Identifying *in silico* knockouts that enhance limonene yield through dynamic modelling of limonene-pathway metabolites

Jasmeet Kaur Khanijou*[a, e], Yan Ting Hee*[b, e], Kumar Selvarajoo[b, c, d, e, #]

[a] Singapore Institute of Food and Biotechnology Innovation (SIFBI), Agency for Science, Technology and Research (A*STAR), Singapore 138673, Republic of Singapore

[b] Bioinformatics Institute (BII), Agency for Science, Technology and Research (A*STAR), Singapore 138671, Republic of Singapore

[c] Synthetic Biology for Clinical and Technological Innovation (SynCTI), National University of Singapore (NUS), Singapore 117456, Republic of Singapore

[d] School of Biological Sciences, Nanyang Technological University (NTU), Singapore 637551, Republic of Singapore

[e] CNRS@CREATE, 1 Create Way, #08-01 Create Tower, Singapore 138602

*These authors contributed equally to the work.

#correspondence: kumar_selvarajoo@bii.a-star.edu.sg

Abstract

Living cells display dynamic and complex behaviors. To understand their response and to infer novel insights not possible with traditional reductionist approaches, over the last few decades various computational modelling methodologies have been developed. In this chapter, we focus on modelling the dynamic metabolic response, using linear and non-linear ordinary differential equations, of an engineered *Escherichia coli* MG1655 strain with plasmid pJBEI-6409 that produces limonene. We show the systems biology steps involved from collecting time-series data of living cells, to dynamic model creation and fitting the model with experimental responses using COPASI software.

Key words: dynamic metabolic modelling, COPASI, parameter estimation, limonene, time-series data

1. Introduction

The issue of food security has been brought into light in recent years, primarily driven by climate change affecting crop yields across the world. The recent coronavirus disease 2019

(COVID-19) pandemic, rising fuel prices and various political strife between countries affecting supply chains magnified the importance of food security especially in land-scarce, import-dependent nations ***(1, 2)***. The environmental impact of current agricultural methods, including destruction of natural habitats and emission of greenhouse gases, has become increasingly urgent to address as the Earth's global temperatures inches towards the 1.5 ^{o}C warming limit. In view of the above mentioned, but not limited to, concerns, many countries have been looking to diversify their food sources, one of which is to investigate alternative and cultured food possibilities. These include plant-based meat, lab-grown meat and using microbes to as biofactories to produce compounds of interest ***(3)***.

(S)-Limonene, hereafter referred to as limonene, is a monoterpene compound with a sweet citrusy fragrance. Holding the generally recognized as safe (GRAS) status issued by the US Food and Drug Administration, there has been a growing application for it, besides its current use as a food, flavoring, and fragrance additive ***(4)***. Traditionally, limonene is produced from waste orange peel with current market price ranging between $7.92 to $20.15 per kilogram ***(5)***. However, this approach can be limited both by crop yields and the difficulty of citrus rind recycling ***(4, 5)***. To address these issues, several groups have engineered *E. coli* to increase limonene and other terpenes yields ***(6-11)***, and consequently to decrease their market price.

The precursor to limonene, geranyl diphosphate (GPP), is produced from the condensation of isopentenyl pyrophosphate (IPP) and its isomer, dimethylallyl pyrophosphate (DMAPP). IPP/DMAPP can be formed from two pathways – the deoxyxyluose phosphate (DXP) and the mevalonate (MVA) pathways (Figure 1). While the DXP pathway occurs in *E. coli* natively, it only results in the production of small amounts of terpenes. Engineering of the DXP pathway for higher yields of terpenes were met with limitations possibly due to physiological pathway regulations in *E. coli* ***(11)***. Therefore, the MVA pathway was engineered to be expressed alongside instead ***(6, 9, 11)*** (Figure 1). Building on these past works, there have been commendable efforts to further increase limonene yields from *E. coli*, ranging from transcriptional tuning ***(12)*** to genetically engineering for better ribosome binding sites, promoter optimization and utilization of plant, instead of native, ispA gene ***(13)***. However, these methods primarily target the MVA pathway elements alone and do not consider other relevant metabolic pathways, such as upstream glycolysis from a carbon source. Existence of bottlenecks and loss of flux are equally important ***(14)*** for consideration to maximize limonene yields.

