Brunel University London

Brandenburg University of Technology Cottbus - Senftenberg

Derivation of a biomass proxy for dynamic analysis of whole genome metabolic models

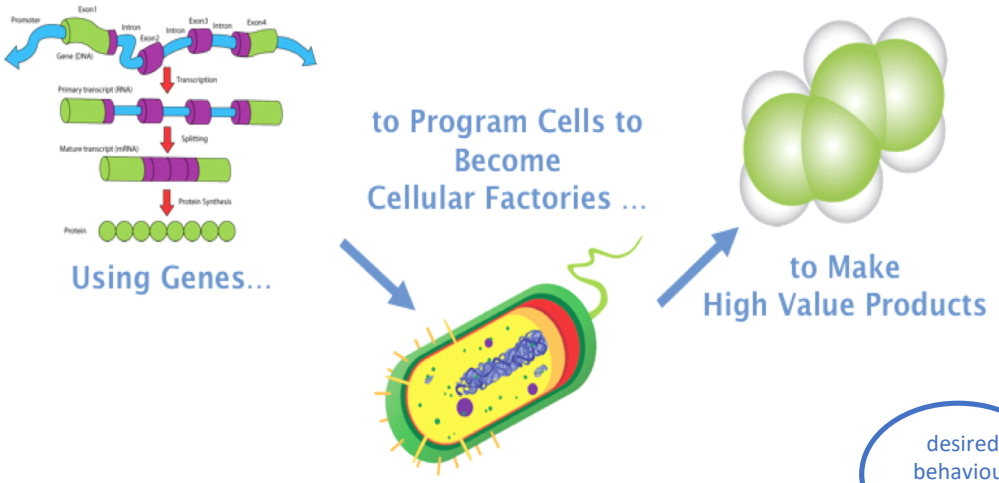
Timothy Self[*], David Gilbert[*], Monika Heiner[$*]

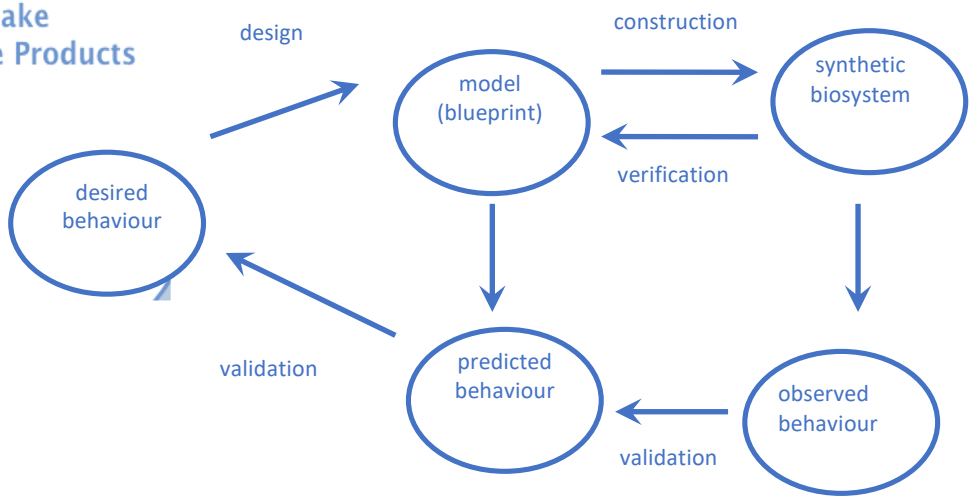
[*] Brunel University London, UK
[$] Brandenburg Technical University, Cottbus, Germany

david.gilbert@brunel.ac.uk, monika.heiner@b-tu.de

CMSB 2018, Brno/CZ, September 14, 2018

Synthetic Biology & Metabolic Engineering





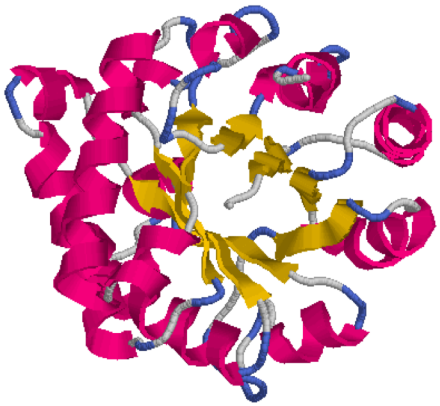
Genes to Systems

DNA

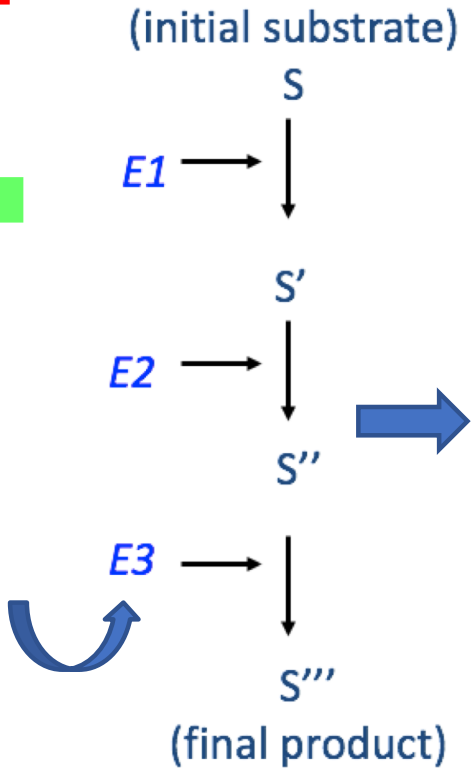
"gene"

