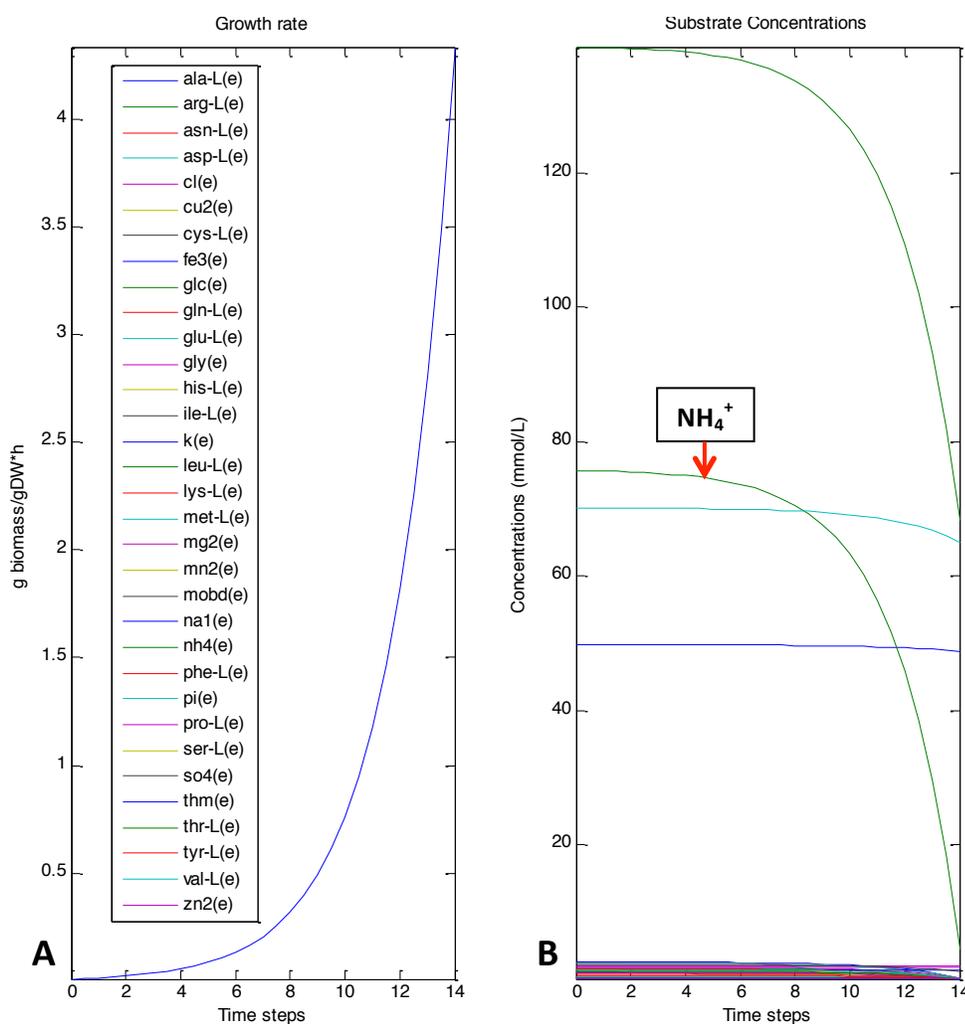
The western blot was performed according to the Wet Blotting Protocol (see Appendix B), using a nitrocellulose transfer membrane. The transfer was run at 100 mAmps per gel for 60 minutes.

The membrane was stained initially using Ponceau dye, then rinsed briefly to ensure the protein bands had transferred to the membrane. The membrane was then rinsed thoroughly, and 20 ml of milk-TBS-Tween mixture (1 gram of powdered milk in 20 ml TBS-Tween) was poured over the membrane in a small container. The container was placed on a gently rocking lab shaker. After 30 minutes, 5  $\mu$ l of the antibody Anti-6X-HIS Mouse Epitope Tag was added to the container and was allowed to shake for 20 minutes. The membrane was then rinsed twice with TBS-Tween, the second time shaking for 5 minutes before discarding the TBS-Tween. Another 20 ml of milk-TBS-Tween mixture and 5  $\mu$ l of the antibody Rb pAb to Ms IgG (AP) were then added to the container. After shaking for 20 minutes, the membrane was again rinsed with TBS-Tween, the second time shaking for 5 minutes before discarding the TBS-Tween. Enough 1-Step NBT/BCIP to submerge the membrane was then added, and the container was allowed to shake for 10 minutes, or until dark bands clearly appeared. Once bands could easily be seen, the membrane was rinsed with deionized water and was allowed to dry.

## RESULTS

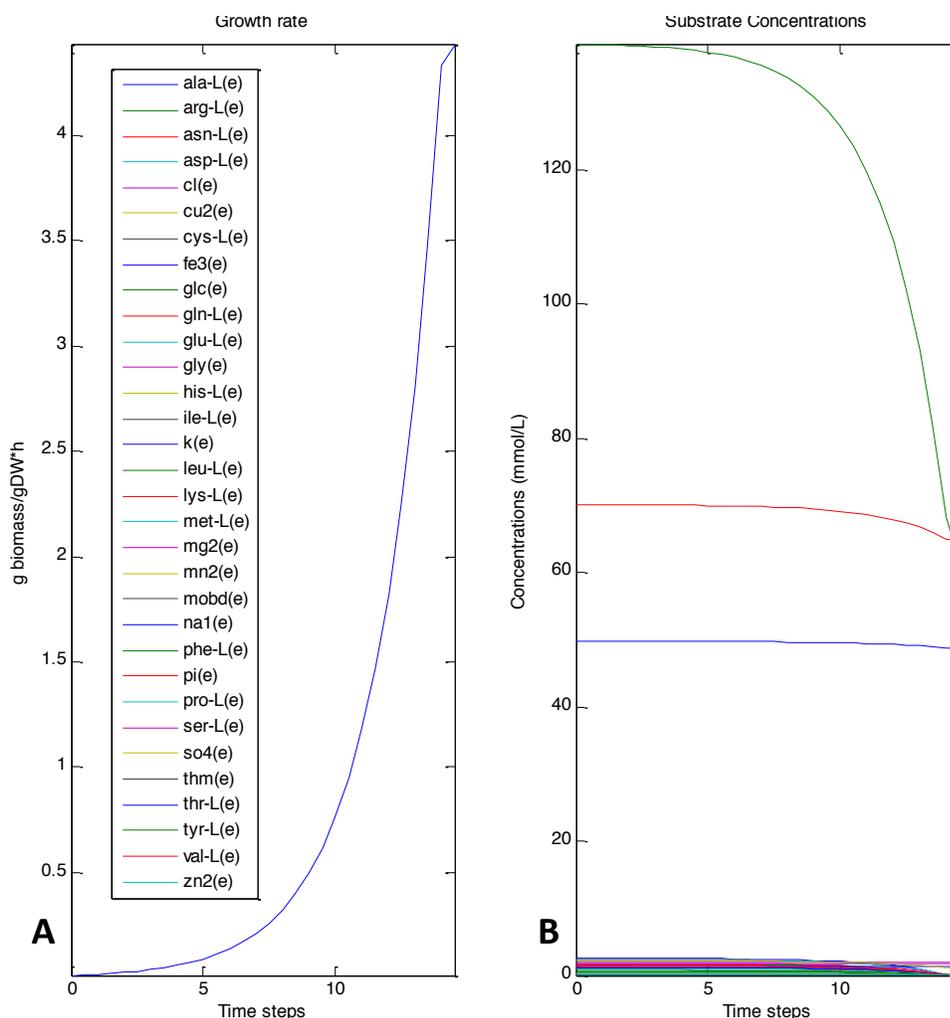
**Metabolic modeling**

Dynamic FBA revealed that ammonium was the first limiting metabolite. When the initial concentration for ammonium was set to 75.72 mmol/L, the maximum growth rate was 4.33 h<sup>-1</sup> (see Fig 11). However, once there was no specific initial concentration for ammonium (therefore, it was unlimited), the growth rate was 4.44 h<sup>-1</sup>, and iron(III) became the limiting metabolite (see Fig 12). Fig 13 shows the same data but is zoomed in to view iron(III) depletion.



**Figure 11. Dynamic FBA output for all metabolites in K12 medium.**

- A When the initial concentration of ammonium was 75.72 mmol/L, the maximum growth rate was 4.33 h<sup>-1</sup>.
- B The first limiting metabolite was ammonium.

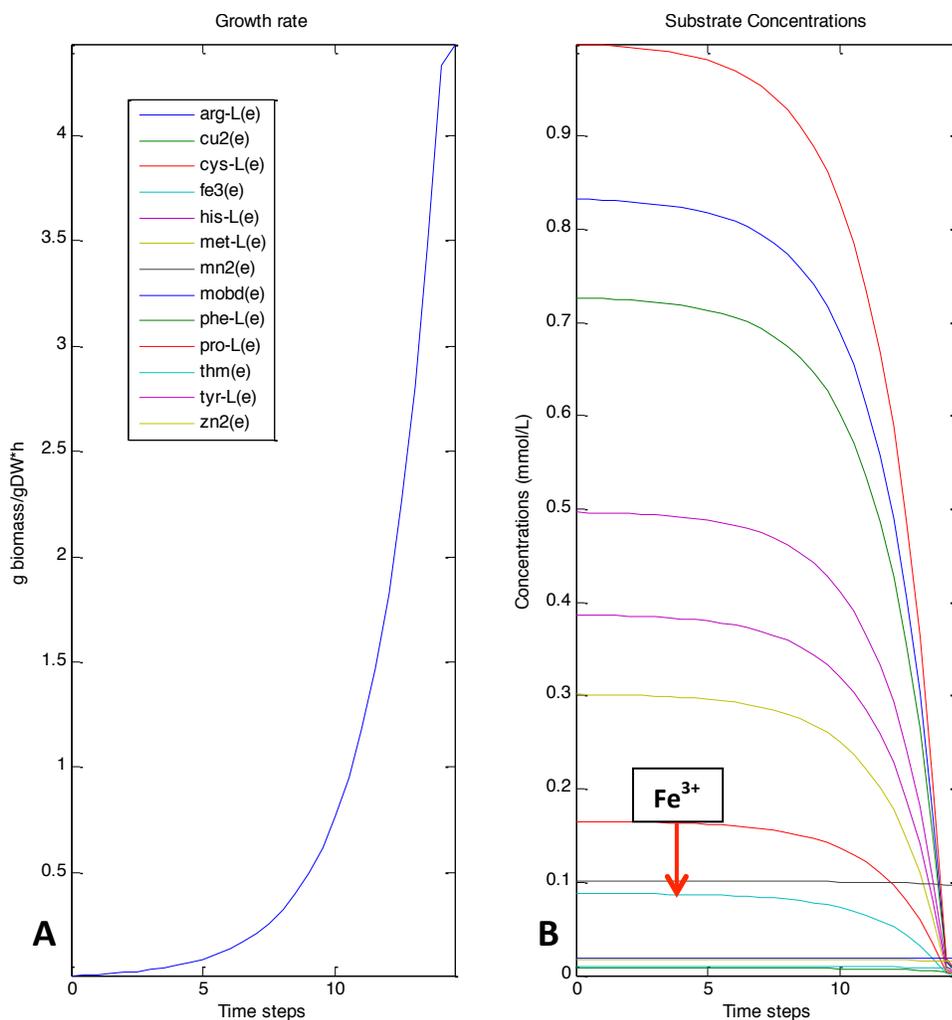


**Figure 12. Dynamic FBA output for all metabolites in K12 medium with unlimited ammonium.**

- A When the initial concentration of iron(III) was 0.09 mmol/L, the maximum growth rate was  $4.44 \text{ h}^{-1}$ .
- B After ammonium, iron(III) is the limiting metabolite.

When the specific initial concentrations for both ammonium and iron(III) were removed (making both metabolites essentially unlimited), the maximum growth rate was  $6.86 \text{ h}^{-1}$ , and glucose became the limiting metabolite (see Fig 14). Therefore, according to the results of dynamic FBA, the initial concentrations of ammonium and iron(III) will become depleted before glucose, thus limiting the growth rate. Although it is beyond the capabilities of dynamic FBA to show this, limiting the growth rate also limits protein production. In order to not hinder protein

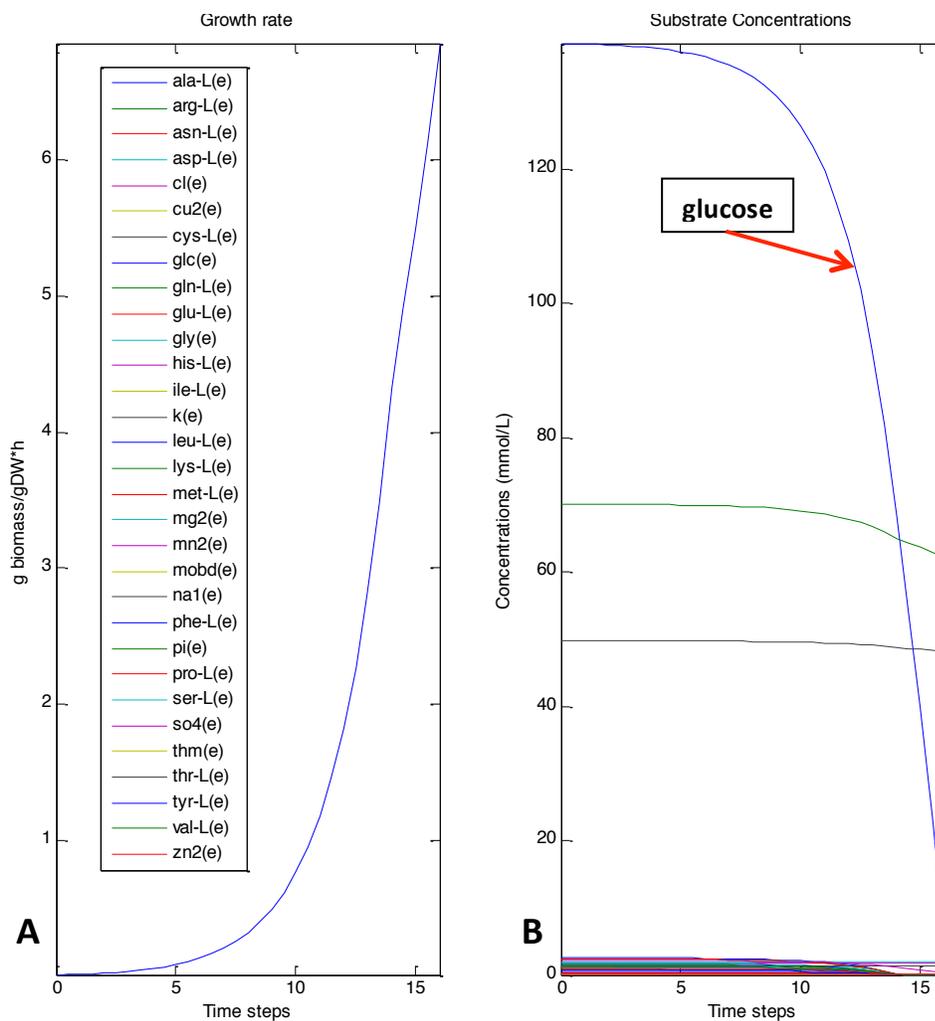
production, ammonium and iron(III) must either be replenished during cell growth or must have higher initial concentrations in the medium.



**Figure 13. Dynamic FBA output for all metabolites in K12 medium with unlimited ammonium, zoomed in to view iron(III) depletion.**

A When the initial concentration of iron(III) was 0.09 mmol/L, the maximum growth rate was 4.44 h<sup>-1</sup>.

B After ammonium, iron(III) is the limiting metabolite.