Analyses of metabolic bottlenecks and flux distribution can be performed by several computational modeling approaches ***(14)***. Kinetic models are dynamic models that are based on linear and non-linear differential equations governed by the underlying (enzyme) kinetics for each reaction in a pathway. They are popularly used to describe dynamic biological processes, ranging from metabolic ***(15)***, to protein signaling ***(16, 17)***, to gene regulatory networks ***(18-20)***, for systems biology research. For instance, dynamic modelling of TRAIL signaling using time-series data revealed PKC as a target to overcome TRAIL resistance in cancer ***(21)***. The same method also helped identify RIP1 as a target to suppress but not abolish proinflammatory signaling ***(22)***. These examples, thus, highlight the huge potential of dynamic models built from time-series experimental data to uncover important insights in complex biological networks.

In the context of metabolic network modelling, the rates of change of metabolites in the network are represented by deterministic ordinary differential equations (ODEs). The rate of each reaction step in the network is defined by a rate law that describes the enzyme’s kinetics, for instance the Michaelis-Menten equation (or mass-action kinetics for protein signaling reactions). Defining the ODE of metabolite *A* in terms of its associated rate laws, we obtain the generalized equation [1]:

$$\frac{dA}{dt} = \sum_{i} s_i \cdot v_i \qquad [1]$$

where s_i represents the stoichiometric coefficient of species *A* in reaction i and v_i denotes the rate law of reaction i. Reactions that produce and consume *A* will have a positive and negative stoichiometric coefficient respectively. All the species in the network are thus represented by equation [1]. An example is shown in Figure 2. From the network topology above (Figure 1), BPG is produced and consumed by the GDH and PGK reactions respectively, with a stoichiometric coefficient of 1 in both. These are thus represented by the '+' and '-' signs in its ODE, respectively. Michaelis-Menten kinetics were used for both GDH and PGK.

Here, we describe the experimental methods to collect time-series data, forming the network topology and kinetic model construction using COmplex PAthway SImulator (COPASI) ***(23)***. Finally, we will also discuss how the time-series data is fed into the model for parameter estimations, further model tuning and model validation.

2. Materials

2.1 Cell Culture

1. MG1655 *E. coli* strain with plasmid pJBEI-6409 obtained from Taek Soon Lee (Addgene plasmid #47048) ***(6)***.
2. Pre-pre-culture Luria-Bertani medium: 10 g/L tryptone, 5 g/L yeast extract, 10 g/L NaCl, and 30 µg/mL chloramphenicol.

3. Pre-culture and culture M9 medium: 12.7 g/L $Na_2HPO_4.7H_2O$, 3.1 g/L KH_2PO_4, 1 g/L NH_4Cl, 0.5 g/L NaCl, 0.25 g/L $MgSO_4.7H_2O$, 15 mg/L $CaCl_2.2H_2O$, 8.1 mg/L $FeCl_3$, 0.89 mg/L $MnCl_2.4H_2O$, 1.7 mg/L $ZnCl_2$, 0.34 mg/L $CuCl_2$, 0.6 mg/L $CoCl_2.6H_2O$, 0.51 mg/L Na_2MoO_4, 10 g/L Glucose.
4. Microplate reader at 600 nm.
5. Inducer: 25 µM isopropyl β-d-1-thiogalactopyranoside (IPTG).
6. Dodecane: 10% of culture volume.

2.2 Sample Preparation

1. Polyamide membrane filter with 0.2 µm pore size.
2. Filtration manifold with vacuum.
3. Wash solution: 12.7 g/L $Na_2HPO_4.7H_2O$, 3.1 g/L KH_2PO_4, 1 g/L NH_4Cl, 0.5 g/L NaCl.
4. Aluminium foil: 7 cm x 7 cm.
5. Liquid nitrogen.
6. Centrifuge.
7. Polyamide filters with 0.45 µm pore size.
8. Oven at 80 ^oC.
9. Weighing Balance.

2.3 Sample Extraction

1. Acetonitrile, methanol, and water (4:4:2).
2. Vortex mixer.
3. Sonicator bath.
4. Ice.

5. Internal standard mixture: 50 µg/mL mevalonic acid-d3 (MVA-d3) and 10 µg/mL thymolphthalein monophosphate (TMP).
6. Centrifugal vacuum concentrator.
7. Methanol: 10 mM ammonium hydroxide (7:3).