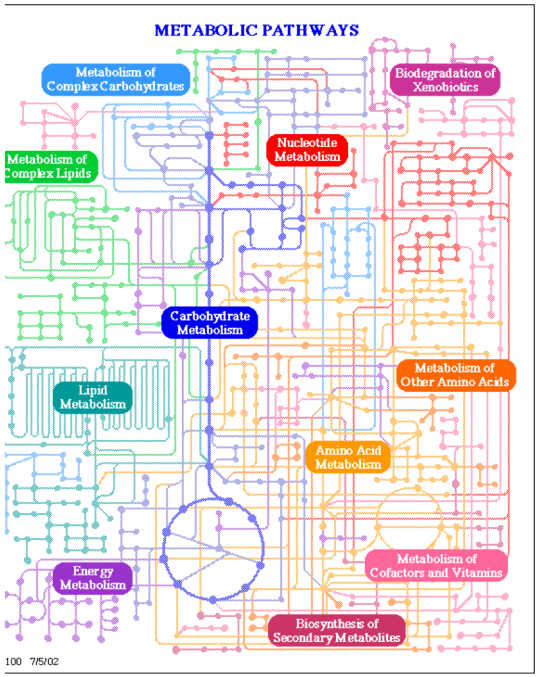
mRNA

Protein sequence



Folded Protein





System design to genes (inverse problem)

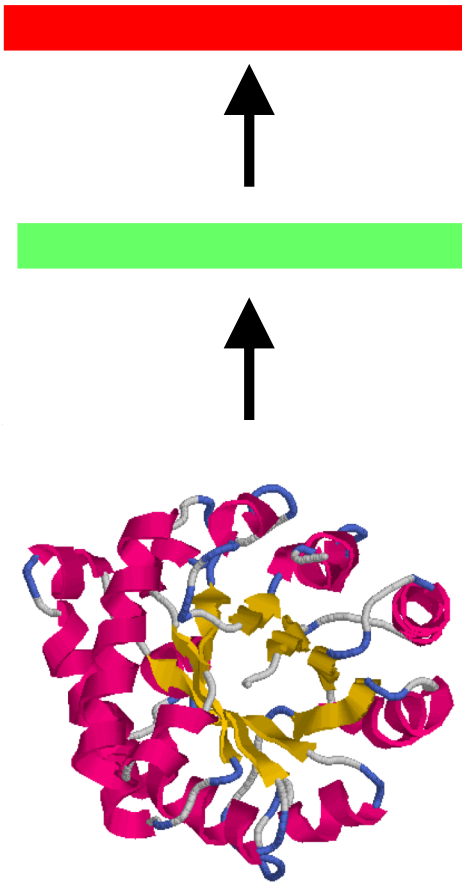
DNA

"gene"

mRNA

Protein sequence

Folded Protein



(initial substrate)

S

E1

S'

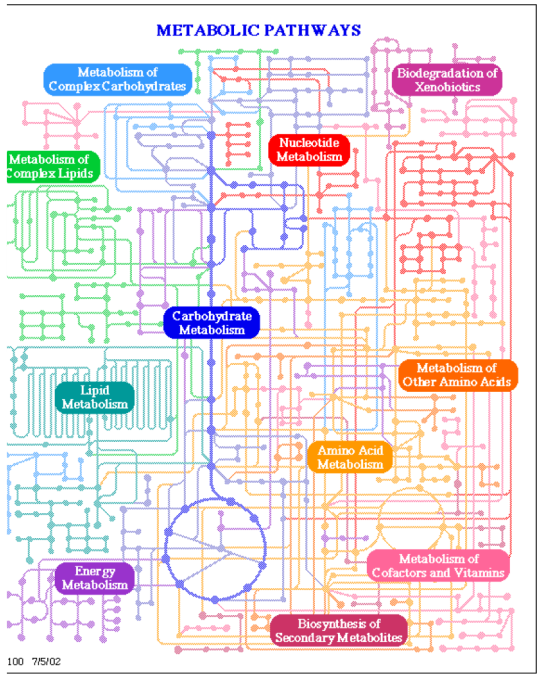
E2

S''

E3

S'''

(final product)



The challenges!

- Engineer micro-organisms to produce target biochemicals
- State of the art - done with FBA models,
  incorporating ‘biomass function’ as one target during optimisation
- Our aim: add the power of dynamic models
- Find a general approach to
  finding dynamic proxy for the biomass function

Motivation for considering biomass per se

When designing modifications to an organism (e.g. *E.coli*),
in order to improve production of some target biochemical