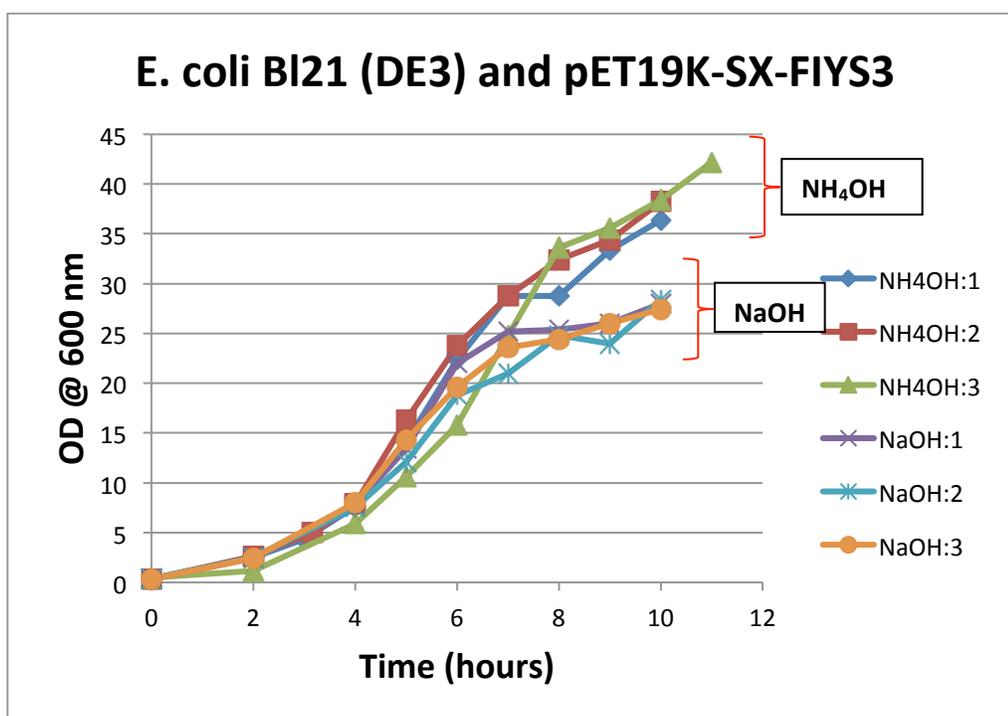


**Figure 14. Dynamic FBA output for all metabolites in K12 medium, with unlimited ammonium and unlimited iron(III).**

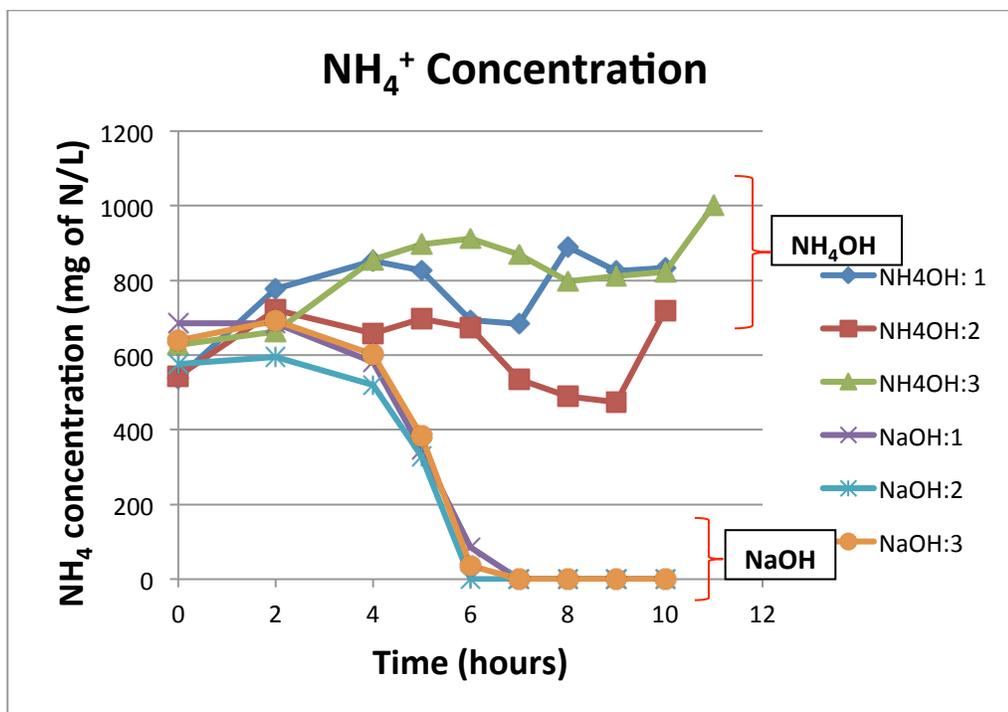
- A When the initial concentration of glucose is 138.77 mmol/L, the maximum growth rate was 6.86 h<sup>-1</sup>.
- B After ammonium and iron(III), the limiting metabolite is glucose.

## Laboratory experiments

Laboratory experiments were performed to determine the effect of ammonium and iron(III) on the production of FIYS3. Three fermentation experiments were performed growing *E. coli* using 10% sodium hydroxide as a pH control. Three additional fermentation experiments were performed using 20% ammonium hydroxide as the pH control, to determine if additional ammonium increased the production of the FIYS3 protein. The growth curves for the fermentation experiments are shown in Fig 15, and the ammonium concentrations are shown in Fig 16.



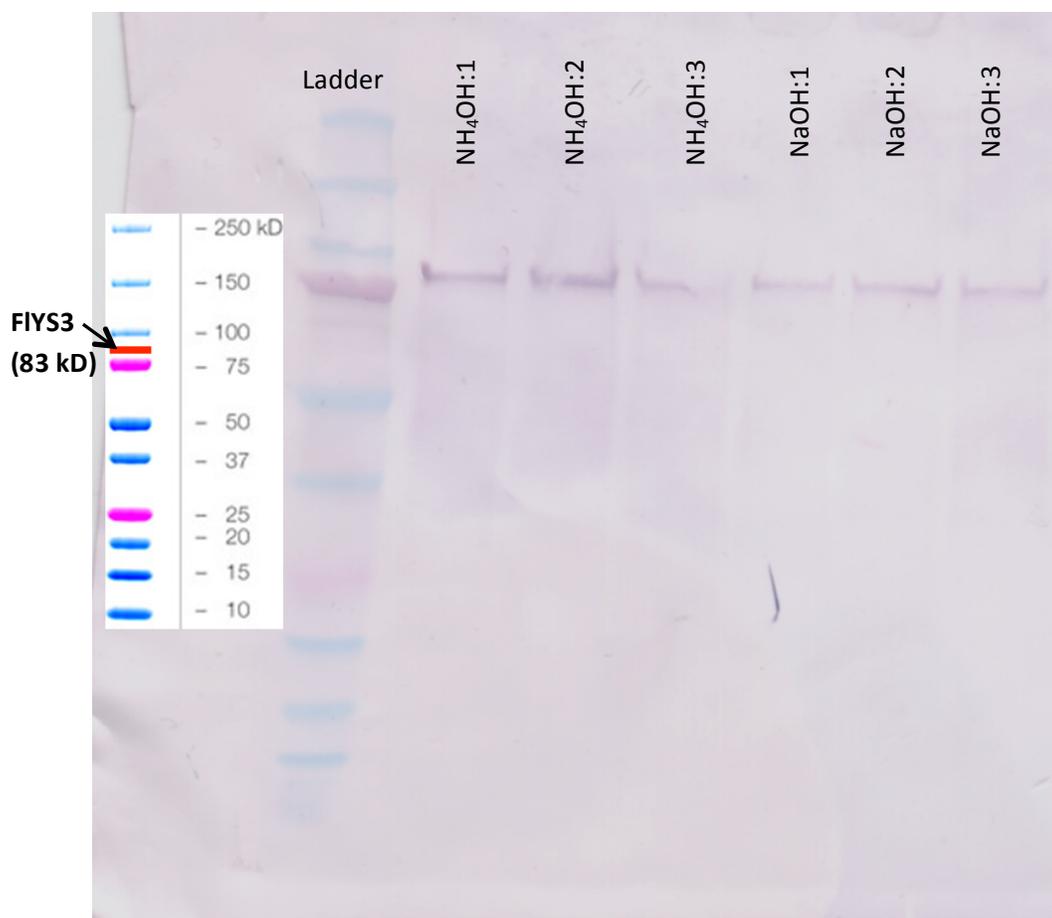
**Figure 15.** The optical density (OD) at 600 nm over time for *E. coli* cells grown using 20%  $\text{NH}_4\text{OH}$  or 10%  $\text{NaOH}$  as the pH control.



**Figure 16.** The ammonium concentration over time for *E. coli* cells grown using 20% NH<sub>4</sub>OH or 10% NaOH as the pH control.

An additional fermentation experiment with a pH control of 20% ammonium hydroxide was performed to determine how quickly iron is depleted from the medium. Preliminary results using Agilent 7700x ICP-MS (inductively coupled plasma mass spectroscopy) showed that iron was not depleted over time, so no further studies were conducted.

A western blot was performed on the samples from the last time point of each fermentation experiment to verify the correct protein (FIYS3, 83 kDa) was produced (see Fig 17). An ELISA was performed on the samples from the last time point of each fermentation experiment to quantify the protein. The ELISA results are shown in Table 7.



**Figure 17.** A western blot from the last time point of each fermentation experiment. This test verifies that the correct protein, FIYS3, 83 kDa was produced in each of the fermentation experiments.

**Table 7. ELISA results**

Sample	ELISA Reading	Calculated concentration (ng/ml)
Standard: 200 ng/ml	8984.50	3878.78
Standard: 2000 ng/ml	8639.00	1768.55
Standard: 10000 ng/ml	3746.00	10325.83
Standard: 20000 ng/ml	2190.50	17998.61
Standard: 30000 ng/ml	1528.50	29278.92
Standard: 50000 ng/ml	1101.00	67459.07
NaOH:1	8810.50	1999.25
NaOH:2	14312.00	0.00
NaOH:3	14711.50	0.00
NH <sub>4</sub> OH:1	6261.00	5410.47
NH <sub>4</sub> OH:2	6456.50	5125.83
NH <sub>4</sub> OH:3	7518.00	3613.03

## DISCUSSION

**Interpretation of results**

The metabolic modeling results correctly predicted that ammonium is a limiting metabolite. It is clear from the ammonium concentration data that if ammonium is not replenished, it will become depleted (see Fig 16). In the samples where 10% NaOH was used as the pH control, the ammonium concentration decreases over time and is completely depleted after 8 hours. The ammonium concentrations of the samples where 20% NH<sub>4</sub>OH was used for pH control were erratic because different amounts of NH<sub>4</sub>OH were pumped into the medium depending on the pH, but the concentration does not steadily decrease over time. Both the growth curves (Fig 15) and the ELISA results (Table 7) demonstrate the importance of ammonium in cell growth and protein production. An unpaired t test showed that the difference between the final ODs of the samples grown using 10% NaOH as a pH control and those of the samples grown using 20% NH<sub>4</sub>OH were statistically significant with a two-tailed P value of 0.0032 (see Table 8). Likewise, the difference between the final protein concentrations, as calculated from the ELISA results, was statistically significant, with a P value of 0.0096 (see Table 9). FIYS3 is a large protein with many repeating amino acids, so it is not surprising that an abundant source of nitrogen is necessary for high protein production.

**Table 8. Statistical analysis of final ODs for NaOH and NH<sub>4</sub>OH groups, P value = 0.0032**

	NaOH samples	NH <sub>4</sub> OH samples
Mean	27.933	38.933
SD	0.503	2.969
SEM	0.291	1.714
N	3	3

**Table 9. Statistical analysis of protein concentrations for NaOH and NH<sub>4</sub>OH groups, P value = 0.0096**

	NaOH samples	NH <sub>4</sub> OH samples
Mean	666.4167	4716.4433
SD	1154.2675	966.1241
SEM	666.4167	557.792
N	3	3

### Recommended research

Although it was demonstrated that adding additional ammonium to the medium increased cell growth and spider silk protein production, the upper threshold of how much ammonium could be added was not explored. While ammonium can be toxic for *E. coli* cells in concentrations above 750 mM (13.53 g/L), *E. coli* cells have been grown with ammonium concentrations as high as 500 mM (9.02 g/L) without any detrimental effects to cell growth (Müller *et al*, 2006). The initial ammonium concentration for the cells in this study was about 75.7 mM (1.37 g/L), so there is potential for increasing the ammonium concentration. It is likely that additional ammonium supplementation could further increase the growth rate and protein production of the cells.

While metabolic modeling correctly predicted the need for additional ammonium in the cell medium while producing spider silk, it is still a developing field with knowledge gaps. Improvements in modeling that are recommended for future research for studies involving recombinant protein production are the following: (1) include plasmid information in the model, (2) incorporate gene regulation, and (3) incorporate the ability to have more than one objective function while using dynamic FBA. As more is learned about cell metabolism, more information can be incorporated into metabolic models. As metabolic models continue to develop with new biological discoveries, the predictive capabilities will become even more powerful.

## REFERENCES

- Adrianos SL, Teulé F, Hinman MB, Jones JA, Weber WS, Yarger JL, Lewis RV (2013) *Nephila clavipes* flagelliform silk-like GGX motifs contribute to extensibility and spacer motifs contribute to strength in synthetic spider silk fibers. *Biomacromolecules* 14: 1751–1760
- Agnarsson I, Kuntner M, Blackledge TA (2010) Bioprospecting Finds the Toughest Biological Material: Extraordinary Silk from a Giant Riverine Orb Spider. *PLoS ONE* 5: e11234
- Allmeling C, Jokuszies A, Reimers K, Kall S, Choi CY, Brandes G, Kasper C, Scheper T, Guggenheim M, Vogt PM (2008) Spider silk fibres in artificial nerve constructs promote peripheral nerve regeneration. *Cell Prolif* 41: 408–420
- Altman GH, Diaz F, Jakuba C, Calabro T, Horan RL, Chen J, Lu H, Richmond J, Kaplan DL (2003) Silk-based biomaterials. *Biomaterials* 24: 401–416
- Andersen SO (1970) Amino acid composition of spider silks. *Comp Biochem Physiol* 35: 705–711
- Becker SA, Feist AM, Mo ML, Hannum G, Palsson BØ, Herrgard MJ (2007) Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox. *Nat Protoc* 2: 727–738
- Bini E, Foo CWP, Huang J, Karageorgiou V, Kitchel B, Kaplan DL (2006) RGD-functionalized bioengineered spider dragline silk biomaterial. *Biomacromolecules* 7: 3139–3145
- Chung H, Kim TY, Lee SY (2012) Recent advances in production of recombinant spider silk proteins. *Curr Opin Biotechnol* 23: 957–964
- Edwards JS, Palsson BØ (2000) Metabolic flux balance analysis and the in silico analysis of *Escherichia coli* K-12 gene deletions. *BMC Bioinformatics* 1: 1
- Fahnestock SR, Bedzyk LA (1997) Production of synthetic spider dragline silk protein in *Pichia pastoris*. *Appl Microbiol Biotechnol* 47: 33–39
- Feist AM, Herrgard MJ, Thiele I, Reed JL, Palsson BØ (2009) Reconstruction of Biochemical Networks in Microbial Organisms. *Nat Rev Microbiol* 7: 129–143
- Feist AM, Palsson BØ (2008) The growing scope of applications of genome-scale metabolic reconstructions: the case of *E. coli*. *Nat Biotechnol* 26: 659–667
- Feist AM, Zielinski DC, Orth JD, Schellenberger J, Herrgard MJ, Palsson BØ (2010) Model-driven evaluation of the production potential for growth-coupled products of *Escherichia coli*. *Metab Eng* 12: 173–186
- Fleischmann RD, Adams MD, White O, Clayton RA, Kirkness EF, Kerlavage AR, Bult CJ, Tomb JF, Dougherty BA, Merrick JM (1995) Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd. *Science* 269: 496–512