2.4 Sample Analysis

2.4.1 Intracellular Metabolites

1. Interface an Agilent 6230 time of flight-mass spectrometer (TOF-MS) and a Dual Agilent Jet Stream (AJS) ion source with an Agilent ultra-performance liquid chromatography (UPLC) 1290 system and a Waters Acquity UPLC BEH C18 (2.1 x 150 mm, 1.7 µm) column with a VanGuard pre-column (2.1 x 5 mm).
2. Use a dual mobile phase system - mobile phase A: 5 mM ammonium formate in water (pH 9.5) and mobile phase B: 5 mM ammonium formate (pH 9.5) in acetonitrile: water (9:1).
3. Calibration mixtures containing different concentrations of intracellular intermediates are used to construct calibration curves: DHAP (DHAP + GAP pool) – 0.04 to 5 µg/mL; DXP – 0.04 to 10 µg/mL; F6P (F6P + G6P pool) – 0.04 to 10 µg/mL; F1,6BP – 0.04 to 6 µg/mL; MVA – 0.04 to 10 µg/mL; R5P (R5P + Ru5P + X5P pool); Pyruvate – 0.04 to 1.5 µg/mL; MVAP – 0.01 to 0.3 µg/mL; GPP – 0.05 to 1.5 µg/mL; FPP – 0.05 to 1.5 µg/mL. To each 100 µL calibration mixture, add 10 µL of internal standard mixture. Calibration curves are used to determine linearity and the concentration of each compound in the prepared samples.
4. Software for metabolite quantitation (e.g., Agilent Masshunter Workstation Quantitative Analysis for TOF).

2.4.2 Extracellular Metabolites

1. Interface an Agilent 1200 high performance liquid chromatography (HPLC) system with a Bio-rad Aminex HPX-87H column (300 x 7.8 mm) and 1260 Infinity II Refractive Index Detector (RID).
2. Prepare 0.01 N sulphuric acid as mobile phase.
3. Use calibration mixtures with different concentrations of extracellular metabolites to construct calibration curves: Glucose – 0.5 to 80 g/L; Lactic acid – 0.125 to 8 g/L; Acetic acid – 0.125 to 80 g/L; Ethanol – 0.5 to 80 g/L. Calibration curves are used to establish linearity and the concentration of each extracellular metabolite in the prepared samples.
4. Software for metabolite quantitation.

2.4.3 Secreted Limonene

1. Equip an Agilent 7890B gas chromatography mass spectrometry (GC-MS) system with a DB-5 ms column.
2. Calibration standards for limonene prepared in ethyl acetate from 0.05 µg/mL to 10 µg/mL. Use calibration curves to determine linearity and the concentration of limonene in the dodecane extracts from samples.
3. Software for limonene quantitation.

3. Methods

An overview of the experimental protocol is presented in Figure 3. The protocol has been divided into five parts: production of bacterial culture; harvesting of samples; preparation of samples; instrumental analysis and mass spectrometry; and data analysis.

3.1 Production of bacterial culture

Limonene bacterial strains are grown as 50 mL cell cultures in 250 mL flasks and induced with IPTG. Each flask is sacrificed in duplicates from 2 h, 3 h, 6 h, and 7 h post-IPTG induction to determine intracellular and extracellular metabolite concentrations.

1. For pre-pre-culture, select a single colony and inoculate in 5 mL Luria-Bertani medium and chloramphenicol overnight at 37 ^{o}C and 220 rpm.
2. For pre-culture, wash cell pellet from pre-pre-culture and re-suspend in 50 mL M9 medium, leave overnight at 30 ^{o}C and 220 rpm.
3. For culture, add 100 µL of pre-cultures to 50 mL M9 medium in 250 mL flasks. Incubate at 30 ^{o}C and 220 rpm.
4. Upon reaching optical density of 1 at 600 nm, add IPTG.
5. Add 5 mL dodecane to cell cultures.
6. Leave at 30 ^{o}C and 220 rpm.
7. Sacrifice flasks over time-points at 2 h, 3 h, 6 h, and 7 h post-IPTG induction.

3.2 Harvesting of samples

3.2.1 Intracellular metabolites

1. Subject 10 mL of cell cultures from various time-points to fast filtration.
2. Wash cell pellet left on polyamide membrane with 5 mL wash solution.
3. Place the polyamide membrane onto an aluminium foil and fold into half prior to quenching in liquid nitrogen.
4. Remove from liquid nitrogen and store at -80 ^{o}C prior to metabolite extraction.

3.2.2 Extracellular metabolites and secreted limonene

1. Centrifuge 50 mL cell cultures from various time-points for 10 mins at 3000 rpm.
2. Remove dodecane layer containing secreted limonene and store at -80 ^{o}C prior to GC-MS analysis.