Ensure that

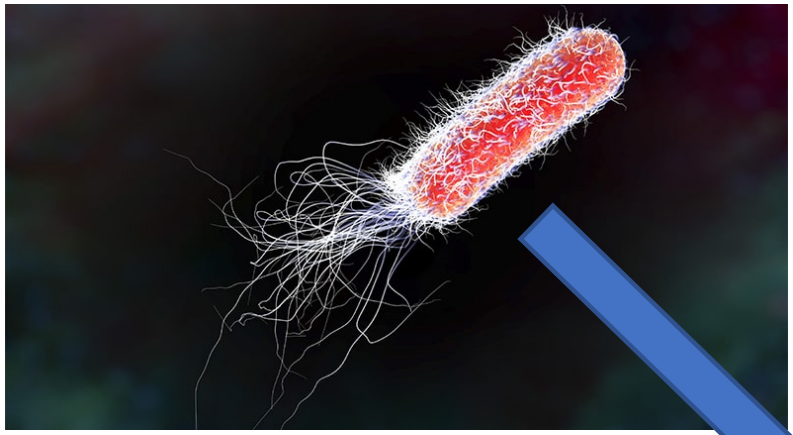
(i) The organism survives [is alive]

(ii) Preserve a good balance between target biochemical & biomass:

A lot of biomass - all energy used for this

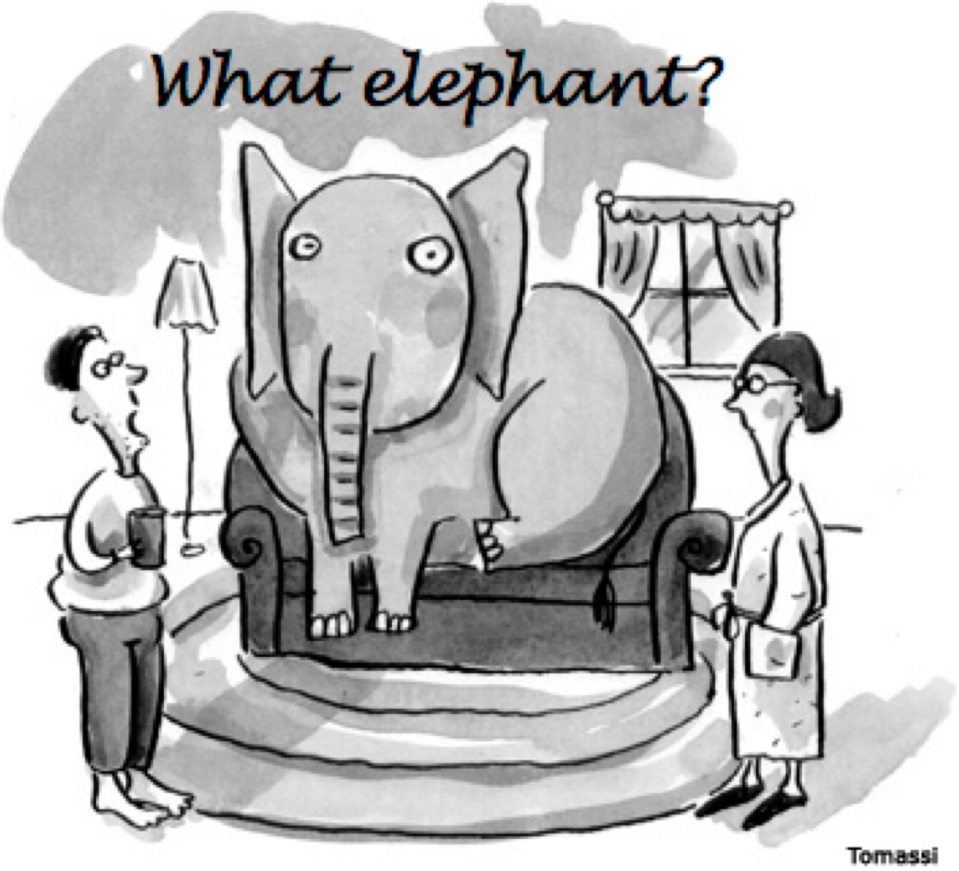
Little biomass - will affect the viability (not very robust organism)

The biomass function



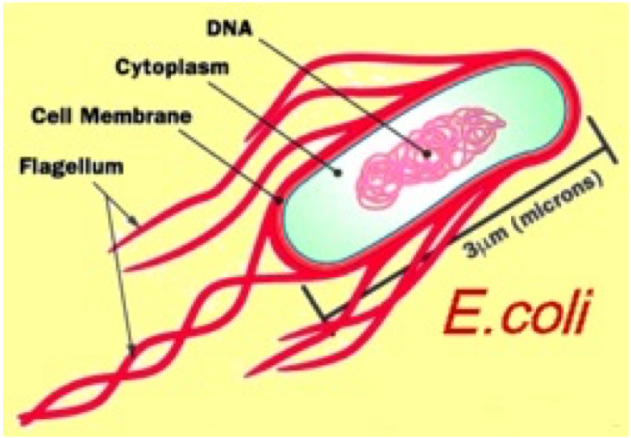
Recycle metabolites

Energy for reproduction



Biomass composition: the "elephant in the room" of metabolic modelling, Dikicioglu, Kirdar, Oliver. Metablomics 2015

The state of the art - models



- Whole genome scale metabolic models (GEMs) exist for a range of bacteria (BiGG database), and also for human (Biomodels database: RECON1&2).