- Gomes S, Leonor IB, Mano JF, Reis RL, Kaplan DL (2011) Spider silk-bone sialoprotein fusion proteins for bone tissue engineering. *Soft Matter* 7: 4964–4973
- Gosline JM, Guerette PA, Ortlepp CS, Savage KN (1999) The mechanical design of spider silks: from fibroin sequence to mechanical function. *J Exp Biol* 202: 3295–3303
- Griffiths JR, Salanitri VR (1980) The strength of spider silk. *J Mater Sci* 15: 491–496
- Hayashi CY, Lewis RV (1998) Evidence from flagelliform silk cDNA for the structural basis of elasticity and modular nature of spider silks. *J Mol Biol* 275: 773–784
- Hayashi CY, Lewis RV (2001) Spider flagelliform silk: lessons in protein design, gene structure, and molecular evolution. *BioEssays News Rev Mol Cell Dev Biol* 23: 750–756
- Heim M, Ackerschott CB, Scheibel T (2010) Characterization of recombinantly produced spider flagelliform silk domains. *J Struct Biol* 170: 420–425
- Henig RM (2000) *The monk in the garden: the lost and found genius of Gregor Mendel, the father of genetics*. Houghton Mifflin Company, New York, New York, USA
- Higgins LE, White S, Nuñez-Farfán J, Vargas J (2007) Patterns of variation among distinct alleles of the Flag silk gene from *Nephila clavipes*. *Int J Biol Macromol* 40: 201–216
- Huemmerich D, Scheibel T, Vollrath F, Cohen S, Gat U, Ittah S (2004) Novel Assembly Properties of Recombinant Spider Dragline Silk Proteins. *Curr Biol* 14: 2070–2074
- Jenkins JE, Creager MS, Butler EB, Lewis RV, Yarger JL, Holland GP (2010) Solid-state NMR evidence for elastin-like  $\beta$ -turn structure in spider dragline silk. *Chem Commun* 46: 6714–6716
- Kim HU, Kim TY, Lee SY (2008a) Metabolic flux analysis and metabolic engineering of microorganisms. *Mol Biosyst* 4: 113–120
- Kim TY, Sohn SB, Kim HU, Lee SY (2008b) Strategies for systems-level metabolic engineering. *Biotechnol J* 3: 612–623
- Lazaris A, Arcidiacono S, Huang Y, Zhou J-F, Duguay F, Chretien N, Welsh EA, Soares JW, Karatzas CN (2002) Spider silk fibers spun from soluble recombinant silk produced in mammalian cells. *Science* 295: 472–476
- Lee SY, Lee D-Y, Kim TY (2005) Systems biotechnology for strain improvement. *Trends Biotechnol* 23: 349–358
- Lewis RV (2006) Spider silk: ancient ideas for new biomaterials. *Chem Rev* 106: 3762–3774
- Lu P, Vogel C, Wang R, Yao X, Marcotte EM (2007) Absolute protein expression profiling estimates the relative contributions of transcriptional and translational regulation. *Nat Biotechnol* 25: 117–124
- Menassa R, Zhu H, Karatzas CN, Lazaris A, Richman A, Brandle J (2004) Spider dragline silk proteins in transgenic tobacco leaves: accumulation and field production. *Plant Biotechnol J* 2: 431–438

- Monk JM, Charusanti P, Aziz RK, Lerman JA, Premyodhin N, Orth JD, Feist AM, Palsson BØ (2013) Genome-scale metabolic reconstructions of multiple *Escherichia coli* strains highlight strain-specific adaptations to nutritional environments. *Proc Natl Acad Sci U S A* 110: 20338–20343
- Müller T, Walter B, Wirtz A, Burkovski A (2006) Ammonium toxicity in bacteria. *Curr Microbiol* 52: 400–406
- New Brunswick (2010) *Guide to Operations: BioFlo 130 Benchtop Fermentor/Bioreactor*. New Brunswick Scientific, Edison, New Jersey, USA
- Orth JD, Palsson BØ (2012) Gap-filling analysis of the iJO1366 *Escherichia coli* metabolic network reconstruction for discovery of metabolic functions. *BMC Syst Biol* 6: 30
- Orth JD, Thiele I, Palsson BØ (2010) What is flux balance analysis? *Nat Biotechnol* 28: 245–248
- Ow DS-W, Lee D-Y, Yap MG-S, Oh SK-W (2009) Identification of cellular objective for elucidating the physiological state of plasmid-bearing *Escherichia coli* using genome-scale in silico analysis. *Biotechnol Prog* 25: 61–67
- Palsson BØ (2006) *Systems biology: properties of reconstructed networks*. Cambridge University Press, New York, New York, USA
- Park JH, Lee SY, Kim TY, Kim HU (2008) Application of systems biology for bioprocess development. *Trends Biotechnol* 26: 404–412
- Phillips R, Kondev J, Theriot J (2008) *Physical biology of the cell*. 1st ed. Garland Science, New York, New York, USA
- Price ND, Papin JA, Schilling CH, Palsson BØ (2003) Genome-scale microbial in silico models: the constraints-based approach. *Trends Biotechnol* 21: 162–169
- Rising A, Widhe M, Johansson J, Hedhammar M (2011) Spider silk proteins: recent advances in recombinant production, structure–function relationships and biomedical applications. *Cell Mol Life Sci* 68: 169–184
- Scheller J, Gührs K-H, Grosse F, Conrad U (2001) Production of spider silk proteins in tobacco and potato. *Nat Biotechnol* 19: 573–577
- Service RF (2002) Mammalian cells spin a spidery new yarn. *Science* 295: 419–421
- Stephanopoulos GN, Aristidou AA, Nielsen J (1998) *Metabolic engineering: principles and methodologies*. San Diego: Elsevier
- Thiele I, Palsson BØ (2010) A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nat Protoc* 5: 93–121
- Varma A, Boesch BW, Palsson BØ (1993) Stoichiometric interpretation of *Escherichia coli* glucose catabolism under various oxygenation rates. *Appl Environ Microbiol* 59: 2465–2473

- Varma A, Palsson BØ (1994) Stoichiometric flux balance models quantitatively predict growth and metabolic by-product secretion in wild-type *Escherichia coli* W3110. *Appl Environ Microbiol* 60: 3724
- Wen H, Lan X, Zhang Y, Zhao T, Wang Y, Kajiura Z, Nakagaki M (2010) Transgenic silkworms (*Bombyx mori*) produce recombinant spider dragline silk in cocoons. *Mol Biol Rep* 37: 1815–1821
- Widhe M, Johansson J, Hedhammar M, Rising A (2012) Current progress and limitations of spider silk for biomedical applications. *Biopolymers* 97: 468–478
- Williams D (2003) Sows' ears, silk purses and goats' milk: new production methods and medical applications for silk. *Med Device Technol* 14: 9
- Wohlrab S, Mueller S, Schmidt A, Neubauer S, Kessler H, Leal-Egana A, Scheibel T (2012) Cell adhesion and proliferation on RGD-modified recombinant spider silk proteins. *Biomaterials* 33: 6650–6659
- Wong Po Foo C, Kaplan DL (2002) Genetic engineering of fibrous proteins: spider dragline silk and collagen. *Adv Drug Deliv Rev* 54: 1131–1143
- Xia X-X, Qian Z-G, Ki CS, Park YH, Kaplan DL, Lee SY, Demain AL (2010) Native-sized recombinant spider silk protein produced in metabolically engineered *Escherichia coli* results in a strong fiber. *Proc Natl Acad Sci U S A* 107: 14059–14063
- Xu H-T, Fan B-L, Yu S-Y, Huang Y-H, Zhao Z-H, Lian Z-X, Dai Y-P, Wang L-L, Liu Z-L, Fei J, Li N (2007) Construct synthetic gene encoding artificial spider dragline silk protein and its expression in milk of transgenic mice. *Anim Biotechnol* 18: 1–12
- Zhang Y, Hu J, Miao Y, Zhao A, Zhao T, Wu D, Liang L, Miikura A, Shiomi K, Kajiura Z, Nakagaki M (2008) Expression of EGFP-spider dragline silk fusion protein in BmN cells and larvae of silkworm showed the solubility is primary limit for dragline proteins yield. *Mol Biol Rep* 35: 329–335

## APPENDICES

## Appendix A: Matlab codes

### Dynamic FBA

```

function [concentrationMatrix,excRxnNames,timeVec,biomassVec] = ...

dynamicFBA(model,substrateRxns,initConcentrations,initBiomass,timeStep,nSteps,p
lotRxns,exclUptakeRxns)
%dynamicFBA Perform dynamic FBA simulation using the static optimization
%approach
%
% [concentrationMatrix,excRxnNames,timeVec,biomassVec]
%
dynamicFBA(model,substrateRxns,initConcentrations,initBiomass,timeStep,nSteps,p
lotRxns,exclUptakeRxns)
%
%INPUTS
% model                COBRA model structure
% substrateRxns        List of exchange reaction names for substrates
%                      initially in the media that may change (e.g. not
%                      h2o or co2)
% initConcentrations   Initial concentrations of substrates (in the same
%                      structure as substrateRxns)
% initBiomass          Initial biomass (must be non zero)
% timeStep             Time step size
% nSteps              Maximum number of time steps
%
%OPTIONAL INPUTS
% plotRxns             Reactions to be plotted (Default =
%                      {'EX_glc(e)','EX_ac(e)','EX_for(e)'})
% exclUptakeRxns       List of uptake reactions whose substrate
%                      concentrations do not change (Default =
%                      {'EX_co2(e)','EX_o2(e)','EX_h2o(e)','EX_h(e)'})
%
%OUTPUTS
% concentrationMatrix  Matrix of extracellular metabolite concentrations
% excRxnNames          Names of exchange reactions for the EC metabolites
% timeVec              Vector of time points
% biomassVec           Vector of biomass values
%
% If no initial concentration is given for a substrate that has an open
% uptake in the model (i.e. model.lb < 0) the concentration is assumed to
% be high enough to not be limiting. If the uptake rate for a nutrient is
% calculated to exceed the maximum uptake rate for that nutrient specified
% in the model and the max uptake rate specified is > 0, the maximum uptake
% rate specified in the model is used instead of the calculated uptake
% rate.
%
% NOTE: The dynamic FBA method implemented in this function is essentially
% the same as the method described in
% [Varma, A., and B. O. Palsson. Appl. Environ. Microbiol. 60:3724 (1994)].
% This function does not implement the dynamic FBA using dynamic optimization
% approach
% described in [Mahadevan, R. et al. Biophys J, 83:1331-1340 (2003)].
%
% Markus Herrgard 8/22/06

if (nargin < 7)
    plotRxns = {'EX_glc(e)','EX_ac(e)','EX_for(e)'};
end

% Uptake reactions whose substrate concentrations do not change

```

```

if (nargin < 8)
    exclUptakeRxns = {'EX_co2(e)', 'EX_o2(e)', 'EX_h2o(e)', 'EX_h(e)'};
end

% Find exchange rxns
excInd = findExcRxns(model, false);
excInd = excInd & ~ismember(model.rxns, exclUptakeRxns);
excRxnNames = model.rxns(excInd);
length(excRxnNames)
% Figure out if substrate reactions are correct
missingInd = find(~ismember(substrateRxns, excRxnNames));
if (~isempty(missingInd))
    for i = 1:length(missingInd)
        fprintf('%s\n', substrateRxns{missingInd(i)});
    end
    error('Invalid substrate uptake reaction!');
end

% Initialize concentrations
substrateMatchInd = ismember(excRxnNames, substrateRxns);
concentrations = zeros(length(excRxnNames), 1);
concentrations(substrateMatchInd) = initConcentrations;

% Deal with reactions for which there are no initial concentrations
originalBound = -model.lb(excInd);
noInitConcentration = (concentrations == 0 & originalBound > 0);
concentrations(noInitConcentration) = 1000;

biomass = initBiomass;

% Initialize bounds
uptakeBound = concentrations/(biomass*timeStep);

% Make sure bounds are not higher than what are specified in the model
aboveOriginal = (uptakeBound > originalBound) & (originalBound > 0);
uptakeBound(aboveOriginal) = originalBound(aboveOriginal);
model.lb(excInd) = -uptakeBound;

concentrationMatrix = sparse(concentrations);
biomassVec = biomass;
timeVec(1) = 0;

fprintf('Step number\tBiomass\n');
h = waitbar(0, 'Dynamic FBA analysis in progress ...');
for stepNo = 1:nSteps
    % Run FBA
    sol = optimizeCbModel(model, 'max', 'one');
    mu = sol.f;
    if (sol.stat ~= 1 || mu == 0)
        fprintf('No feasible solution - nutrients exhausted\n');
        break;
    end
    uptakeFlux = sol.x(excInd);
    biomass = biomass*exp(mu*timeStep);
    %biomass = biomass*(1+mu*timeStep);
    biomassVec(end+1) = biomass;

    % Update concentrations
    concentrations = concentrations - uptakeFlux/mu*biomass*(1-
exp(mu*timeStep));
    %concentrations = concentrations + uptakeFlux*biomass*timeStep;
    concentrations(concentrations <= 0) = 0;
    concentrationMatrix(:,end+1) = sparse(concentrations);

```

```

% Update bounds for uptake reactions
uptakeBound = concentrations/(biomass*timeStep);
% This is to avoid any numerical issues
uptakeBound(uptakeBound > 1000) = 1000;
% Figure out if the computed bounds were above the original bounds
aboveOriginal = (uptakeBound > originalBound) & (originalBound > 0);
% Revert to original bounds if the rate was too high
uptakeBound(aboveOriginal) = originalBound(aboveOriginal);
uptakeBound(abs(uptakeBound) < 1e-9) = 0;

model.lb(excInd) = -uptakeBound;

fprintf('%d\t%f\n',stepNo,biomass);
waitbar(stepNo/nSteps,h);
timeVec(stepNo+1) = stepNo*timeStep;
end
if ( regexp( version, 'R20' ) )
    close(h);
end

selNonZero = any(concentrationMatrix>0,2);
concentrationMatrix = concentrationMatrix(selNonZero,:);
excRxnNames = excRxnNames(selNonZero);
selPlot = ismember(excRxnNames,plotRxns);

% Plot concentrations as a function of time
clf
subplot(1,2,1);
plot(timeVec,biomassVec);
axis tight
title('Biomass');
subplot(1,2,2);
plot(timeVec,concentrationMatrix(selPlot,:));
axis tight
legend(strrep(excRxnNames(selPlot),'EX_',''));

```

### *Optimize Cb model (FBA)*

```

function FBAolution = optimizeCbModel(model,osenseStr, minNorm, allowLoops)
%optimizeCbModel Solve a flux balance analysis problem
%
% Solves LP problems of the form: max/min c'*v
%                               subject to S*v = b
%                               lb <= v <= ub
% FBAolution = optimizeCbModel(model,osenseStr,minNormFlag)
%
%INPUT
% model (the following fields are required - others can be supplied)
% S      Stoichiometric matrix
% b      Right hand side = dx/dt
% c      Objective coefficients
% lb     Lower bounds
% ub     Upper bounds
%
%OPTIONAL INPUTS
% osenseStr  Maximize ('max')/minimize ('min') (opt, default = 'max')
%
% minNorm   {(0), 'one', > 0 , n x 1 vector}, where [m,n]=size(S);
%           0      Default, normal LP
%           'one'  Minimise the Taxicab Norm using LP.
%
%                               min |v|

```

```

%                                     s.t. S*v = b
%                                     c*v = f
%                                     lb <= v <= ub
%
% -----
% The remaining options work only with a valid QP solver:
% -----
% > 0   Minimises the Euclidean Norm of internal fluxes.
%       Typically 1e-6 works well.
%       min ||v||
%       s.t. S*v = b
%           c*v = f
%           lb <= v <= ub
% n x 1   Forms the diagonal of positive definite
%         matrix F in the quadratic program
%         min 0.5*v'*F*v
%         st. S*v = b
%           c'*v = f
%           lb <= v <= ub
%
% allowLoops   {0,(1)} If true, then instead of a conventional FBA,
%               the solver will run an MILP version which does not allow
%               loops in the final solution. Default is true.
%               Runs much slower when set to false.
%               See addLoopLawConstraints.m to for more info.
%
% OUTPUT
% FBAsolution
% f         Objective value
% x         Primal
% y         Dual
% w         Reduced costs
% s         Slacks
% stat      Solver status in standardized form
%           1   Optimal solution
%           2   Unbounded solution
%           0   Infeasible
%          -1   No solution reported (timelimit, numerical problem etc)
%
% Markus Herrgard           9/16/03
% Ronan Fleming              4/25/09 Option to minimise the Euclidean Norm of
internal
%                               fluxes using 'cplex_direct' solver
% Ronan Fleming              7/27/09 Return an error if any inputs are NaN
% Ronan Fleming              10/24/09 Fixed 'E' for all equality constraints
% Jan Schellenberger         12/07/09 MILP option to remove flux around loops
% Ronan Fleming              12/07/09 Reworked minNorm parameter option to allow
%                               the full range of approaches for getting
%                               rid of net flux around loops.
% Jan Schellenberger         2/3/09   fixed bug with .f being set incorrectly
%                               when minNorm was set.
% Nathan Lewis               12/2/10 Modified code to allow for inequality
%                               constraints.
% Ronan Fleming              12/03/10 Minor changes to the internal handling of
global parameters.
%% Process arguments and set up problem

if exist('osenseStr', 'var')
    if isempty(osenseStr)
        osenseStr = 'max';
    end
else
    osenseStr = 'max';
end