3. Place collected supernatant samples at -80 ^{o}C prior to HPLC-RID analysis for extracellular metabolites.
4. Dry cell pellets in oven overnight prior to weighing.

3.3 Preparation of samples

3.3.1 Intracellular metabolites

1. Remove membranes from aluminum foil and place into 5 mL extraction solvent consisting of methanol, acetonitrile, and water.
2. Vortex for 1 min.
3. Sonicate for 3 mins, 3 times, placing in ice prior to each sonication.
4. Place extracts into glass tubes and spike with 20 µL internal standards mixture.
5. Remove solvents using vacuum concentrator and reconstitute with 200 µL methanol and 10 mM ammonium hydroxide mixture. Dilute samples by 10 times and 100 times, as necessary.

3.3.2 Extracellular metabolites and secreted limonene

1. Filter 1 mL of supernatant sample using polyamide filter prior to HPLC-RID analysis.
2. Dilute limonene samples using ethyl acetate by 10 or 100 times prior to GC-MS quantitative analysis.

3.4 Instrumental analysis and mass spectrometry

3.4.1 Intracellular metabolites

1. For each run, inject 2 µL of sample. Start with 100% mobile phase A from 0-3.5 min with 0.1 mL/min flow rate to 100% mobile phase B at 12 min and hold for 8 min with increased flow rate to 0.5 mL/min. At 20 min, recalibrate system back to 100%

mobile phase A for 5 min and hold for another 5 min. Keep column temperature constant at 35 ^{o}C.

2. Negative electrospray ionisation is used with the following TOF settings: Gas temperature, 325 °C; Gas flow, 11 L/min; Nebuliser pressure, 35 psi; Sheath gas temperature, 375 °C; Sheath gas flow, 11 L/min; Vcap voltage, 3500 V; Nozzle voltage, 500 V; Skimmer, 65; OctopoleRFPeak, 750; Scan rate, 2 spectra/s. Fragmentor voltage varied is throughout each 35 min sample analysis: 2 – 7.5 min, 140 V; 7.5 – 15 min, 100 V, 140 V and 150 V. UPLC flow diversions are as follows: 0 – 2 min to waste, 2 – 15 min to TOF-MS, and 15 – 35 min to waste.

3.4.2 Extracellular metabolites

1. Set RID temperature at 30 ^{o}C with positive polarity.
2. For each run, inject 5 µL of sample with isocratic gradient and 0.6 mL/min flow rate for 28 min. Set column temperature at 35 ^{o}C.

3.4.3 Secreted limonene

1. Equip an Agilent 7890B GC-MS system with a DB-5 ms column.
2. For each run, use a 10:1 split ratio with 10 mL/ min split flow and an injection of 1 µL. Use a GC oven temperature program of 40 ^{o}C for 3 min, followed by a 10 ^{o}C/min ramp to 100 ^{o}C and another 60 ^{o}C/min ramp to 220 ^{o}C with a hold time of 2 min. Injector and MS transfer line temperatures are 250 ^{o}C and 280 ^{o}C, respectively.
3. Operate the MS in selected ion-monitoring (SIM) mode using ions of m/z 136, 68 and 93, representing the molecular ion and two abundant fragmental ions of limonene.

3.5 Data analysis

1. Execute the quantitation of metabolites with appropriate dilutions as necessary with calibration curves obtained from metabolite standards.

3.6 COPASI software and execution

The construction and analysis of kinetic models can be done via the open-source and stand-alone program COPASI ***(23)***. COPASI (https://copasi.org) is available both in a graphical user interface (CopasiUI) and command line version (CopasiSE). Basico is also available as a simplified interface to using COPASI from Python.

3.6.1 Model construction

1. Begin first by defining the model’s units under the Model tab. Set “Quantity Unit” and “Volume Unit” to be the same as the metabolite concentration unit used in the time-series data. Thus, if μmol/l is used, set “μmol” and “l” respectively. Keep “Time Unit” no longer than the minute scale (see Note 2).
2. Under Model → Biochemical → Compartments, create the compartment(s) to describe where the reactions would be taking place. For model simplicity, we would create just one compartment, using COPASI’s default settings of 3D dimensionality and fixed simulation type.
3. Under Reactions, add the reactions that occur in the metabolic network. Specify the reaction name and its chemical equation. Use equal sign (=) and “->” to indicate reversible and irreversible reaction respectively. Alternatively, use the reversible checkbox to toggle the reaction reversibility. For reactions that involve stoichiometric coefficient greater than 1, for instance AtoB reaction of 2AcCoA = AtAcCoA + CoA, write the reaction equation as shown in Figure 4A. If “2AcCoA” or “2*AcCoA” (no spaces) were used instead, COPASI will treat it as a separate species from AcCoA.
4. Assign the appropriate rate law to describe the reaction kinetics from the drop-down menu. Note that rate laws that match the reaction characteristics (reversibility and the same number of substrates and/or products) would only appear for selection. COPASI