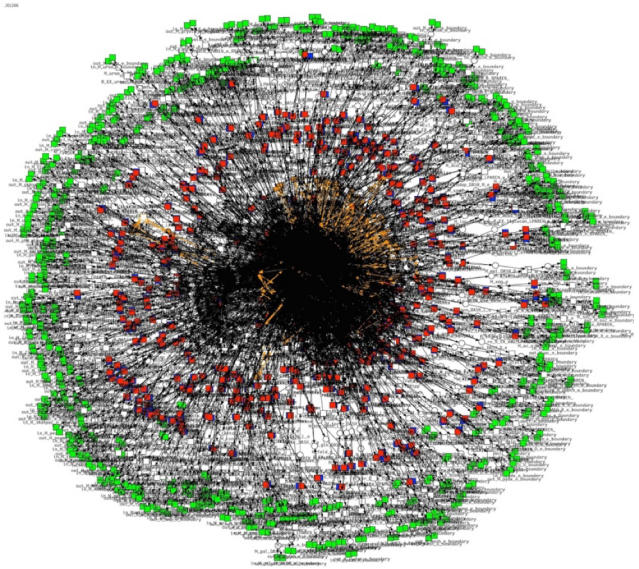
- Stoichiometric models, enriched by constraints of biological meaningful flux ranges -> constraint-based models -> FBA.

- Can be configured for different growth conditions, (an/aerobic & varying carbon sources).

- No rate constants, no initial concentrations.

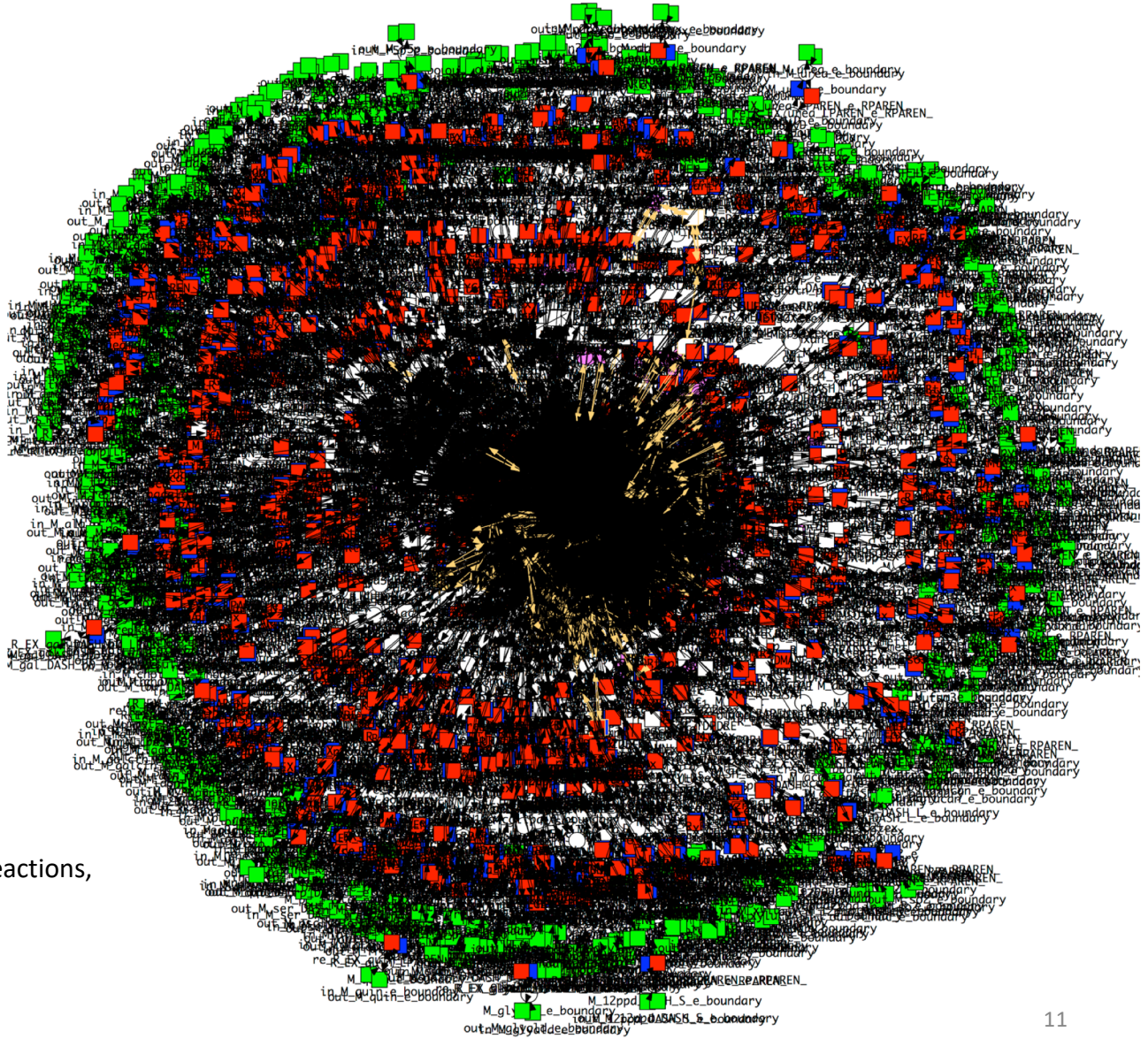
Monk et al., Genome-scale metabolic reconstructions of multiple Escherichia coli strains highlight strain-specific adaptations to nutritional environments. PNAS, 110(50):20338–20343, 2013.