```

```

end

if exist('minNorm', 'var')
    if isempty(minNorm)
        minNorm = false;
        changeOK = changeCobraSolverParams('LP', 'minNorm', minNorm);
    else
        changeOK = changeCobraSolverParams('LP', 'minNorm', minNorm);
    end
end
else
    minNorm = false;
    changeOK = changeCobraSolverParams('LP', 'minNorm', minNorm);
end
if exist('allowLoops', 'var')
    if isempty(allowLoops)
        allowLoops = true;
    end
else
    allowLoops = true;
end

[minNorm, printLevel, primalOnlyFlag, saveInput] =
getCobraSolverParams('LP', {'minNorm', 'printLevel', 'primalOnly', 'saveInput'});

% if exist('minNorm', 'var')
%     if isempty(minNorm)
%         minNorm = false;
%     end
% else
%     minNorm = false;
% end
% if exist('allowLoops', 'var')
%     if isempty(allowLoops)
%         allowLoops = true;
%     end
% else
%     allowLoops = true;
% end
%
%
% global CBT_LP_PARAMS
% if (exist('CBT_LP_PARAMS', 'var'))
%     if isfield(CBT_LP_PARAMS, 'objTol')
%         tol = CBT_LP_PARAMS.objTol;
%     else
%         tol = 1e-6;
%     end
%     if isfield(CBT_LP_PARAMS, 'primalOnly')
%         primalOnlyFlag = CBT_LP_PARAMS.primalOnly;
%     else
%         primalOnlyFlag = false;
%     end
%     if isfield(CBT_LP_PARAMS, 'printLevel')
%         printLevel = CBT_LP_PARAMS.printLevel;
%     else
%         printLevel = 0;
%     end
% else
%     tol = 1e-6;
%     primalOnlyFlag = false;
%     printLevel = 0;
% end

```

```

% Figure out objective sense
if strcmpi(osenseStr,'max')
    LPproblem.osense = -1;
else
    LPproblem.osense = +1;
end

% this is dangerous... if model does not have S, it should not be called in
% this function.
% if ~isfield(model,'S')
%     model.S=model.A;
% end

[nMets,nRxns] = size(model.S);

% add csense
%Doing this makes csense a double array.  Totally smart design move.
%LPproblem.csense = [];
if ~isfield(model,'csense')
    % If csense is not declared in the model, assume that all
    % constraints are equalities.
    LPproblem.csense(1:nMets,1) = 'E';
else % if csense is in the model, move it to the lp problem structure
    if length(model.csense)~=nMets,
        warning('Length of csense is invalid! Defaulting to equality
constraints.')
```

```

        LPproblem.csense(1:nMets,1) = 'E';
    else
        model.csense = columnVector(model.csense);
        LPproblem.csense = model.csense;
    end
end

% Fill in the RHS vector if not provided
if (~isfield(model,'b'))
    LPproblem.b = zeros(size(model.S,1),1);
else
    LPproblem.b = model.b;
end

% Rest of the LP problem
LPproblem.A = model.S;
LPproblem.c = model.c;
LPproblem.lb = model.lb;
LPproblem.ub = model.ub;

%Double check that all inputs are valid:
if ~(verifyCobraProblem(LPproblem, [], [], false) == 1)
    warning('invalid problem');
    return;
end

%%
t1 = clock;
% Solve initial LP
if allowLoops
    solution = solveCobraLP(LPproblem);
else
    MILPproblem = addLoopLawConstraints(LPproblem, model, 1:nRxns);
    solution = solveCobraMILP(MILPproblem);
end

```

```

if (solution.stat ~= 1) % check if initial solution was successful.
    if printLevel>0
        warning('Optimal solution was not found');
    end
    FBAAsolution.f = 0;
    FBAAsolution.x = [];
    FBAAsolution.stat = solution.stat;
    FBAAsolution.origStat = solution.origStat;
    FBAAsolution.solver = solution.solver;
    FBAAsolution.time = etime(clock, t1);
    return;
end

objective = solution.obj; % save for later use.

if strcmp(minNorm, 'one')
    % Minimize the absolute value of fluxes to 'avoid' loopy solutions
    % Solve secondary LP to minimize one-norm of |v|
    % Set up the optimization problem
    % min sum(delta+ + delta-)
    % 1: S*v1 = 0
    % 3: delta+ >= -v1
    % 4: delta- >= v1
    % 5: c*v1 >= f (optimal value of objective)
    %
    % delta+,delta- >= 0
    LPproblem2.A = [model.S sparse(nMets,2*nRxns);
        speye(nRxns,nRxns) speye(nRxns,nRxns) sparse(nRxns,nRxns);
        -speye(nRxns,nRxns) sparse(nRxns,nRxns) speye(nRxns,nRxns);
        model.c' sparse(1,2*nRxns)];
    LPproblem2.c = [zeros(nRxns,1);ones(2*nRxns,1)];
    LPproblem2.lb = [model.lb;zeros(2*nRxns,1)];
    LPproblem2.ub = [model.ub;10000*ones(2*nRxns,1)];
    LPproblem2.b = [LPproblem.b;zeros(2*nRxns,1);solution.obj];
    if ~isfield(model,'csense')
        % If csense is not declared in the model, assume that all
        % constraints are equalities.
        LPproblem2.csense(1:nMets) = 'E';
    else % if csense is in the model, move it to the lp problem structure
        if length(model.csense)~=nMets,
            warning('Length of csense is invalid! Defaulting to equality
constraints.')
```

LPproblem2.csense(1:nMets) = 'E';

```

        else
            LPproblem2.csense = columnVector(model.csense);
        end
    end
    LPproblem2.csense((nMets+1):(nMets+2*nRxns)) = 'G';
    LPproblem2.csense(nMets+2*nRxns+1) = 'G';
    LPproblem2.csense = columnVector(LPproblem2.csense);
    LPproblem2.osense = 1;
    % Re-solve the problem
    if allowLoops
        solution = solveCobraLP(LPproblem2); % ,printLevel,minNorm);
        solution.dual = []; % slacks and duals will not be valid for this
computation.
        solution.rcost = [];
    else
        MILPproblem2 = addLoopLawConstraints(LPproblem, model, 1:nRxns);
        solution = solveCobraMILP(MILPproblem2);
    end
elseif length(minNorm)> 1 || minNorm > 0

```

```

% quadratic minimization of the norm.
% set previous optimum as constraint.
LPproblem.A = [LPproblem.A;
               LPproblem.c'];
LPproblem.csense(end+1) = 'E';
if nnz(LPproblem.c)>1
    error('Code assumes only one non-negative coefficient in linear part of
objective');
end
LPproblem.b = [LPproblem.b;solution.full(LPproblem.c~=0)];
LPproblem.c = zeros(size(LPproblem.c)); % no need for c anymore.
%Minimise Euclidean norm using quadratic programming
if length(minNorm)==1
    minNorm=ones(nRxns,1)*minNorm;
end
LPproblem.F = spdiags(minNorm,0,nRxns,nRxns);
%quadratic optimization
if allowLoops
    solution = solveCobraQP(LPproblem);
else
    MIQPproblem = addLoopLawConstraints(LPproblem, model, 1:nRxns);
    solution = solveCobraMIQP(MIQPproblem);
end
%     if isempty(solution.full)
%         % QP problem did not work. This will return empty structure
later.
%     else
%         %dont include dual variable to additional constraint
%         %solution.dual=solution.dual(1:end-1,1);
%     end
end

% Store results
if (solution.stat == 1)
    %solution found.
    FBAsolution.x = solution.full(1:nRxns);

    %this line IS necessary.
    FBAsolution.f = model.c'*solution.full(1:nRxns); %objective from original
optimization problem.
    if abs(FBAsolution.f - objective) > .01
        display('warning: objective appears to have changed while performing
secondary optimization (minNorm)');
    end

    if (~primalOnlyFlag && allowLoops && any(~minNorm)) % rcost/dual only
correct if not doing minNorm
        FBAsolution.y = solution.dual;
        FBAsolution.w = solution.rcost;
    end
else
    %some sort of error occurred.
    if printLevel>0
        warning('Optimal solution was not found!');
    end
    FBAsolution.f = 0;
    FBAsolution.x = [];
end

FBAsolution.stat = solution.stat;
FBAsolution.origStat = solution.origStat;
FBAsolution.solver = solution.solver;
FBAsolution.time = etime(clock, t1);

```

*Dynamic growth simulation*

```

% DynamicGrowthSimulation_new.m
clear

changeCobraSolver('glpk','LP'); % LP solver set to glpk
changeCobraSolver('glpk','MILP'); % MILP solver set to glpk
changeCobraSolver('qpng','QP'); % QP solver set to qpng
%Read iECD_1391 model
model = readCbModel('iECD_1391');
%Add FLYS3 reaction, change lower bound
model = addReaction(model,'FLYS3','120 ala-L[c] + 4 asp-L[c] + 522 gly[c] + 12
his-L[c] + ile-L[c] + lys-L[c] + 2 met-L[c] + 189 pro-L[c] + 111 ser-L[c] +
thr-L[c] + 78 tyr-L[c] + 4476.3 atp[c] + 4476.3 h2o[c] -> 4476.3 adp[c] +
4476.3 h[c] + 4476.3 pi[c]');
model = changeRxnBounds(model,'FLYS3',0.00336705,'l');
%Set objective
model = changeObjective(model,'Ec_biomass_iJO1366_core_53p95M');
%Setting carbon source and oxygen
model = changeRxnBounds(model,'EX_glc(e)',-11,'l');
model = changeRxnBounds(model,'EX_o2(e)',-20,'l');
%Setting exchange reactions to zero

model = changeRxnBounds(model,'EX_cb11(e)',0,'l');
model = changeRxnBounds(model,'EX_fe2(e)',0,'l');
model = changeRxnBounds(model,'EX_sel(e)',0,'l');
model = changeRxnBounds(model,'EX_slnt(e)',0,'l');
model = changeRxnBounds(model,'EX_tungs(e)',0,'l');

%A small amount of flux must go through EX_ca2(e), EX_cobalt2(e), and
EX_ni2(e) or the model won't run
model = changeRxnBounds(model,'EX_ca2(e)',-0.00237348,'l');
model = changeRxnBounds(model,'EX_cobalt2(e)',-1.14e-05,'l');
model = changeRxnBounds(model,'EX_ni2(e)',-0.000147288,'l');

%Simulating media conditions
model = changeRxnBounds(model,'EX_nh4(e)',-6.002150375,'l');
model = changeRxnBounds(model,'EX_pi(e)',-5.555386948,'l');
model = changeRxnBounds(model,'EX_k(e)',-3.943619038,'l');
model = changeRxnBounds(model,'EX_so4(e)',-0.167768684,'l');

model = changeRxnBounds(model,'EX_cl(e)',-0.138740225,'l');
model = changeRxnBounds(model,'EX_cu2(e)',-0.000634963,'l');
model = changeRxnBounds(model,'EX_fe3(e)',-0.00696512,'l');
model = changeRxnBounds(model,'EX_mg2(e)',-0.160804933,'l');
model = changeRxnBounds(model,'EX_mn2(e)',-0.008010947,'l');
model = changeRxnBounds(model,'EX_mobd(e)',-0.001508618,'l');
model = changeRxnBounds(model,'EX_na1(e)',-0.102668246,'l');
model = changeRxnBounds(model,'EX_thm(e)',-0.000746848,'l');
model = changeRxnBounds(model,'EX_zn2(e)',-0.001378378,'l');
%Amino acids
model = changeRxnBounds(model,'EX_ala-L(e)',-0.200200247,'l');
model = changeRxnBounds(model,'EX_arg-L(e)',-0.065982824,'l');
model = changeRxnBounds(model,'EX_asn-L(e)',-0.095998062,'l');
model = changeRxnBounds(model,'EX_asp-L(e)',-0.09529124,'l');
model = changeRxnBounds(model,'EX_cys-L(e)',-0.013085243,'l');
model = changeRxnBounds(model,'EX_gln-L(e)',-0.187137594,'l');
model = changeRxnBounds(model,'EX_glu-L(e)',-0.185878393,'l');
model = changeRxnBounds(model,'EX_gly(e)',-0.14255367,'l');
model = changeRxnBounds(model,'EX_his-L(e)',-0.030655649,'l');
model = changeRxnBounds(model,'EX_ile-L(e)',-0.08762833,'l');

```

```

model = changeRxnBounds(model, 'EX_leu-L(e)', -0.126909995, '1');
model = changeRxnBounds(model, 'EX_lys-L(e)', -0.119293303, '1');
model = changeRxnBounds(model, 'EX_met-L(e)', -0.02390703, '1');
model = changeRxnBounds(model, 'EX_phe-L(e)', -0.05758489, '1');
model = changeRxnBounds(model, 'EX_pro-L(e)', -0.079180891, '1');
model = changeRxnBounds(model, 'EX_ser-L(e)', -0.098060253, '1');
model = changeRxnBounds(model, 'EX_thr-L(e)', -0.089838012, '1');
model = changeRxnBounds(model, 'EX_tyr-L(e)', -0.039374888, '1');
model = changeRxnBounds(model, 'EX_val-L(e)', -0.111648451, '1');

%FOR FBA
%FBAsolution = optimizeCbModel(model, 'max');
%printFluxVector(model, FBAsolution.x, true);

% Set-up variables for dynamicFBA
%NOTE- substrate rxns and plot rxns need to be in the order that they
%appear in the model

initBiomass = .01;
timeStep = 0.5; nSteps = 100;

substrateRxns = {'EX_ala-L(e)', 'EX_arg-L(e)', 'EX_asn-L(e)', 'EX_asp-
L(e)', 'EX_cl(e)', 'EX_cu2(e)', 'EX_cys-L(e)', 'EX_fe3(e)', 'EX_glc(e)', 'EX_gln-
L(e)', 'EX_glu-L(e)', 'EX_gly(e)', 'EX_his-L(e)', 'EX_ile-L(e)', 'EX_k(e)', 'EX_leu-
L(e)', 'EX_lys-L(e)', 'EX_met-
L(e)', 'EX_mg2(e)', 'EX_mn2(e)', 'EX_mobd(e)', 'EX_na1(e)', 'EX_nh4(e)', 'EX_phe-
L(e)', 'EX_pi(e)', 'EX_pro-L(e)', 'EX_ser-L(e)', 'EX_so4(e)', 'EX_thm(e)', 'EX_thr-
L(e)', 'EX_tyr-L(e)', 'EX_val-L(e)', 'EX_zn2(e)'};
initConcentrations =
[2.525535975, 0.832376579, 1.211020285, 1.202103681, 1.750214773, 0.008010093, 0.1650
70981, 0.087865335, 138.7655417, 2.360749966, 2.344865085, 1.798321567, 0.386722527, 1
.105435694, 49.74894838, 1.600975833, 1.504890895, 0.301588365, 2.028562155, 0.101058
487, 0.019031286, 1.295164976, 75.71742258, 0.726436225, 70.08147994, 0.998870842, 1.2
37034922, 2.11641021, 0.009421519, 1.133310947, 0.496716154, 1.408450704, 0.017388304
];
plotRxns = {'EX_ala-L(e)', 'EX_arg-L(e)', 'EX_asn-L(e)', 'EX_asp-
L(e)', 'EX_cl(e)', 'EX_cu2(e)', 'EX_cys-L(e)', 'EX_fe3(e)', 'EX_glc(e)', 'EX_gln-
L(e)', 'EX_glu-L(e)', 'EX_gly(e)', 'EX_his-L(e)', 'EX_ile-L(e)', 'EX_k(e)', 'EX_leu-
L(e)', 'EX_lys-L(e)', 'EX_met-
L(e)', 'EX_mg2(e)', 'EX_mn2(e)', 'EX_mobd(e)', 'EX_na1(e)', 'EX_nh4(e)', 'EX_phe-
L(e)', 'EX_pi(e)', 'EX_pro-L(e)', 'EX_ser-L(e)', 'EX_so4(e)', 'EX_thm(e)', 'EX_thr-
L(e)', 'EX_tyr-L(e)', 'EX_val-L(e)', 'EX_zn2(e)'};

dynamicFBA(model, substrateRxns, initConcentrations, initBiomass, timeStep,
nSteps, plotRxns);
%labeling
subplot(1,2,1);
title('Growth rate');
xlabel('Time steps');
ylabel('g biomass/gDW*h');
subplot(1,2,2);
title('Substrate Concentrations');
xlabel('Time steps');
ylabel('Concentrations (mmol/L)');

```

## Appendix B: Recipes and protocols

### *Yeast-Hy ES II medium*

#### **Initial medium:**

Yeast Extract	5 g/L
Hy-ES II	15g/L

#### **After autoclaved add:**

Glycerol	16g/L
Glucose	25g/L
Trace element 1	1.5ml
Trace element 2	1.5ml
x20 N.P.S stock	50ml
0.5g/ml MgSO <sub>4</sub>	1ml
5mg/ml Thiamine	0.8ml