has a few predefined rate laws to choose from, but a user-defined rate law can also be created by clicking on the “+” sign (Figure 4A).

5. In the window that appears next, enter the rate law formula in the Formula box. If the equation is syntactically correct, the button next to the box will become white and clickable to view the equation in a more visually understandable manner (Figure 4B, top). The button will appear grey and unclickable otherwise (Figure 4B, bottom; also see Note 1). Additionally, indicate if the rate law is for a “reversible”, “irreversible” or “general” reaction below.
6. While the rate law is being entered, COPASI automatically parse and extract the variables as the default “Parameter” type. From the dropdown under Description, select if a variable is a “Substrate”, “Product”, or “Modifier” (reaction activator or inhibitor), where necessary.
7. When defining the new rate law is complete, click “Commit”, return to the reaction, and select it from the dropdown menu. Perform mapping of variables between the reaction and formula variables (Figure 4C).
8. As the reactions are created, species in the network will be automatically populated under the Species tab.

3.6.2 Parameter estimation

The reaction parameters such as $V_{\max}$ and K_M can be obtained from *in vitro* studies of the enzyme kinetics. However, such information may not be available for some enzymes. In addition, *in vitro* conditions may not necessarily reflect the conditions in living cells or systems ***(24)***. In such cases, parameter values would need to be estimated based on time-series experimental data relevant to the metabolic network being built.

1. Save experimental time course results in tab- or comma-delimited plain text file format (.csv, .tsv, .txt etc.), having the column names in the first row.
2. Under Tasks → Parameter Estimation, import the experimental data and correctly inform COPASI the separator that should be used to automatically parse in the table of values (Figure 5A).
3. Select “Time Course” as the Experiment Type and indicate how each column of the input data associates with model elements, either “Time”, “independent”, “dependent” or “ignored” (default). Upon selecting “dependent”, a pop-up window appears prompting you to specify what model element the column maps to. Navigate to Species → Transient Concentrations and select for the correct mapping (Figure 5B).
4. Choose either mean or mean square as the “Weight Method” as either ensures that all data columns contribute in the same order of magnitude towards the error of fit regardless of the size of their numerical values (Figure 5A) ***(25)***.
5. Save the setup of the imported experimental data by clicking “OK”.
6. At the main screen of Parameter Estimation, select for which parameters to perform estimation for. The start value, lower and upper bounds can be informed by previously reported parameter values of the enzyme, if available.
7. Select for the method to use for estimation. Typically, the global optimization method, genetic algorithm, is used.
8. Set the number of generations and population size. While higher numbers would increase the probability of reaching global minimum, 1000 generations and a population size of 300 is a good start, balancing for both optimized solutions as well as time-intensiveness to run the task.
9. Choose where to save the parameter estimation report file on your computer by clicking on the “Report” button and selecting the path and file name.

10. Under Output Assistant, select to create Progress of Fit and Parameter Estimation Result plots. The former will inform us if there is improvement in the fit while the estimation task is still running (Figure 6A), and the latter will let us know how the simulations compare to the experimental data after parameter values has been estimated (Figure 6B).

The details (kinetic equations and parameter values) of our fitted model can be found in the following GitHub repository: https://github.com/heeyanting93/CBioOmics_COPASI. As parameter estimation runs can take up to hours to complete and CopasiUI does not allow multiple application instances (other than through the command line `open -n CopasiUI.app` call), we have also included in the repository a Python script `run_parameter_estimation.py` utilizing Basico to run parameter estimation and can be adapted to one's own use. This way, the CopasiUI remains available to build and test out other model variations. The script also reads the output parameter estimation text file to plot the progress of fit curve and simulation vs measurements plot.

3.6.3 Model testing

Besides the time-series data from the wildtype strain, further experimental time-series data from mutant strains should be obtained where parts of the metabolic network are suppressed (knockdown/out). These additional datasets would enable us to assess if our built model is valid and reflects the system not just under wildtype conditions.