Example model: E. coli K-12



- > 4000 genes
- 1400 involved in metabolism
- Compartmental structure: periplasm, cytosol, extracellular space
- ~3000 reactions → 4000 Petri net transitions
- ~1200 unique metabolites → 2300 metabolites (species, Petri net places) respecting the compartmental structure
- 35 subsystems

Petri net for *E. coli* K-12 genome scale metabolic model (GEM)



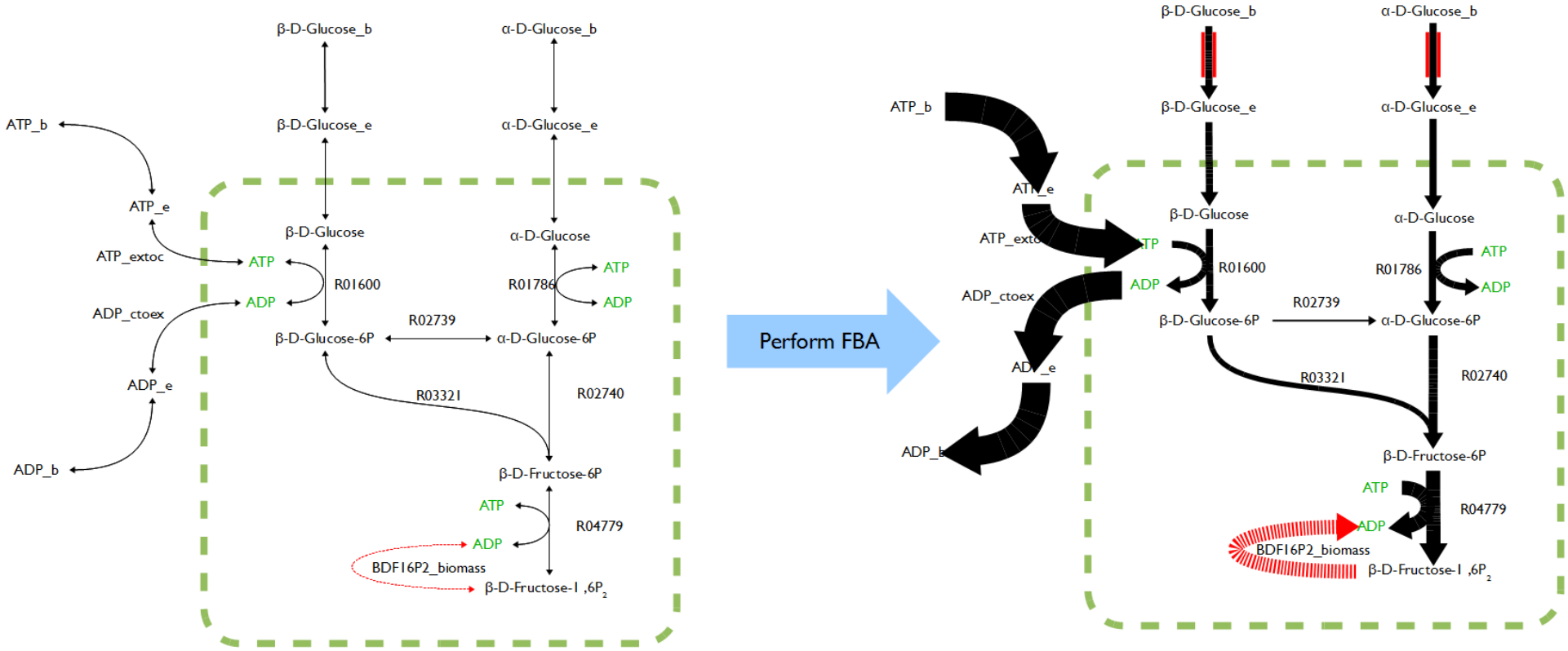
Layout generated with Snoopy.

Colour code:

green: generated boundary transitions,
blue: reversible reactions,
red: generated reverse direction for reversible reactions,
yellow: P-invariants.

Flux Balance Analysis

FBA calculates the flow of metabolites through a metabolic network **in the steady state**



By Thomas Forth Wikimedia Commons

FBA versus Dynamic simulation

FBA

- *Does not require initial concentrations nor rate functions, thus no rate parameters;*
- Steady state only (based on flux constraints);
- Can only determine ***relative*** fluxes at steady state;

Dynamic simulation

- *Requires initial concentrations and rate functions, including rate parameters;*
- Additionally considers the *transient state;*
- Can model the development of the system, e.g. under changing environmental conditions;