#### **Trace element 1**

CuSO <sub>4</sub> .5H <sub>2</sub> O	0.25g/100ml
MnSO <sub>4</sub> .H <sub>2</sub> O	2.4g/100ml
Na <sub>2</sub> MoO <sub>4</sub> .2H <sub>2</sub> O	0.30g/100ml
Ni(NO <sub>3</sub> ) <sub>2</sub>	2.5g/100ml
ZnSO <sub>4</sub>	1.5ml/100ml
6N H <sub>2</sub> SO <sub>4</sub>	0.3ml/100ml

#### **Trace element 2**

NaCl	0.5g/100ml
FeCl <sub>3</sub> .6H <sub>2</sub> O	0.475g/100ml
CoCl <sub>2</sub> .6H <sub>2</sub> O	0.075g/100ml
H <sub>3</sub> BO <sub>3</sub>	0.05g/100ml
CaCl <sub>2</sub> .2H <sub>2</sub> O	0.29g/100ml
6N H <sub>2</sub> SO <sub>4</sub>	0.3ml/100ml

#### **x20 N.P.S. stock solution**

(NH <sub>4</sub> ) <sub>2</sub> SO <sub>4</sub>	66g/L
KH <sub>2</sub> PO <sub>4</sub>	136g/L
NaH <sub>2</sub> PO <sub>4</sub>	142g/L

### *K12 medium and K12 trace metal solution*

**Initial medium:**

KH <sub>2</sub> PO <sub>4</sub>	2g/L
K <sub>2</sub> HPO <sub>4</sub> ·3H <sub>2</sub> O	4g/L
(NH <sub>4</sub> ) <sub>2</sub> HPO <sub>4</sub>	5g/L
Yeast Extract	5g/L

**After autoclaved add:**

Glucose	25g/L
MgSO <sub>4</sub> ·7H <sub>2</sub> O	0.5g/L
Thiamine	2.5mg/L
K12 trace metal	5 ml/L

**K12 trace metal solution**

NaCl	5 g/L
ZnSO <sub>4</sub> ·7H <sub>2</sub> O	1 g/L
MnCl <sub>2</sub> ·4H <sub>2</sub> O	4 g/L
FeCl <sub>3</sub> ·6H <sub>2</sub> O	4.75 g/L
CuSO <sub>4</sub> ·5H <sub>2</sub> O	0.4 g/L
H <sub>3</sub> BO <sub>3</sub>	0.575 g/L
NaMoO <sub>4</sub> ·2H <sub>2</sub> O	0.5 g/L
6N H <sub>2</sub> SO <sub>4</sub>	12.5 ml/L

**Sterilize with 0.2 µm syringe filter**

*Glucose feeding solution*

Glucose	500g/L
MgSO <sub>4</sub> ·7H <sub>2</sub> O	10g/L
Thiamine	40mg/L
K12 trace metal	5ml/L

## IMPORTANT PRECAUTIONS

- 1. Read all instructions before beginning test.**
- 2. For research use only.**
3. The kit should not be used beyond the expiration date on the kit label. Send requests for replacement reagents to: [INFO@QUANSYSBIO.COM](mailto:INFO@QUANSYSBIO.COM).
4. Do not mix or substitute reagents with those from other kits or lots.
5. This kit is validated for use with bacterial lysates and purified proteins.
6. All products are carefully validated, however due to the variability encountered in biological buffers and sample matrices, the possibility of interference or sample matrix effects cannot be excluded.

## Q-VIEW™ SOFTWARE

A free copy of the Q-View Software, a tool for the quantitative analysis of multiplex ELISAs and for controlling Q-View Imagers, is available to all Q-Plex users. Please send requests to download the software to [INFO@QUANSYSBIO.COM](mailto:INFO@QUANSYSBIO.COM).

A summary of how to use the Q-View Software to analyze an image is contained in this manual (Page 9). The full Q-View Software Manual is also available at [www.quansysbio.com/manuals](http://www.quansysbio.com/manuals).

## DO'S

- DO set up, calibrate, and practice using the Q-View imager BEFORE starting the assay.
- DO be exact when setting shaker speed to 500 RPM, being off by even 100 RPM can affect results.
- DO dilute all sample types at least 1:2 (one part sample to one part diluent) with the provided sample diluent and mix thoroughly.
- DO load all samples into the microplate within 10 minutes of each other.
- DO be exact with incubation times, particularly the SHRP incubation.
- DO be exact when mixing Substrate A and B, being off by even 100  $\mu$ L can affect results, and mix thoroughly.

## DON'TS

- DON'T allow the plate to dry out between steps.
- DON'T allow the substrate or SHRP to be exposed to UV light, as this may degrade it.
- DON'T analyze from a jpeg, bmp, or png image; only TIFF or other full resolution, lossless file types with at least 16-bit depth are acceptable for analysis.

Further tips on troubleshooting, data analysis, and assay sensitivity are available at [www.quansysbio.com/tech-tips](http://www.quansysbio.com/tech-tips).

We take great care to ensure that customers have success using our products and services. If you have any further questions about the assay or our products or services, please contact us at [888-QUANSYS \(782-6797\)](tel:888-QUANSYS) or at [INFO@QUANSYSBIO.COM](mailto:INFO@QUANSYSBIO.COM).

## IMAGING SYSTEMS

1. Recommended imager optimized for use with Q-Plex Arrays:  
Quansys Q-View™ Imager (Cat #104450GR) with Q-View™ Software
2. Suggested imagers that are commonly used with Q-Plex™ Arrays:  
Bio-Rad: Versa Doc 4000 or Chemi Doc XRS  
GE Healthcare (formerly Fujifilm): (with NP Tray accessory) LAS-3000, LAS-3000 Mini, LAS-4000  
LI-COR: Aeries®, Odyssey-CLx®
3. Other imagers that may be compatible, for use with Q-Plex™ Arrays:  
Alpha Innotech: Fluorchem HD, SP, 8000, 8900, 9900, HD2, and FC2  
Fujifilm: LAS-4000 Mini  
Carestream (formerly Kodak): 4000MM, 2000MM, Gel Logic 100  
UVP: BioDoc-IT System, EC3 Darkroom

For imager-specific imaging instructions, see [www.quansysbio.com/compatible-imagers](http://www.quansysbio.com/compatible-imagers).

For minimum imager requirements, see [www.quansysbio.com/third-party-imagers](http://www.quansysbio.com/third-party-imagers).

To receive a FREE Calibration Kit that can be used to set up and validate an imager, contact us at [888-QUANSYS \(782-6797\)](tel:888-QUANSYS) or at [INFO@QUANSYSBIO.COM](mailto:INFO@QUANSYSBIO.COM).

## KIT CONTENTS, PREPARATION, & STORAGE

Part	Description	Reagent Preparation	Storage of opened/reconstituted material*
Q-Plex Anti Poly-Histidine Microplate	Spotted and blocked 96-well polystyrene microtiter plate	Ready for use	If the plate contains unused wells, return the plate to the sealed foil pouch at 4°C
Wash Buffer Concentrate (20X)	Liquid, 50 mL/vial of a concentrated solution of buffered surfactant	Place 50 mL of the 20X concentrate into 950 mL deionized water, mix thoroughly.	4°C until kit expiration
Poly-His Competitor	Lyophilized, biotinylated competitive antigen	Add 1 mL Sample Diluent to vial, mix thoroughly. Add 1 mL of reconstituted competitor mix back into remaining 9 mL vial of Sample Diluent	Discard unused competitor mix. Good for one day.
Anti Poly-His Sample Diluent	Liquid, 10 mL/vial of a buffered protein solution with preservatives	After mixing Competitor in, it is ready for use	4°C until kit expiration
Streptavidin-HRP 1X	Liquid, 6 mL/vial of streptavidin-conjugated horse radish peroxidase	Ready for use. Do not expose to UV light.	Do not expose to UV light. 4°C until kit expiration
Substrate A	Liquid, 3 mL/vial of stabilized hydrogen peroxide	Do not expose to UV light. Do not cool after mixing. During the assay, mix 3 mL of Substrate A with 3 mL of Substrate B, and mix gently.	Do not expose to UV light. Do not cool after mixing. Store mixed substrate solution at room temperature for up to 1 week. Store unmixed solution at 4°C until kit expiration.
Substrate B	Liquid, 3 mL/vial of stabilized signal enhancer		
Plate Seals (2)	Adhesive strips	None	Non-perishable

\*Provided this is within the expiration date of the kit

## OTHER REQUIRED MATERIALS: INSTRUMENTS AND ACCESSORIES

In addition to the kit contents listed, the following materials are required to run Q-Plex Assays or panels for optimal results. We recommend the use of these specific items.

1. 8- or 12-channel pipette (20-200  $\mu$ L) and/or 1-channel pipette (20-200  $\mu$ L) and tips
2. Low-binding polypropylene tubes or low-binding 96-well plate(s).
3. Imaging system
4. Q-View™ Software
5. Microplate shaker
  - a. Barnstead/labline 4625 titer plateshaker, IKA MTS 2/4 for 2 or 4 microtiter plates, or equivalent, capable of 500-1,100 RPM.
6. Optional: spare microtiter plate to test an automatic plate washer (See Appendix B Step 4).

## ASSAY PREPARATION

1. Install the FREE Q-View Software on any number of computers to be used for analysis or driving a Q-View Imager (Page 4).
2. Set up the imager (Page 9). For imager-specific instructions, see [www.quansysbio.com/compatible-imagers](http://www.quansysbio.com/compatible-imagers). To receive a free Calibration Kit that can be used to validate imager settings, contact us at 888-QUANSYS (782-6797) or at [INFO@QUANSYSBIO.COM](mailto:INFO@QUANSYSBIO.COM).
3. Set the plate shaker to 500 RPM.
4. Choose a plate washing method (Appendix B).
5. Prepare the Wash Buffer: Place 50 mL of the 20X concentrate into 950 mL deionized water, and mix thoroughly. Store at 4°C.

## ASSAY PROCEDURE

**Bring all reagents to room temperature before use. It is recommended that all standards, samples, and controls be assayed in duplicate.**

1. Reconstitute the lyophilized Poly-His Competitor.
  - a. Add 1 mL of Anti Poly-His Sample Diluent to the lyophilized Poly-His Competitor vial, mix thoroughly. Add 1 mL of the reconstituted competitor back into the remaining 9 mL of Anti Poly-His Sample Diluent
2. Prepare samples by diluting at least 1:2 (one part sample to one part diluent) with enough Anti Poly-His Sample Diluent containing competitor to have 50  $\mu$ L per well in either low-binding polypropylene tubes or a low-binding 96-well plate.
3. Add 50  $\mu$ L per well of the pre-diluted samples to the Q-Plex Anti Poly-Histidine 96-well plate. Load all samples to the plate within ten minutes.
4. Cover the plate with the plate seal provided, and place on a plate shaker set to 500 RPM for 30 minutes at room temperature (23°C).

*Note: Record the plate layout in the Well Assignment section of the Q-View Software, in the Q-View compatible Excel template (**Well Assignment > Templates > New Template**), or using the plate diagram (See page 18).*

5. Wash the plate three times according to the preferred washing method (see Appendix B).
6. Add 50  $\mu$ L per well of Streptavidin-HRP 1X, cover with a new seal, and return to the plate shaker set to 500 RPM for 20 minutes at room temperature.
7. Allow Substrate A and B to come to room temperature. Fifteen minutes prior to use, combine 3 mL of Substrate A with 3 mL of Substrate B, and mix gently. **Do not expose to UV light. Store at room temperature after mixing.**
8. Wash the plate six times as in step 6.
9. Add 50 $\mu$ L per well of mixed substrate, and image the plate, as described below, immediately for optimal results. Wait no longer than 15 minutes

to commence imaging.

**Note:** If imaging cannot commence immediately, protect the plate from drying for up to 15 minutes by dispensing 100  $\mu$ L of wash buffer into each well of the plate prior to adding the mixed substrate. When ready to image, remove the wash buffer from the plate and add the substrate.

## ACQUIRING AN IMAGE USING THE Q-VIEW™ IMAGER

1. Place the plate in the calibrated Q-View Imager and shut the lid.
2. Open the Q-View Software, create or open a project, and click Acquire Image.

Recommended settings for Q-Plex™ kits are: Standard

Exposure Time(s)(seconds): 180

3. Click the Capture Image(s) button. Users can continue on to Well Assignment while images are being captured.

Details about these imaging steps are available in the Q-View Software Manual viewable at [www.quansysbio.com/manuals](http://www.quansysbio.com/manuals) or within the Q-View Software under **Support > Manual**.

## ACQUIRING AN IMAGE FROM AN ALTERNATE IMAGER

Imager-specific imaging instructions for alternate imagers are available at [www.quansysbio.com/compatible-imagers](http://www.quansysbio.com/compatible-imagers).

## ANALYZING A Q-PLEX IMAGE

The following summarizes a general workflow for analyzing a Q-Plex image in the Q-View Software. Each of these steps is described in greater detail in the Q-View Software and Imager Manual, viewable at [www.quansysbio.com/manuals](http://www.quansysbio.com/manuals), or within the Q-View Software under **Support > Manual**.

1. Acquire or import an image into Q-View as described above.

*Note: Images with black spots on a white background MUST be inverted either by selecting the option during import, or by going to **Image Options > Invert** after importing. DO NOT analyze from a jpeg, bmp, or png image; only grayscale TIFF or other full resolution, lossless file types with at least 16-bit depth are acceptable for analysis.*

2. Enter the **Product Number** (found on the Antigen Standard Card) into

the **Product** field.

- 3. Image Processing:** Align the plate overlay as follows:
  - a. To visualize bright or dim spots, optimize the display using **Image Options > Adjust Gamma** (does not affect the data).
  - b. Set the overlay: If using the **Auto-Set Plate Overlay** feature, this will occur automatically. Otherwise, go to **Overlay Options > Set Plate**.
  - c. Optimize overlay alignment: Go to **Overlay Options > Adjust plate** to pivot the overlay, **Adjust Well** and **Adjust Spot** to move individual wells and spots, then **Auto-Adjust Spots** to automatically snap each circle of the overlay to the nearest spot of the image beneath.
- 4. Well Assignment:** Label wells as samples, controls, standards, or negatives, and specify their dilution factors. Use **Templates** to quickly assign layouts that are repeated often, or export the layout as a .csv file.
- 5. Data Analysis:** Choose a **Curve Fit Option**, mask outliers, and select limits. The software will automatically compile customizable reports with tables, charts, and statistical information. Finally, copy or **Export** the data as needed.

Further tips for data analysis are available at [www.quansysbio.com/tech-tips](http://www.quansysbio.com/tech-tips). If you have further questions about running the assay, data analysis, troubleshooting, or our products or services, please contact us at 888-QUANSYS (782-6797) or at [INFO@QUANSYSBIO.COM](mailto:INFO@QUANSYSBIO.COM)

## PLATE WASHING METHODS

Before running the assay, select and become familiar with a plate washing method. If you have an automatic plate washer, use the automatic plate washer method described below. If you do not have an automatic plate washer, follow the instructions for the multichannel pipette method

### Preferred: Automatic Plate Washer Method

1. Use a program that will aspirate and dispense 300-400  $\mu\text{L}$  wash buffer.
2. Ensure that the plate washer has a program that will not scratch the bottom of the microplate and will prevent plate drying. This is critical to prevent damage to the capture antibody spots. The simplest method to avoid plate drying is to leave a small, uniform amount (1-3  $\mu\text{L}$ ) of wash in the well after the final aspiration, and add the next reagent to the plate as quickly as possible. For this reason, we do not recommend blotting the plate on a paper towel when using an automatic plate washer. Leaving a uniform amount of wash in the wells can be accomplished with an automated washer by slightly increasing the aspiration height of the washer head. For example:

Process	Distance	Steps on a Biotek ELX-405
Aspiration Height	3.810 mm	30
Aspiration Position	1.28 mm from center	-28
Dispense Height	15.24 mm	120
<i>no soak or shake cycles are needed</i>		

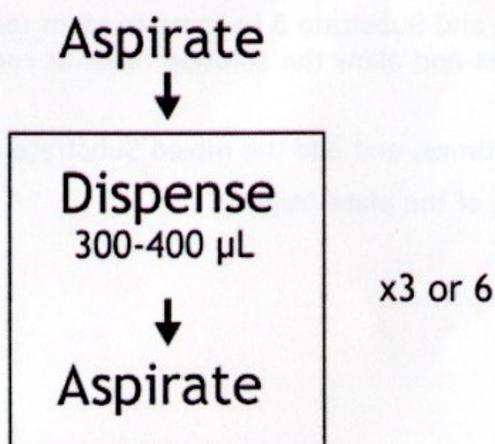
3. Connect the prepared wash buffer to your automatic plate washer.
4. Run 1-2 priming cycles to make sure that the wash buffer is running through the plate washer. When the buffer has run through the machine, the waste will be foamy.
5. To ensure that all pins are functioning, in a spare microtiter plate, dispense 100 $\mu\text{L}$  wash buffer and ensure that all pins dispense uniformly, then aspirate and ensure that all pins aspirate uniformly.