To perform *in silico* knockouts, set the $V_{\max}$ of a reaction step of interest to zero. Run a time course simulation and compare if the simulation results agree with the experimental data. Ideally, such model testing should be tested at multiple parts of the network.

3.6.4 Model validation

After the model has been tested, subsequent perturbations to the network can be done to predict how to improve limonene yield. For instance, looking at the network topology in Figure 1, reducing the flux to lactic acid and ethanol by knocking down ADH and LDH enzymes look likely to increase limonene production (Figure 7). Using these predictions, we can then perform the necessary manipulation in the laboratory to validate if these predictions hold true.

4. Notes

1. COPASI considers variables in rate law equations that start with a number as syntactically incorrect, thus 3PG is written as PG3 instead.
2. Reaction rate (V_{max}) values will get very large if defined in terms of hours, for example, and might pose an issue when trying to narrow the upper and lower bounds for parameter estimation later.

Summary/Conclusion

This work shows the generation of time-series metabolomics data from an engineered limonene-producing bacterial strain that can be used to create a dynamic model. Bottlenecks, such as the loss of flux or lack of co-factors, in the validated dynamic model can then be studied *in silico* to enhance limonene yield. This approach to enhance the yield of a desired product can also be applied to other engineered microbes.

Acknowledgement

This work was supported by the Intra-create Thematic Grant "Cities" (grant number: NRF2019-THE001-0007) under the EcoCTs project. The EcoCTs research project is supported by the National Research Foundation, Prime Minister's Office, Singapore, under its campus for Research Excellence and Technological Enterprise (CREATE) programme. In addition, we are thankful to Dr. Floriant Bellvert from MetaToul (Metabolomics & Fluxomics Facilities, Toulouse, France) and its staff members for their experimental guidance and insights. We would also like to acknowledge Dr. Wee Chew from the Singapore Institute of Food and Biotechnology Innovation (SIFBI) for technical support and discussion.

References

1. Pereira M, Oliveira AM (2020) Poverty and food insecurity may increase as the threat of COVID-19 spreads. Public Health Nutr 23 (17):3236-3240. doi:10.1017/S1368980020003493
2. FAO, IFAD, UNICEF et al (2021) The State of Food Security and Nutrition in the World 2021. Transforming food systems for food security, improved nutrition and affordable healthy diets for all. The State of Food Security and Nutrition in the World (SOFI). FAO, Rome, Italy. doi:https://doi.org/10.4060/cb4474en
3. Smith DJ, Helmy M, Lindley ND et al (2022) The transformation of our food system using cellular agriculture: What lies ahead and who will lead it? Trends in Food Science & Technology 127:368-376. doi:https://doi.org/10.1016/j.tifs.2022.04.015
4. Ciriminna R, Lomeli-Rodriguez M, Demma Cara P et al (2014) Limonene: a versatile chemical of the bioeconomy. Chem Commun (Camb) 50 (97):15288-15296. doi:10.1039/c4cc06147k
5. Sun C, Theodoropoulos C, Scrutton NS (2020) Techno-economic assessment of microbial limonene production. Bioresour Technol 300:122666. doi:10.1016/j.biortech.2019.122666
6. Alonso-Gutierrez J, Chan R, Batth TS et al (2013) Metabolic engineering of Escherichia coli for limonene and perillyl alcohol production. Metab Eng 19:33-41. doi:10.1016/j.ymben.2013.05.004
7. Carter OA, Peters RJ, Croteau R (2003) Monoterpene biosynthesis pathway construction in Escherichia coli. Phytochemistry 64 (2):425-433. doi:10.1016/s0031-9422(03)00204-8
8. Reiling KK, Yoshikuni Y, Martin VJ et al (2004) Mono and diterpene production in Escherichia coli. Biotechnol Bioeng 87 (2):200-212. doi:10.1002/bit.20128
9. Willrodt C, David C, Cornelissen S et al (2014) Engineering the productivity of recombinant Escherichia coli for limonene formation from glycerol in minimal media. Biotechnol J 9 (8):1000-1012. doi:10.1002/biot.201400023
10. Yang J, Nie Q, Ren M et al (2013) Metabolic engineering of Escherichia coli for the biosynthesis of alpha-pinene. Biotechnol Biofuels 6 (1):60. doi:10.1186/1754-6834-6-60
11. Martin VJ, Pitera DJ, Withers ST et al (2003) Engineering a mevalonate pathway in Escherichia coli for production of terpenoids. Nat Biotechnol 21 (7):796-802. doi:10.1038/nbt833