Simulation efficiency - challenges

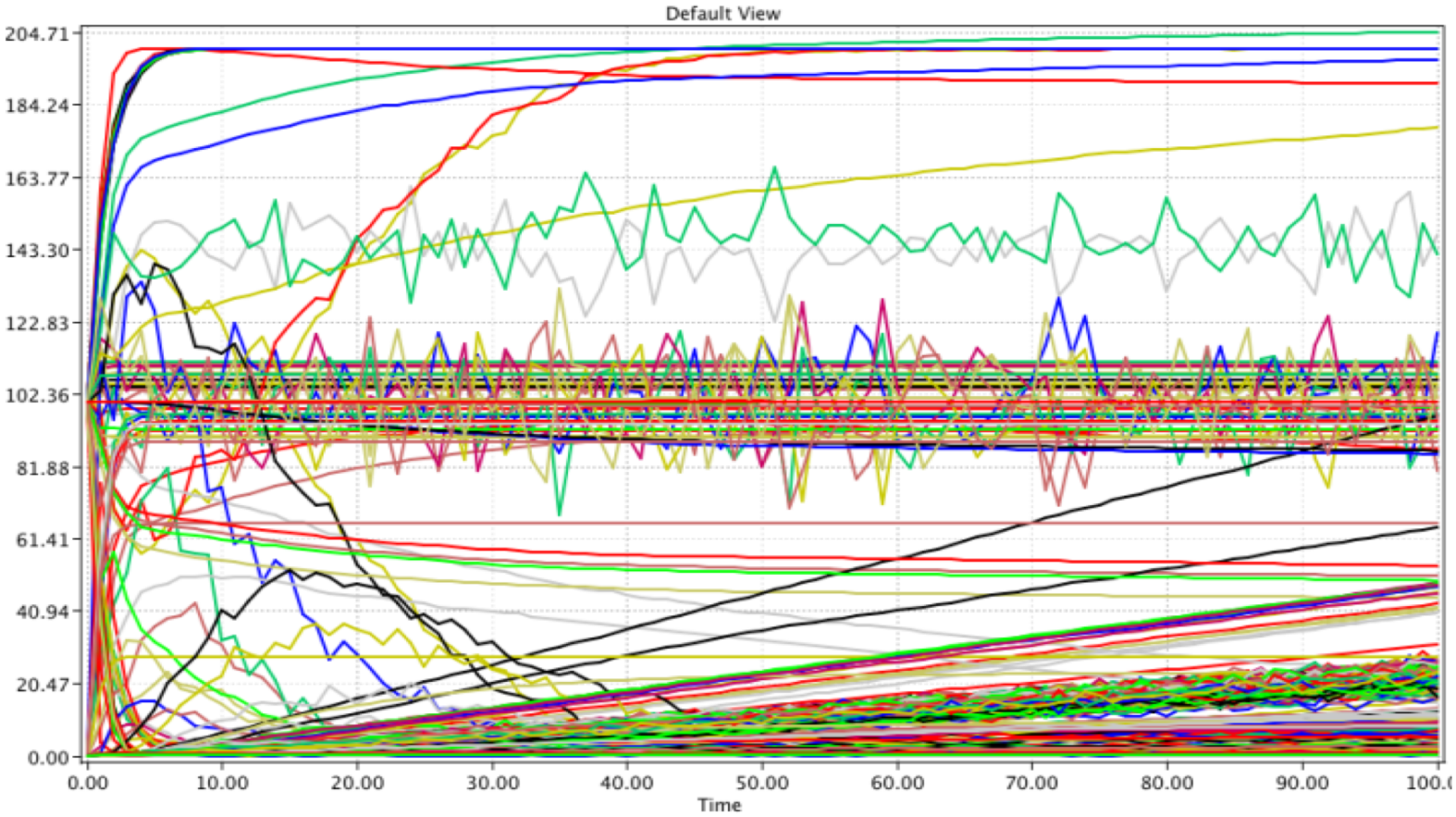
- Very large systems, highly stiff in nature
- Severe numerical problems for continuous simulation of the set of Ordinary Differential Equations induced by a biochemical reaction network
- Unacceptably long runtimes for stochastic simulation algorithms (SSA).

- Gillespie's direct method: GEM E. coli K-12
  - ~90 minutes for 1 run of 1000 time points,
  - 62.5 days for 1000 runs on a single-core workstation
  - +50% for tau-leaping SSA.

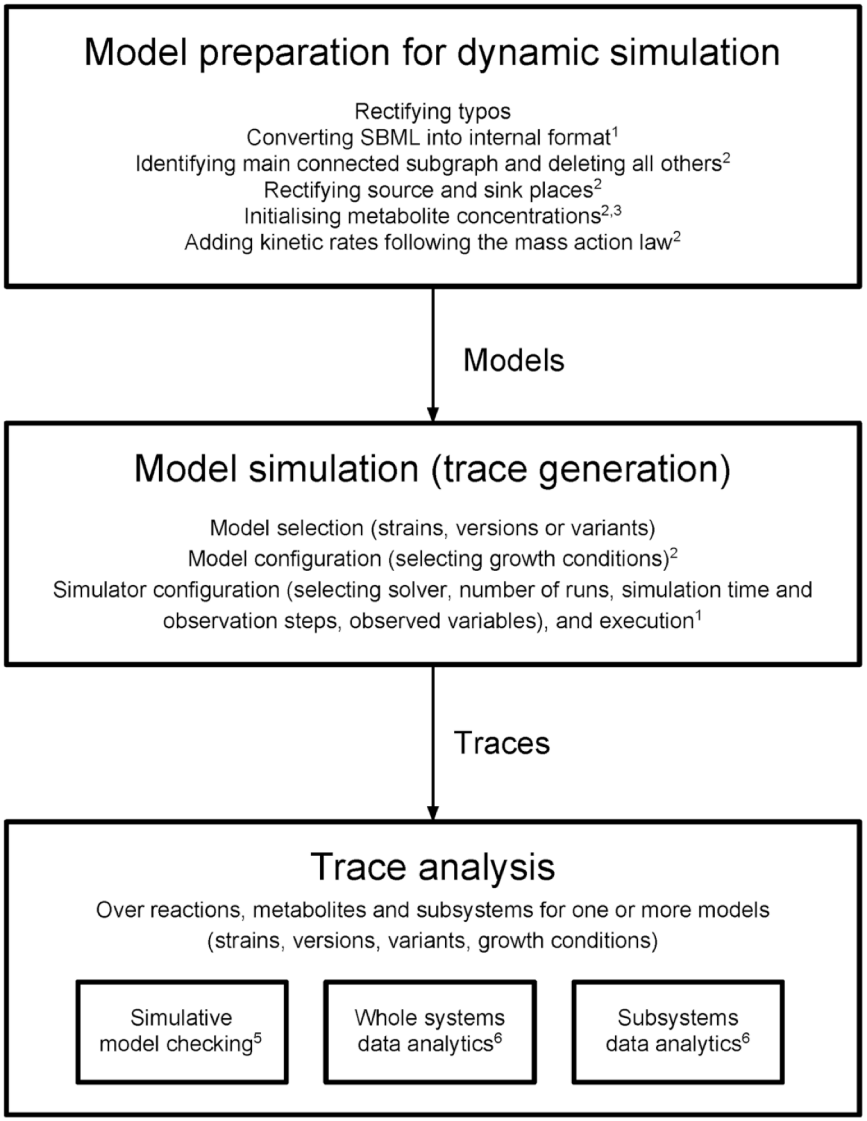
- Discrete-time δ-leaping
  - Typically <1s for 1 run, ~14 minutes for 1000 runs for a GEM.