6. Prime the plate washer one time before the first wash step. When running the assay, perform the wash 3 or 6 times according to the protocol.

### Multichannel Pipette Method

1. Just prior to washing, pour the prepared wash buffer into a trough or tray.
2. After each incubation, flick the solution out of the plate over a waste container before starting the wash protocol.
3. Using a multichannel pipette, dispense 200-300  $\mu\text{L}$  of wash buffer into each of the wells used in the test.
4. Aggressively flick the wash buffer out over a waste container.
5. This washes the plate one time. When the assay procedure calls for three or six washes, repeat steps 3-4 accordingly.

### Wash Cycle Diagram



# Q-Plex™ ARRAY

Anti Poly-Histidine

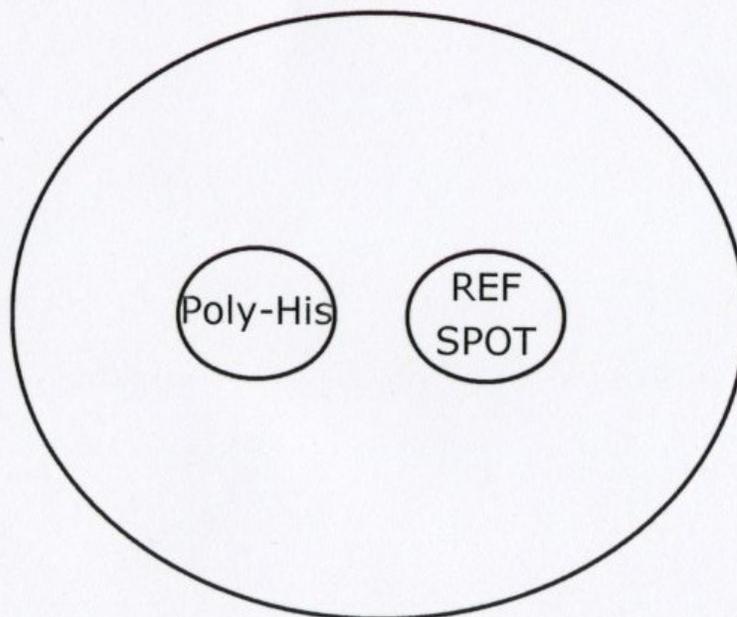
Software Product Code: **HISM140414US**

Competitor Lot Number: HISM140414

Reconstitution Volume: See Manual

101896GR

<u>Analyte</u>	<u>Units</u>
Anti Poly-Histidine	N/A
REF SPOT	N/A



**QUANSYS**   
BIOSCIENCES

V1.0  
365 North 600 West Logan, UT 84321  
Toll Free: 1-888-QUANSYS (782-6797)  
[www.quansysbio.com](http://www.quansysbio.com)

## Instructions for using Precise Protein Gels

### A. Preparing the Gel Cassette and Gel Tank

Important: Please see notes at the end of this section (Section A) concerning special instructions for using Bio-Rad® Mini PROTEAN Cell and NOVEX Tanks.

1. Dissolve one packet of BupH™ Tris-HEPES-SDS Running Buffer (Product No. 28398) in 500 ml of ultrapure water. This buffer volume of running buffer (500 ml) is sufficient for one electrophoresis unit. (See Buffer Recipes on page 12 for instructions for preparing a 10X stock of the required Tris-HEPES-SDS Running Buffer.)

**Note: Do not use Tris-glycine-SDS running buffer. This buffer formulation is not compatible with Precise Gels. Proteins will not migrate or resolve properly. Use only the required Tris-HEPES-SDS running buffer.**

2. Remove Precise Protein Gel from the pouch and insert into the gel running apparatus. Refer to the apparatus manufacturer's instructions.
3. Add sufficient volume of Tris-HEPES-SDS running buffer into the inner tank of the gel running apparatus to cover the sample wells by 5-7 mm.
4. Add the remaining volume of Tris-HEPES-SDS running buffer to the outer tank to ensure proper cooling. The buffer in the outer tank should be approximately level with the bottom of the sample wells.

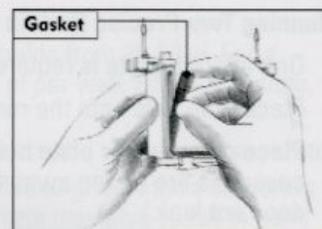
Note: For best resolution, the buffer in the outer tank must reach the bottom of the sample wells to keep the gels cool.

5. Using a transfer (pasteur) pipette, rinse the sample wells thoroughly with Tris-HEPES-SDS running buffer to remove air bubbles and to displace any storage buffer. The gel may be pre-electrophoresed for 5-10 minutes.

### Notes for using Bio-Rad Mini-PROTEAN Cell and NOVEX Tanks:

#### Using a Bio-Rad Mini-PROTEAN Cell

- To use a Bio-Rad Mini-PROTEAN Cell apparatus, remove the gasket from the inner frame (Figure 3), turn it around so the flat side is facing outwards and re-insert into the inner frame.



**Figure 3.** Removing the gasket from the cell.

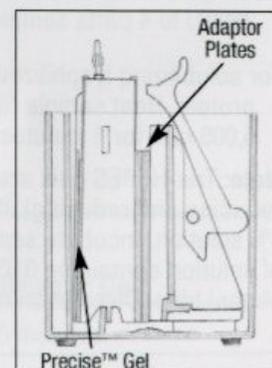
#### Using a NOVEX Gel Running Apparatus

Note: A tight seal must be formed between the gel cassette and the gasket of the running frame to prevent leakage. In the NOVEX Tanks, adaptor plates must be used to form a tight seal. Below are instructions for adaptor plate placements when running only one gel and when running two gels.

Precise Protein Gels are 8.5 cm high and are positioned lower in the Novex Tank than other brands of precast gels. However, the resolving portion of the gel is the same length as a Novex Gel.

#### Running One Precise Protein Gel in a NOVEX Tank

- Two adaptor plates are required when running just one gel.
- Place the gel onto the running frame.
- Place two adaptor plates onto the back of the running frame. (Ensure that the wells are facing away from the inner tank so that buffer does not leak.)
- Put the running frame into the tank.
- Using the wedge device, clamp the running frame and the gels into place (Figure 4). The running frame should be fixed tightly and the gels should not be able to move.



**Figure 4.** Adaptor plates placement when using one gel.

### Running Two Precise Protein Gels in a NOVEX Tank

- One adaptor plate is required when running two gels.
- Place both gels onto the running frame.
- Place one adaptor plate behind the back gel. (Ensure that the cassettes are facing away from the inner tank so that the buffer does not leak.)
- Place the running frame into the tank.
- Using the wedge device, clamp the running frame and the gels into place (Figure 5). The running frame should be fixed tightly and the gels should not be able to move.

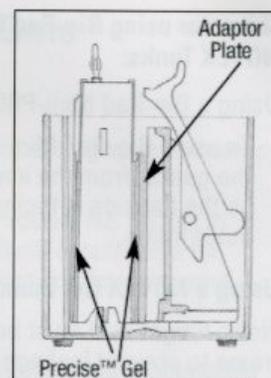


Figure 5. Adaptor plate placement when using two gels.

### B. SDS Sample Preparation

Add 1 part Lane Marker Reducing or Non-Reducing 5X Sample Buffer (Product No. 39000 or 39001) to 4 parts sample. Alternatively, use the sample buffer recipe on page 13.

For solubilizing lyophilized samples, mix 100  $\mu$ l of Tris-HCl SDS Sample Buffer (1X) per mg of protein. Heat sample for 3-5 minutes at approximately 100°C. Clarify by centrifugation at 6,000 rpm for 3 minutes and collect the supernatant.

**Note:** Tris-HEPES gels are compatible with Tris-Glycine SDS sample loading buffer (non-reducing and reducing). If the sample is thermally labile, add SDS to the sample as a 4% solution. Incubate sample for 1 hour at room temperature then add an equal volume of solution containing 0.05 M HEPES, 1 M sodium chloride, 10% glycerol and 0.05% Bromophenol blue. Dissolution may be helped by sonication.

### C. Sample Loading

Apply 5-50  $\mu\text{g}$  (total protein) per sample well. Each sample well holds from 25-50  $\mu\text{l}$ . For a sample with a total protein concentration of 10 mg/ml, apply 2-5  $\mu\text{l}$  per well. For best results, use pipette tips specifically designed for gel loading.

**Caution:** Inserting the pipette tip too far into the cassette may cause the cassette to separate.

**Note:** Optimal sample size must be established empirically. Overloading will cause smearing and distortion. Excessive loading of proteins with free carbohydrate may also result in band distortion or failure of the protein to penetrate into the gel (see Troubleshooting Section).

### D. Running Conditions

Connect the gel rig leads to the power supply (Figure 6) and electrophorese according to Table 1.

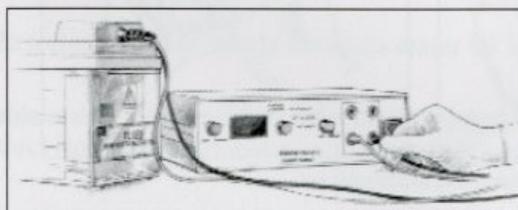


Figure 6. Gel tank and power supply.

Table 1. Electrophoresis conditions for Precise Gels.

Voltage	Approximate Current		Run Time per Gel*
	Start	Finish	
100-120 V	100-130 mA/gel	40-60 mA/gel	~45 minutes

\*Gel running time is dependent on the temperature in the laboratory. These run times are recommended at a laboratory temperature of 20°C.

### E. Removing a Gel from the Cassette

1. Once the run is finished, remove the gel from the gel tank according to the manufacturer's instructions.
2. To open the cassette, insert a coin in one of the slots on the side and twist (Figure 7a).
3. Pull the top plate of the cassette away from the bottom plate (Figure 7b). The two halves will snap apart completely, exposing the gel.
4. Loosen the gel at the bottom with water and remove.

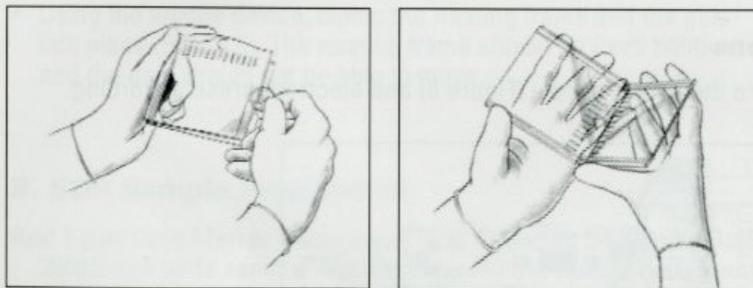


Figure 7a and 7b. Opening the cassette to expose the gel.

### Wet Blotting Protocol

1. Cool the transfer buffer to 4°C.
2. Equilibrate the gels in transfer buffer for 5 minutes.
3. Soak filter papers (8 × 10 cm) in transfer buffer.
4. Soak membrane(s) (8 × 10 cm) in transfer buffer (PVDF membranes must be wetted in methanol first and then equilibrated in aqueous solution).
5. Soak the Scotch-Brite™ Pads in transfer buffer.
6. Assemble the transfer sandwich as follows:
  - Cathode (---)
  - Scotch-Brite Pad
  - 2x filter paper
  - Gel
  - Transfer Membrane
  - 2x filter paper
  - Scotch-Brite Pad
  - Anode (+++)

**Note:** Blotter should be firmly packed. If two membranes are to be blotted, repeat the above transfer sandwich. If only one gel is to be blotted, fill the space with more filter paper and another Scotch-Brite Pad.

7. Pour the transfer buffer through the sandwich and place it into the apparatus. Fill the apparatus with transfer buffer.
8. Transfer at 40V for 90 minutes (maintain buffer temperature at  $\sim 4^{\circ}\text{C}$ ).
9. Gently remove gel from sandwich and rinse with transfer buffer.
10. Use a cotton swab to remove any adhering gel from the membrane.

## Buffer Recipes

Use high-purity reagents and high-purity water when making buffers.

### Tris-HEPES-SDS Running Buffer (10X)

Tris Base (MW = 121) .....121 g

HEPES (free acid MW = 238) .....238 g

SDS (MW = 288) .....10 g

Add ultrapure water to 1 L

- Before use dilute 10-fold with water. The pH of the 1X buffer should be  $\sim 8.0$ ; do not adjust pH.
- Final composition of the 1X buffer is 100 mM Tris, 100 mM HEPES, 0.1% ( $\sim 3$  mM) SDS, pH 8.0.
- Do not use Tris-glycine-SDS running buffer with Precise Protein Gels.

## Appendix C: Sequence information

### *FLYS3* sequence



ID Construct      **13ABW7UC**      **Map**  
 Customer          **Utah State University - Randy Lewis**  
 Name of the gene   **FIYS**  
 optimized for      Escherichia coli

02.09.2013 09:25:42

```

                                NdeI
                                HindIII
                                AgeI
CACTATAGGGCGAATTGGCGGAAGGCCGTCAAGGCCGCATAAGCTTCATATGGGATCAAC
1  -----+-----+-----+-----+-----+-----+-----+
GTGATATCCCGCTTAACCGCCTTCCGGCAGTTCCGGCGTATTCGAAGTATAACCCTAGTTG
                                K L H M G S T

                                XmaI
                                SmaI
CGGTCCCAGGGGTCCGGGTGGTTATGGTCCTGGTGGTAGTGGTCCAGGTGGCTATGGACC
61 -----+-----+-----+-----+-----+-----+
GCCAGGGCCCCCAGGCCACCAATACCAGGACCACCATCACCAGGTCCACCGATACCTGG
G P G G P G G Y G P G G S G P G G Y G P

GGGTGGTTCGGTCCAGGCGGTTATGGCCCTGGCGGTTCCAGGTCCGGGTGGATACGGACC
121 -----+-----+-----+-----+-----+-----+
CCCACCAAGGCCAGGTCCGCCAATACCGGGACCGCCAAGTCCAGGCCACCTATGCCTGG
G G S G P G G Y G P G G S G P G G Y G P

                                PvuII
                                PstI
                                BspMI
AGGTGGCAGCGGTCCGAGTGGTCCGGTAGTGCAGCAGCAGCAGCCGAGCTGCAGGTCC
181 -----+-----+-----+-----+-----+-----+
TCCACCGTCGCCAGGCTCACCAGGCCATCACGTCGTCGTCGTCGGCGTCGACGTCCAGG
G G S G P S G P G S A A A A A A A A G P

```

AGGGGGATATGGTCCAGGGGGTAGCGGACCTGGCGGTTATGGGCCAGGTGGCTCTGGCCC  
 241 -----+-----+-----+-----+-----+-----+-----+  
 TCCCCCTATACCAGGTCCCCCATCGCCTGGACCGCCAATACCCGGTCCACCGAGACCGGG  
  G  G  Y  G  P  G  G  S  G  P  G  G  Y  G  P  G  G  S  G  P  

TGGTGGATATGGCCCAGGCGGAAGTGGCCAGGTGGTTACGGACCTGGGGGATCAGGACC  
 301 -----+-----+-----+-----+-----+-----+-----+  
 ACCACCTATACCGGGTCCGCCTTACCCGGGTCCACCAATGCCTGGACCCCTAGTCCTGG  
  G  G  Y  G  P  G  G  S  G  P  G  G  Y  G  P  G  G  S  G  P  

AGGCGGTTACGGTCCGGGTGGCTCAGGTCCTAGCGGTCCGGGTTCCAGCCGACGCGGCAGC  
 361 -----+-----+-----+-----+-----+-----+-----+  
 TCCGCCAATGCCAGGCCACCGAGTCCAGGATCGCCAGGCCAAGTCGGCGTCGCCGTCG  
  G  G  Y  G  P  G  G  S  G  P  S  G  P  G  S  A  A  A  A  A  

AGCAGCGGCAGGACCGGGTGGCTATGGGCCAGGGGGTTCGGGACCTGGTGGTTATGGACC  
 421 -----+-----+-----+-----+-----+-----+-----+  
 TCGTCGCCGTCCTGGCCCACCGATAACCCGGTCCCCAAGCCCTGGACCACCAATACCTGG  
  A  A  A  G  P  G  G  Y  G  P  G  G  S  G  P  G  G  Y  G  P  