12. Shin J, South EJ, Dunlop MJ (2022) Transcriptional Tuning of Mevalonate Pathway Enzymes to Identify the Impact on Limonene Production in Escherichia coli. ACS Omega 7 (22):18331-18338. doi:10.1021/acsomega.2c00483
13. Wu J, Cheng S, Cao J et al (2019) Systematic Optimization of Limonene Production in Engineered Escherichia coli. J Agric Food Chem 67 (25):7087-7097. doi:10.1021/acs.jafc.9b01427
14. Helmy M, Smith D, Selvarajoo K (2020) Systems biology approaches integrated with artificial intelligence for optimized metabolic engineering. Metab Eng Commun 11:e00149. doi:10.1016/j.mec.2020.e00149
15. Kim OD, Rocha M, Maia P (2018) A Review of Dynamic Modeling Approaches and Their Application in Computational Strain Optimization for Metabolic Engineering. Front Microbiol 9:1690. doi:10.3389/fmicb.2018.01690
16. Selvarajoo K (2018) Complexity of Biochemical and Genetic Responses Reduced Using Simple Theoretical Models. Methods Mol Biol 1702:171-201. doi:10.1007/978-1-4939-7456-6_9
17. Selvarajoo K, Takada Y, Gohda J et al (2008) Signaling flux redistribution at toll-like receptor pathway junctions. PLoS One 3 (10):e3430. doi:10.1371/journal.pone.0003430
18. Thornburg ZR, Melo MCR, Bianchi D et al (2019) Kinetic Modeling of the Genetic Information Processes in a Minimal Cell. Front Mol Biosci 6:130. doi:10.3389/fmolb.2019.00130
19. Hargrove JL, Hulsey MG, Beale EG (1991) The kinetics of mammalian gene expression. Bioessays 13 (12):667-674. doi:10.1002/bies.950131209
20. Ryu DD, Park SH (1987) Genetically structured kinetic model for gene product and application of gene switching system to fermentation process control. Ann N Y Acad Sci 506:396-405. doi:10.1111/j.1749-6632.1987.tb23836.x
21. Selvarajoo K (2017) A systems biology approach to overcome TRAIL resistance in cancer treatment. Prog Biophys Mol Biol 128:142-154. doi:10.1016/j.pbiomolbio.2017.02.009
22. Hayashi K, Piras V, Tabata S et al (2013) A systems biology approach to suppress TNF-induced proinflammatory gene expressions. Cell Commun Signal 11:84. doi:10.1186/1478-811X-11-84
23. Hoops S, Sahle S, Gauges R et al (2006) COPASI--a COmplex PAthway SImulator. Bioinformatics 22 (24):3067-3074. doi:10.1093/bioinformatics/btl485
24. Mauch K, Vaseghi S, Reuss M (2000) Quantitative Analysis of Metabolic and Signaling Pathways in Saccharomyces cerevisiae. In: Schiigerl K, Bellgardt K-H (eds) Bioreaction Engineering. Springer, Berlin, Heidelberg, pp 435-477. doi:https://doi.org/10.1007/978-3-642-59735-0_15
25. Mendes P, Hoops S, Sahle S et al (2009) Computational modeling of biochemical networks using COPASI. Methods Mol Biol 500:17-59. doi:10.1007/978-1-59745-525-1_2

Figure Captions

Figure 1: Schematic of different pathways involved in limonene production in *E. coli* engineered with the mevalonate (MVA) pathway. Cofactor consumption is represented by