C Rohr: Discrete-Time Leap Method For Stochastic Simulation; Fundamenta Informaticae, 160(1-2):181-198, 2018

Dynamic simulation using delta leaping



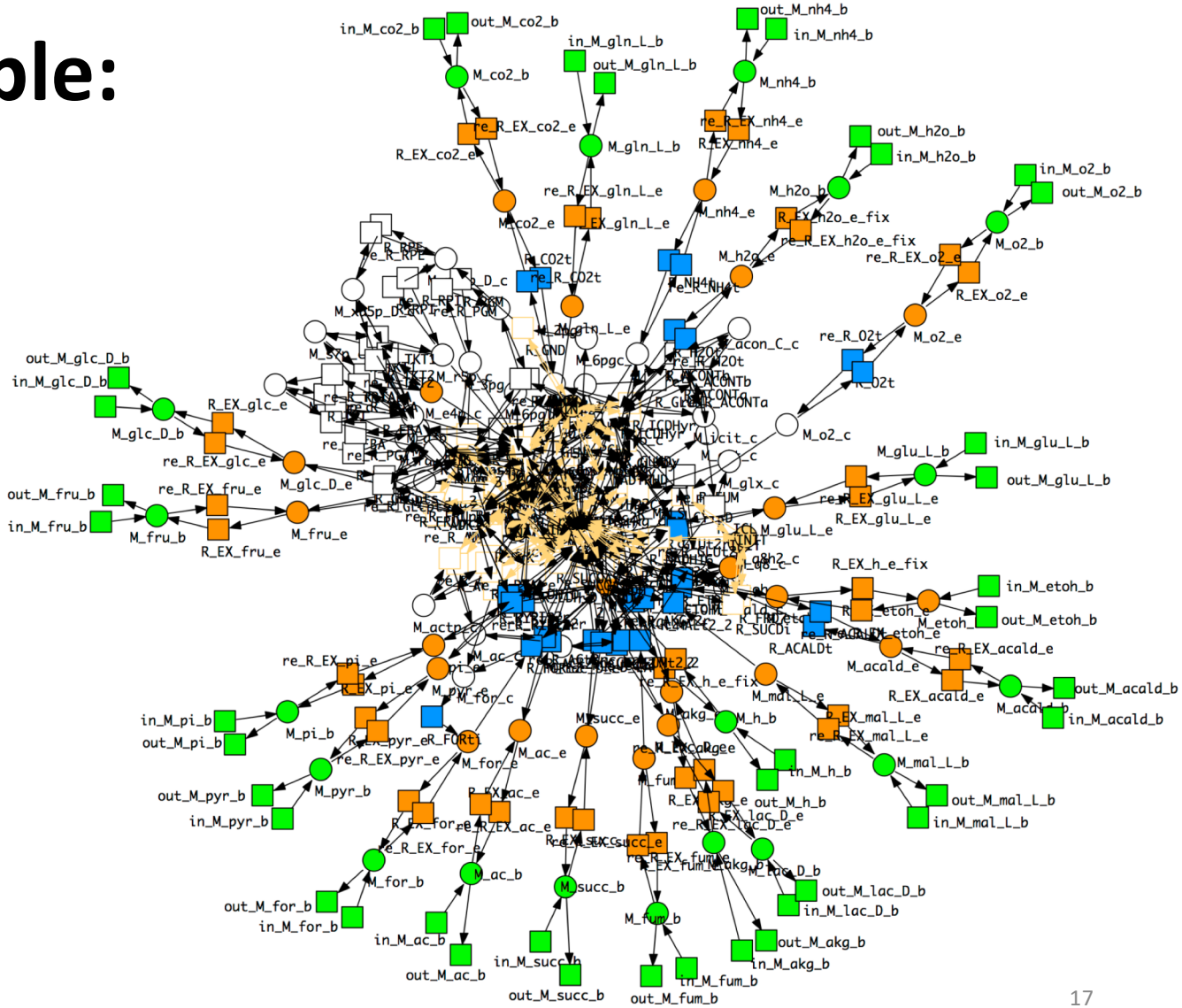
Workflow from static FBA to dynamic models



Gilbert, Heiner, Jayaweera, Rohr:
Towards dynamic genome-scale models,
Briefings in Bioinformatics, 2017.