TGGCGGAAGCGGTCTGGGGGTTACGGTCCAGGTGGAAGTGGACCGTCAGGTCCAGGTAG  
 481 -----+-----+-----+-----+-----+-----+-----+  
 ACCGCCTTCGCCAGGACCCCCAATGCCAGGTCCACCTTACCTGGCAGTCCAGGTCCATC  
  G  G  S  G  P  G  G  Y  G  P  G  G  S  G  P  S  G  P  G  S  

*PvuII*      *PstI*          *BspMI*  
 CGCAGCTGCCGCTGCAGCCGACGAGGTCCAGGTGGGTACGGTCTGGTGGTTCTGGACC  
 541 -----+-----+-----+-----+-----+-----+-----+  
 GCGTCGACGGCGACGTCCGGCGTCGTCCAGGTCCACCCATGCCAGGACCACCAAGACCTGG  
  A  A  A  A  A  A  A  A  G  P  G  G  Y  G  P  G  G  S  G  P  

GGGTGGGTATGGTCCGGGTGGAAGCGGACCGGGTGGATATGGCCCTGGGGGATCTGGTCC  
 601 -----+-----+-----+-----+-----+-----+-----+  
 CCCACCCATAACCAGGCCACCTTCGCCTGGCCCACCTATACCGGGACCCCTAGACCAGG  
  G  G  Y  G  P  G  G  S  G  P  G  G  Y  G  P  G  G  S  G  P

TGGCGGATATGGACCTGGTGGGTGCGGACCAGGGGGATACGGACCGGGTGGTAGTGGCCC  
 661 -----+-----+-----+-----+-----+-----+-----+  
 ACCGCCTATACCTGGACCACCCAGCCCTGGTCCCCCTATGCCTGGCCCACCATCACCGGG  
G G Y G P G G S G P G G Y G P G G S G P

*PvuII*

AGGCGGATACGGTCCTGGCGGTAGCGGTCCATCAGGTCCGGGATCTGCTGCTGCTGCGGC  
 721 -----+-----+-----+-----+-----+-----+-----+  
 TCCGCCTATGCCAGGACCGCCATCGCCAGGTAGTCCAGGCCCTAGACGACGACGACGCCG  
G G Y G P G G S G P S G P G S A A A A A

*PstI*

AGCTGCAGCCGGACCCAGGGGGTTATGGACCAGGTGGTTCAGGACCAGGTGGCTACGGTCC  
 781 -----+-----+-----+-----+-----+-----+-----+  
 TCGACGTCGGCCTGGTCCCCAATACCTGGTCCACCAAGTCCTGGTCCACCGATGCCAGG  
A A A G P G G Y G P G G S G P G G Y G P

AGGCGGTAGTGGGCCTGGGGGATATGGTCCGGGTGGCTCTGGGCCTGGCGGTTACGGACC  
 841 -----+-----+-----+-----+-----+-----+-----+  
 TCCGCCATCACCCGGACCCCTATAACCAGGCCACCGAGACCCGGACCGCCAATGCCTGG  
G G S G P G G Y G P G G S G P G G Y G P

TGGCGGTAGTGGACCGGGTGGTTATGGCCCAGGTGGCTCCGGTCCGGGTGGGTATGGGCC  
 901 -----+-----+-----+-----+-----+-----+-----+  
 ACCGCCATCACCTGGCCCACCAATACCGGGTCCACCGAGGCCAGGCCACCCATAACCCGG  
G G S G P G G Y G P G G S G P G G Y G P

AGGTGGATCTGGGCCAGGCGGTTATGGTCCAGGGGGATCGGGTCCAGGTGGATATGGCCC  
 961 -----+-----+-----+-----+-----+-----+-----+  
 TCCACCTAGACCCGGTCCGCCAATACCGGTCCCCCTAGCCCAGGTCCACCTATAACCCGG  
G G S G P G G Y G P G G S G P G G Y G P

*PvuII**PstI**BspEI*

AGGTGGTTCAGGTCCATCTGGTCCGGGTCCGCGAGCTGCAGCCGCAGCCGCAGCTTCCGG  
 1021 -----+-----+-----+-----+-----+-----+  
 TCCACCAAGTCCAGGTAGACCAGGCCCAAGGCGTCGACGTCGGCGTCGGCGTCGAAGGCC  
  G  G  S  G  P  S  G  P  G  S  A  A  A  A  A  A  A  A  S  G  

*ApaI  EcoRV  XhoI  BamHI*

AGGGCCCGATATCCTCGAGGGATCCCTGGGCCTCATGGGCCTTCCGCTCACTGCCCGCTT  
 1081 -----+-----+-----+-----+-----+-----+  
 TCCCGGGCTATAGGAGCTCCCTAGGGACCCGGAGTACCCGGAAGGCGAGTGACGGGCGAA  
  G  P  D  I  L  E  G  S  

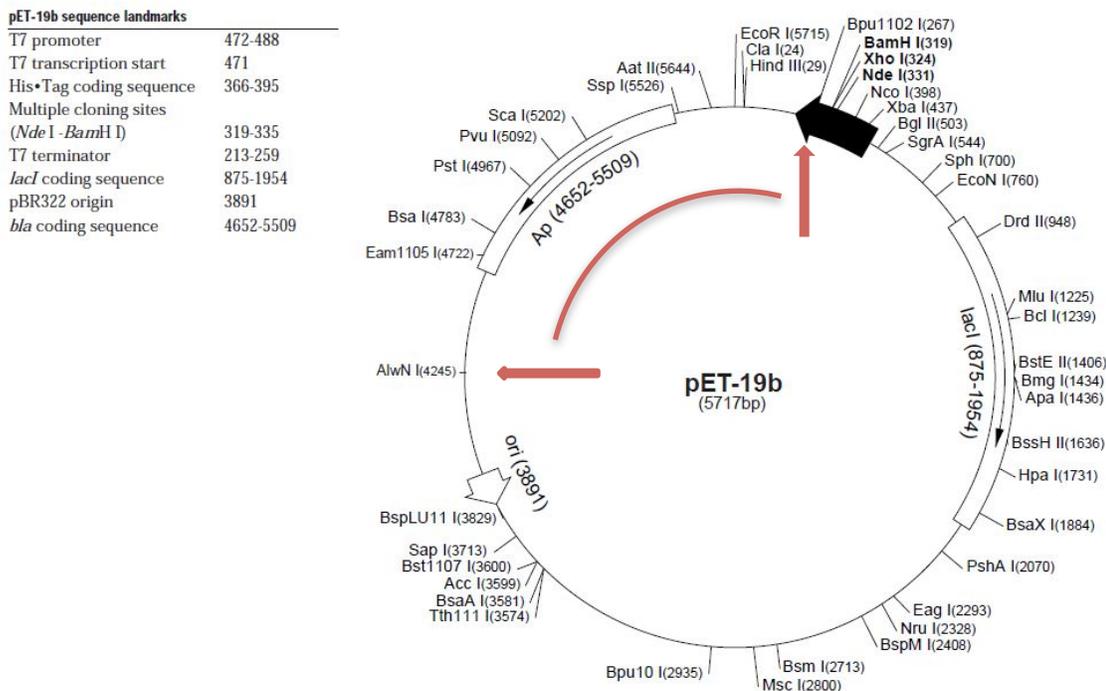
TCCAG

1141 -----

AGGTC

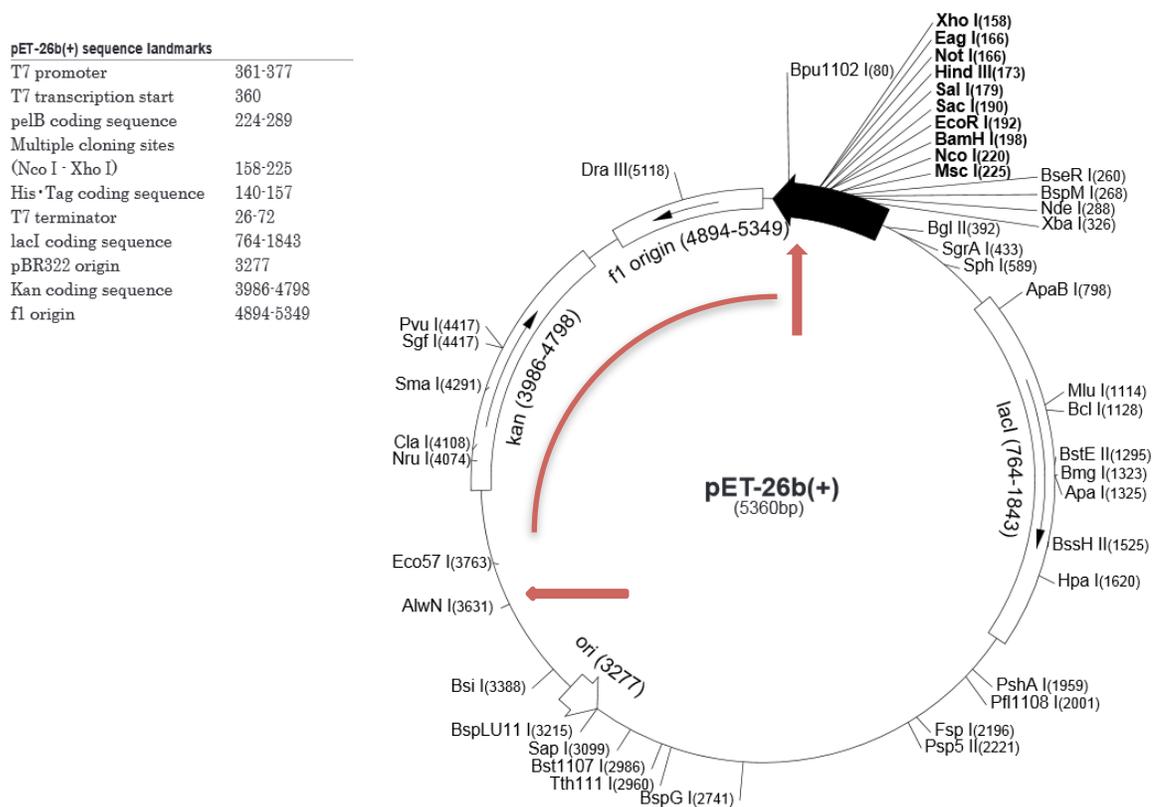
## Plasmid Modification

The pET19k-SX FIYS3 plasmid was used in this study. The pET19k-SX FIYS3 plasmid is a modified form of the pET-19b plasmid (See Fig 18).

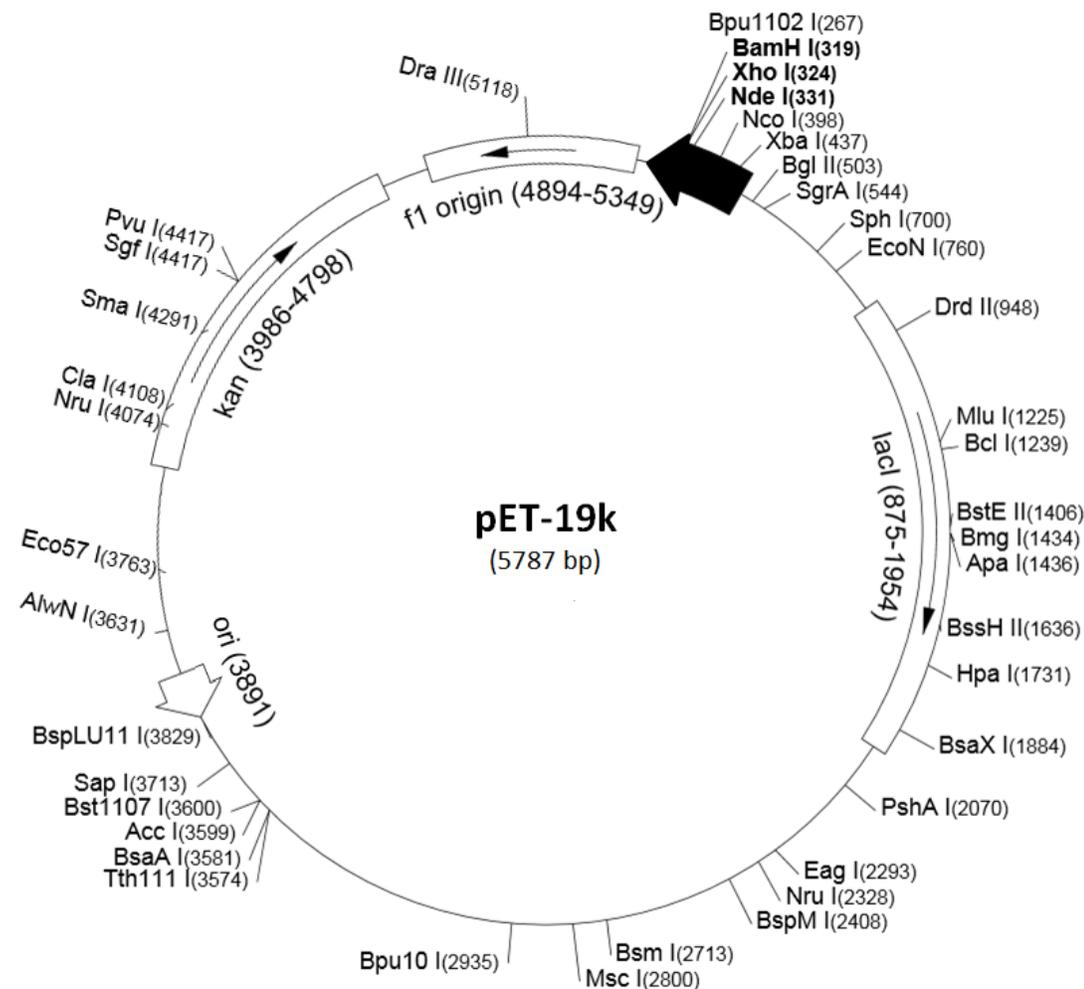


**Figure 18. Vector map of pET-19b.**

The modifications are as follows. The section AlwN I to Bpu1102 I was removed from pET19b, and the remaining large fragment was purified. The fragment AlwN I to Bpu1102 I from the plasmid pET-26b (see Fig 19) was then ligated into the large purified fragment, creating vector pET-19k (see Fig 20).



**Figure 19. Vector map of pET-26b. Obtained from Dr. Randy Lewis.**



**Figure 20. Vector map of pET-19k. Obtained from Dr. Randy Lewis.**

Bgl II to Sph I was then replaced with SHMT (Serine hydroxymethyltransferase) and GlyT, ProL, and ProM (tRNAs for glycine and two prolines). To reduce the size of the plasmid, a non-critical region between PshA I (blunt-ended GACNN/NNGTC) at bp 2070 of the original pET19b sequence and Hpy166 II (blunt-ended, GTN/NAC) at bp 3599 of the original pET19b sequence was excised. The resulting vector is known as pET19-SX, as shown in Fig 21. An insert containing 3 multiples of a 1000 bp fragment of flagelliform silk known as FIYS3 was inserted between NdeI and BamHI, creating the pET19k-SX-FIYS3 vector.