curved arrows. Intermediates: Glcex, glucose extracellular; Glc, glucose; G6P, glucose-6-phosphate; 6PG, 6-phosphogluconate; X5P, xylulose-5-phosphate; Ru5P, ribulose-5-phosphate; R5P, ribose-5-phosphate; F6P, fructose-6-phosphate; F16BP, fructose-1,6-biphosphate; GAP, glyceraldehyde-3-phosphate; BPG, 1,3-bisphosphoglycerate; 3PG, 3-phosphoglycerate; PEP, phosphoenolpyruvate; PYR, pyruvate; DXP, 1-deoxy-D-xylulose-5-phosphate; Vit B6, flux to vitamin B6 pathway; MEP, 2-C-methylerythritol-4-phosphate; CDPME, 4-diphosphocytidyl-2-C-methylerythritol; CDPMEP, 4-diphosphocytidyl-2-C-methylerythritol-2-phosphate; MEcPP, 2-C-Methylerythritol-2,4-cyclodiphosphate; HMBPP, hydroxymethylbutenyl 4-diphosphate; IPP, isopentenyl diphosphate; DMAPP, dimethylallyl diphosphate; GPP, geranyl diphosphate; FPP, farnesyl diphosphate; LIM, limonene; LIMex, limonene extracellular; AcCoA, acetyl coenzyme A; AtAcCoA, acetoacetyl-CoA; HMGCoA, hydroxymethylglutaryl-CoA; MVA, mevalonate; MVAP, 5-phosphomevalonate; MVAPP, 5-diphosphomevalonate; ACE, acetic acid; ACEex, acetic acid extracellular; ETH, ethanol; ETHex, ethanol extracellular; LAC, lactic acid; LACex, lactic acid extracellular; AKG, α-ketoglutarate; SucCoA, succinyl CoA; SUC, succinate; SUCex, succinate extracellular; FUM, fumarate; OAA, oxaloacetate. Enzymes: PTS, phosphotransferase system; HK, hexokinase; G6PDH, lumped reactions of glucose-6-phosphate dehydrogenase and 6-phosphogluconolactonase; PGDH, 6-phosphogluconate dehydrogenase; Tkb, transketolase; PGI, phosphoglucose isomerase; PFK, phosphofructokinase; FBA, fructose-1,6-biphosphate aldolase; GDH, glutamate dehydrogenase; PGK, phosphoglycerate kinase; ENO, enolase; PYK, pyruvate kinase; DXS, DXP synthase; DXR, DXP reductase; ISPD, CDPME synthase; ISPE, CDPME kinase; ISPF, MEcPP synthase; ISPG, HMBPP synthase; ISPH, HMBPP reductase; IDI, isopentenyl diphosphate isomerase; ISPA, farnesyl diphosphate synthase; LS, limonene synthase; PDH, pyruvate dehydrogenase; AtoB, acetyl-CoA acetyltransferase; HMGS, HMGCoA synthase; HMGR, HMGCoA reductase; MK, mevalonate kinase; PMK;

phosphomevalonate kinase; PMD, diphosphate mevalonate decarboxylase; LDH, lactate dehydrogenase; PoxB, pyruvate oxidase; PCK, phosphoenolpyruvate carboxykinase; PPC, phosphoenolpyruvate carboxylase; ACS, acetyl-CoA synthetase; PTACK, lumped reactions of phosphate acetyltransferase and acetate kinase; ALDHB, aldehyde dehydrogenase B; ALDH, aldehyde dehydrogenase; ADH, alcohol dehydrogenase; CSICD, lumped enzymatic reactions of citrate synthase, aconitate hydratase A, aconitate hydratase B and isocitrate dehydrogenase; AKGDH, α-ketoglutarate dehydrogenase; SCS, succinyl-CoA synthetase; FRD, fumarate reductase; MDH, malate dehydrogenase.

Figure 2: ODE of BPG, which is produced from GAP by the GDH enzyme and later converted into PG3 by PGK. 3PG is written here as PG3 instead (see Note 1). Kinetic parameters, V_{max} of forward (V_{f}) and reverse (V_{r}) reaction, and K_M belong to their respective enzyme as denoted by the parentheses as either (GDH) or (PGK).

Figure 3: Workflow of generating time-series metabolomics data.

Figure 4: Model construction. (A) Adding reaction to the model. (B) Adding user-defined rate laws. Top and bottom show syntactically correct and incorrect (due to a missing closing bracket) mathematical formula. (C) Performing mapping of variables between the reaction and rate law variables.

Figure 5: Setting up experimental data for parameter estimation. (A) Loading in the experimental data text file. (B) Mapping input data columns to model species.

Figure 6: Output from a parameter estimation run. (A) Progress of fit graph from running COPASI’s parameter estimation. (B) Parameter estimation result. For each species, the

simulation result (line) is plotted along with the experimental data (points with standard deviation).

Figure 7: *In silico* limonene simulations for four knockout conditions (ISPG KO, ADH KO, ALDH KO, LDH KO), where WT refers to the engineered *Escherichia coli* MG1655 strain with plasmid pJBEI-6409. The results show that ALDH KO produces the best enhancement of limonene yields.