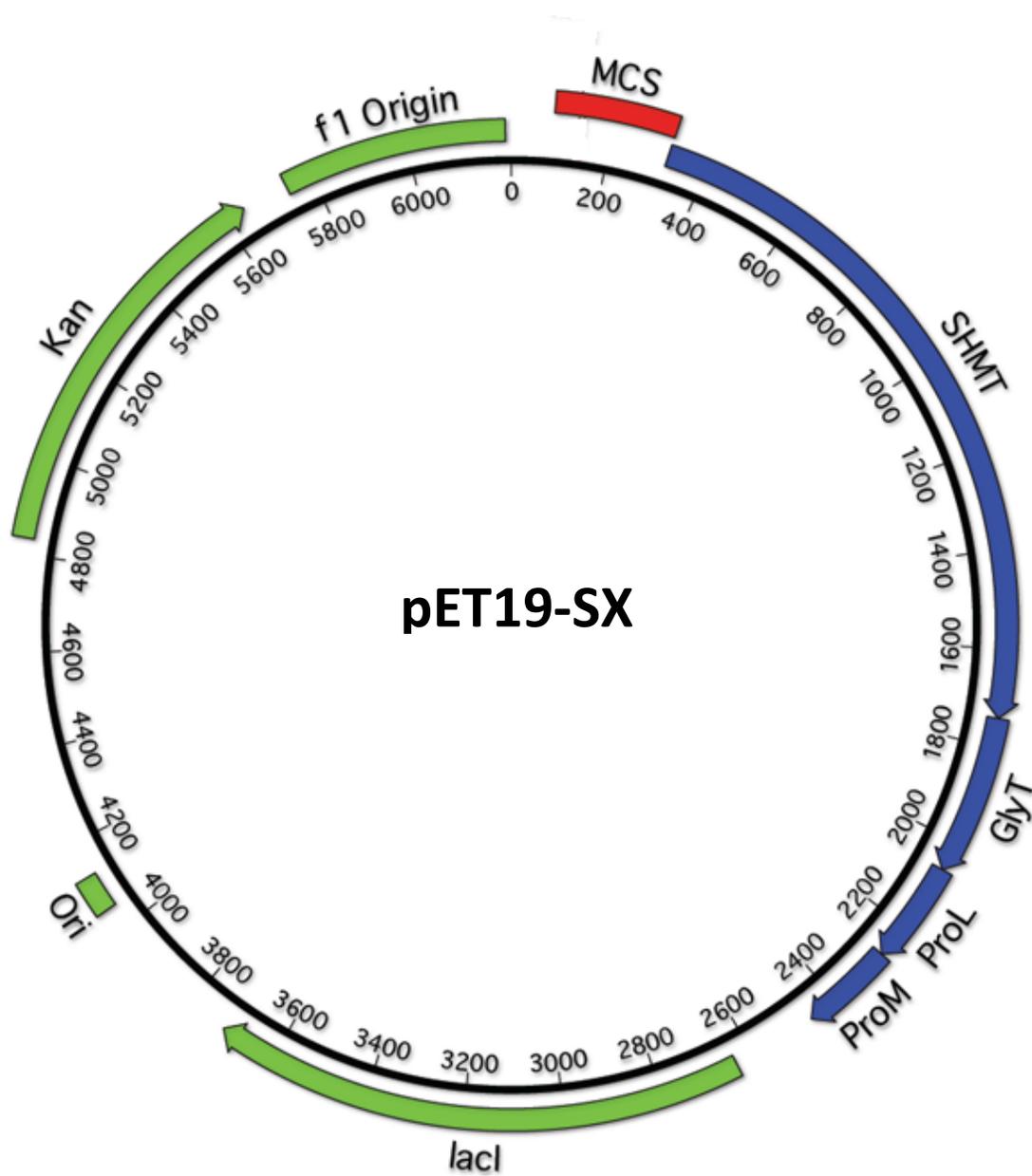


Figure 21. Vector map of pET19-SX. Obtained from Dr. Randy Lewis.

*Plasmid Sequence*

The amino acid sequence for pET19k-SX (6209 bp), obtained from Dr. Randy Lewis, is as follows:

ATCCGGATATAGTTCCTCCTTTTCAGCAAAAAACCCCTCAAGACCCGTTTAGAGGCCCAAGGG  
 GTTATGCTAGTTATTGCTCAGCGGTGGCAGCAGCCAACCTCAGCTTCCTTTTCGGGCTTTGTTAGC  
 AGCCGGATCCTCGAGCATATGCTTGTCTGTCGTCGATATGGCCGCTGCTGTGATGATGATG  
 ATGATGATGATGATGATGGCCATGGTATATCTCCTTCTTAAAGTTAAACAAAATTATTCTAG  
 AGGGGAATTGTTATCCGCTCACAATCCCCTATAGTGAGTCGTATTAATTTTCGCGGGATCGAG  
 ATCTGACCTGTTATCGACAATGATTCGGTTATACTGTTCGCCGTTGTCCAACAGGACCCGCTA  
 TAAAGGCCAAAAATTTTATTGTTAGCTGAGTCAGGAGATGCGGATGTTAAAGCGTGAAATGA  
 ACATTGCCGATTATGATGCCGAACTGTGGCAGGCTATGGAGCAGGAAAAAGTACGTCAGGAA  
 GAGCACATCGAACTGATCGCCTCCGAAAACCTACACCAGCCCGCGCTAATGCAGGCGCAGGG  
 TTCTCAGCTGACCAACAAATATGCTGAAGGTTATCCGGGCAAACGCTACTACGGCGGTTGCGA  
 GTATGTTGATATCGTTGAACAACCTGGCGATCGATCGTGCGAAAGAAGTTCGGCGCTGACTA  
 CGCTAACGTCCAGCCGCACTCCGGTCCCAGGCTAACTTTGCGGTCTACACCAGCGCTGCTGGA  
 ACCAGGTGATACCGTTCTGGGTATGAACCTGGCGCATGGCGGTCACCTGACTCACGGTTCTCC  
 GGTTAACTTCTCCGGTAAACTGTACAACATCGTTCTTACGGTATCGATGCTACCGGTCATATC  
 GACTACGCCGATCTGGAAAAACAAGCCAAAGAACAAGCCGAAAATGATTATCGGTGGTTT  
 CTCTGCATATTCGCGCGTGGTGGACTGGGCGAAAATGCGTGAAATCGCTGACAGCATCGGTGC  
 TTACCTGTTGCTGATATGGCGCACGTTGCGGGCCTGTTGCTGCTGGCGTCTACCCGAACCCG  
 GTTCTCATGCTCACGTTGTTACTACCACCACTACAAAACCCTGGCGGGTCCGCGCGGCGGC  
 CTGATCCTGGCGAAAGGTGGTAGCGAAGAGCTGTACAAAAAAGTAACTCTGCCGTTTTCCCT  
 GGTGGTCAGGGCGGTCCGTTGATGCACGTAATCGCCGGTAAAGCGGTTGCTCTGAAAGAAGC  
 GATGGAGCCTGAGTTCAAACTTACCAGCAGCAGGTCGCGAAAAACGCTAAAGCGATGGTAG  
 AAGTGTTCCTCGAGCGCGGCTACAAAGTGGTTCCGGCGGCACTGATAACCACCTGTTCTCGG  
 TTGATCTGGTTGATAAAAAACCTGACCGGTAAAGAAGCAGACGCCGCTCTGGGCCGTGCTAAC  
 ATCACCGTCAACAAAAACAGCGTACCGAACGATCCGAAGAGCCCGTTTGTGACCTCCGGTATT  
 CGCGTGGGTACTCCGGCAATTACCGCTCGCGGCTTCAAAGAAGCAGAAGCGAAAGAAGTGGC  
 TGGCTGGATGTGTGACGTGCTGGACAGCATCAATGATGAAGCCGTTATCGAGCGCATCAAAG  
 GTAAATGCTCGACATCTGCGCACGTTACCCGGTTTACGCATAAGCAGAAACGGTGATTTGCTG  
 ACAATGTGCTCGATTGTTTCAATGTTGGATGCGGCTGAAACACGTCGACCGTAGCCGAGACGATA  
 AGTTCGCTTACCGGCTCGAATGAAGAGAGCTTCTCTCGATATTCAGTGCAGAATGAAAATCAG  
 GTAGCCGAGTTCAGGATGCGGGCATCGTATAATGGCTATTACCTCAGCCTTCCAAGCTGATG  
 ATGCGGGTTCGATTCCCGCTGCCCGCTCCAAGATGTGCTGATATAGCTCAGTTGGTAGAGCGC  
 ACCCTTGGTAAGGGTGAGGTTCGGCAGTTCGAATCTGCCTATCAGCACCACTTCTTTTCTCCTCC  
 CTGTTTTTTCTTCTGTTTATTGCATTCAACAAGTCGGGCATGTTGCAAGCTTCTTGCAATCGGT  
 GTGGAAAACGGTAGTATTAGCAGCCACGAGTCGGCACGTAGCGCAGCCTGGTAGCGCACCGT  
 CATGGGGTGTGCGGGGTCGGAGGTTCAAATCCTCTCGTGCCGACCAAAAATCCCAAGAAAA  
 ACCAACCCCTTACGGTTGGTTTTTTTATATCTGCAATTAATTCGATAAACAGACCGTGACACATC  
 ACGAATTCCTCGCACCACTTTAAAGAATTGAACTAAAAATTCAAAAAGCAGTATTTTCGGC  
 GAGTAGCGCAGCTTGGTAGCGCAACTGGTTTGGGACCAGTGGGTCGGAGGTTTCAATCCTCTC  
 TCGCCGACCAATTTTGAACCCCGCTTCGGCGGGGTTTTTTGTTTTCTGTGCATTTTCGTCATTT  
 CCGCATGCACCATTCTTTCGCGCGGCGGTGCTCAACGGCCTCAACCTACTACTGGGCTGCTTC  
 CTAATGCAGGAGTCGCATAAGGGAGAGCGTCGAGATCCCGGACACCATCGAATGGCGCAAAA  
 CCTTTCGCGGTATGGCATGATAGCGCCGGAAGAGAGTCAATTCAGGGTGGTGAATGTGAAA  
 CCAGTAACGTTATACGATGTGCGAGAGTATGCCGGTGTCTTATCAGACCGTTTCCCGCGTG  
 GTGAACCAGGCCAGCCACGTTTCTGCGAAAAACGCGGGAAAAAGTGGAAAGCGGCGATGGCGG  
 AGCTGAATTACATTCCTAACCGCGTGGCACAACAACCTGGCGGGCAAACAGTCGTTGCTGATTG  
 GCGTTGCCACCTCCAGTCTGGCCCTGCACGCGCCGTCGCAAAATTGTCGCGGCGATTAAATCTC  
 GCGCCGATCAACTGGGTGCCAGCGTGGTGGTGTGATGGTAGAACGAAGCGGCGTCAAGCC  
 TGTAAGCGGCGGTGCACAATCTTCTCGCGCAACGCGTCAGTGGGCTGATCATTAACTATCCG  
 CTGGATGACCAGGATGCCATTGCTGTGGAAGCTGCCTGCACTAATGTTCCGGCGTTATTTCTG

ATGTCTCTGACCAGACACCCATCAACAGTATTATTTTCTCCCATGAAGACGGTACGCGACTGG  
GCGTGGAGCATCTGGTCGCATTGGGTCACCAGCAAATCGCGCTGTTAGCGGGCCCATTAAGTT  
CTGTCTCGGCGCGTCTGCGTCTGGCTGGCTGGCATAAATATCTCACTCGCAATCAAATTCAGC  
CGATAGCGGAACGGGAAGGCGACTGGAGTGCCATGTCCGGTTTTCAACAAACCATGCAAATG  
CTGAATGAGGGCATCGTTCCTACTGCGATGCTGGTTGCCAACGATCAGATGGCGCTGGGCGCA  
ATGCGCGCCATTACCGAGTCCGGGCTGCGCGTTGGTGCGGATATCTCGGTAGTGGGATACGAC  
GATACCGAAGACAGCTCATGTTATATCCCGCCGTTAACACCATCAAACAGGATTTTCGCCTG  
CTGGGGCAAACCAGCGTGGACCGCTTGTGCAACTCTCTCAGGGCCAGGCGGTGAAGGGCAA  
TCAGCTGTTGCCCGTCTCACTGGTGAAGAAAGAAAACCACCCTGGCGCCCAATACGCAAACCGC  
CTCTCCCCGCGCGTTGGCCGATTCATTAATGCAGCTGGCACGACAGGTTTTCCCGACTGGAAAG  
CGGGCAGTGAGCGCAACGCAATTAATGTAAGTTAGCTCACTCATTAGGCACCGGGATCTCGAC  
CGATGCCCTTGAGAGCCTTCAACCCAGTCAGCTCCTTCCGGTGGGCGCGGGGCATGACTAATA  
CTGGCTTAACTATGCGGCATCAGAGCAGATTGACTGAGAGTGCACCATATATGCGGTGTGAA  
ATACCGCACAGATGCGTAAGGAGAAAATACCGCATCAGGCGCTCTTCCGCTTCTCGCTCACT  
GACTCGCTGCGCTCGGTGCTTCCGGCTGCGGCGAGCGGTATCAGCTCACTCAAAGGCGGTAATA  
CGGTTATCCACAGAATCAGGGGATAACGACAGAAAGAACATGTGAGCAAAGGCCAGCAAA  
AGGCCAGGAACCGTAAAAAGGCCGCGTTGCTGGCGTTTTTCCATAGGCTCCGCCCCCTGACG  
AGCATCACAAAATCGACGCTCAAGTCAGAGGTGGCGAAACCCGACAGGACTATAAAGATAC  
CAGGCGTTTTCCCCCTGGAAGCTCCCTCGTGCCTCTCCTGTTCCGACCCTGCCGCTTACCGGAT  
ACCTGTCCGCTTTTCTCCCTTCGGGAAGCGTGGCGCTTTTCTCATAGCTCACGCTGTAGGTATCT  
CAGTTCGGTGTAGGTGCTTCGCTCCAAGCTGGGCTGTGTGCACGAACCCCCCGTTACGCCCGA  
CCGCTGCGCCTTATCCGGTAACTATCGTCTTGAGTCCAACCCGGTAAGACACGACTTATCGCC  
ACTGGCAGCAGCCACTGGTAACAGGATTAGCAGAGCGAGGTATGTAGGCGGTGCTACAGAGT  
TCTTGAAGTGGTGGCCTAACTACGGCTACACTAGAAGGACAGTATTTGGTATCTGCGCTCTGC  
TGAAGCCAGTTACCTTCGGAAAAAGAGTTGGTAGCTCTTGATCCGGCAAACAAACCACCGCTG  
GTAGCGGTGGTTTTTTTTGTTTGAAGCAGCAGATTACGCGCAGAAAAAAGGATCTCAAGAA  
GATCCTTTGATCTTTTCTACGGGGTCTGACGCTCAGTGGAACGAAAACCTCACGTTAAGGGATT  
TTGGTCATGAACAATAAACTGTCTGCTTACATAAACAGTAATACAAGGGGTGTTATGAGCCA  
TATTCAACGGGAAACGTCTTGCTCTAGGCCGCGATTAAATTCCAACATGGATGCTGATTTATA  
TGGGTATAAATGGGCTCGCGATAATGTCCGGCAATCAGGTGCGACAATCTATCGATTGTATGG  
GAAGCCCGATGCGCCAGAGTTGTTTCTGAAACATGGCAAAGGTAGCGTTGCCAATGATGTTAC  
AGATGAGATGGTCAGACTAACTGGCTGACGGAATTTATGCCTCTTCCGACCATCAAGCATT  
TATCCGTAATCCTGATGATGCATGGTACTCACCCTGCGATCCCCGGGAAAACAGCATTCCA  
GGTATTAGAAGAATATCCTGATTCAGGTGAAAATATTGTTGATGCGCTGGCAGTGTTCCTGCG  
CCGTTGCAATTCGATTCCTGTTTGAATTTGCTCTTTAACAGCGATCGCGTATTTTCGTCTCGCTC  
AGGCGCAATCACGAATGAATAACGGTTTTGGTTGATGCGAGTGAATTTGATGACGAGCGTAATG  
GCTGGCCTGTTGAACAAGTCTGGAAAGAAATGCATAAACTTTTGCCATTCTCACCGGATTCAG  
TCGTCACTCATGGTGATTTCTCACTTGATAACCTTATTTTTGACGAGGGGAAATTAATAGGTTG  
TATTGATGTTGGACGAGTCGGAATCGCAGACCGATAACCAGGATCTTGCATCCTATGGAACG  
CCTCGTGAGTTTTCTCCTTATTACAGAAACGGCTTTTTCAAAAATATGGTATTGATAATCCT  
GATATGAATAAATTCAGTTTTCAATTTGATGCTCGATGAGTTTTTCTAAGAATTAATTCATGAGC  
GGATACATATTTGAATGTATTTAGAAAAATAAACAAATAGGGGTTCCGCGCACATTTCCCCGA  
AAAGTGCCACCTGAAATTGTAACGTTAATATTTGTTAAAATTCGCGTTAAATTTTTGTTAAA  
TCAGCTCATTTTTTAACCAATAGGCCGAAATCGGCAAAATCCCTTATAAATCAAAGAATAGA  
CCGAGATAGGGTTGAGTGTGTTCCAGTTTGGAAACAAGAGTCCACTATTAAGAACGTGGACT  
CCAACGTCAAAGGGCGAAAAACCGTCTATCAGGGCGATGGCCCACTACGTGAACCATCACCC  
TAATCAAGTTTTTTGGGGTTCGAGGTGCCGTAAGCACTAAATCGGAACCCTAAAGGGAGCCCC  
CGATTTAGAGCTTGACGGGGAAAGCCGGCGAACGTGGCGAGAAAGGAAGGGAAGAAAGCGA  
AAGGAGCGGGCGCTAGGGCGCTGGCAAGTGTAGCGGTACGCTGCGCGTAACCACCACACC  
GCCGCGTTAATGCGCCGCTACAGGGCGCGTCCCATTCGCCA