Runnning example: E.coli core

Reduced version of E.coli K-12



Layout generated with Snoopy.

colour code

- green:
  boundary condition
- orange:
  reversible exchange reactions
- blue:
  transport reactions
- yellow:
  P-invariants

FBA biomass function for E. coli core K12

	metabolite	stoichiometry	metabolite	stoichiometry
substrates	M_3pg_c	1.496	M_accoa_c	3.7478
	M_atp_c	59.81	M_e4p_c	0.361
	M_f6p_c	0.0709	M_g3p_c	0.129
	M_g6p_c	0.205	M_gln_L_c	0.2557
	M_glu_L_c	4.9414	M_h2o_c	59.81
	M_nad_c	3.547	M_nadph_c	13.0279
	M_oaa_c	1.7867	M_pep_c	0.5191
	M_pyr_c	2.8328	M_r5p_c	0.8977
products	M_adp_c	59.81	M_akg_c	4.1182
	M_coa_c	3.7478	M_h_c	59.81
	M_nadh_c	3.547	M_nadp_c	13.0279
	M_pi_c	59.81		

Growth conditions

- The model can be configured to investigate the effect of different growth conditions using the 20 extracellular species.
- 13: carbon source growth conditions (formate ignored due to viability).
- 5: ingredients of a minimal growth medium based on M9: CO_2, H^+, H_2O, D-Glucose, ammonium & phosphate.
- Plus Oxygen.
- Each carbon source considered aerobically & anaerobically:
  2*13 = 26 single growth conditions
  (ignoring formate).

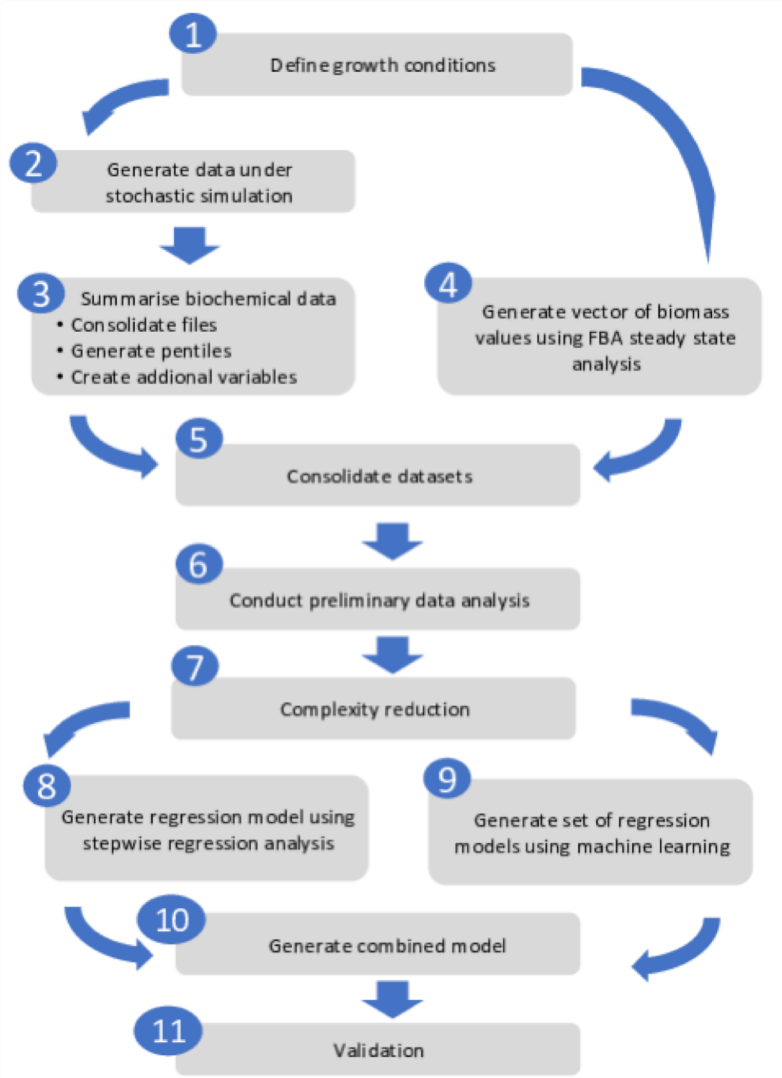
The biomass function & dynamic simulation

- The biomass function for constraint-based GEMs does not work correctly under the dynamic simulation of transient behaviour without quasi-steady state assumption

- → Due to the complexity in terms of the number of variables and specificity in terms of the stoichiometries of the function.

	metabolite	stoichiometry	metabolite	stoichiometry
substrates	*M_3pg_c*	1.496	*M_accoa_c*	3.7478
	M_atp_c	59.81	*M_e4p_c*	0.361
	M_f6p_c	0.0709	*M_g3p_c*	0.129
	M_g6p_c	0.205	*M_gln_L_c*	0.2557
	M_glu_L_c	4.9414	*M_h2o_c*	59.81
	M_nad_c	3.547	*M_nadph_c*	13.0279
	M_oaa_c	1.7867	*M_pep_c*	0.5191
	M_pyr_c	2.8328	*M_r5p_c*	0.8977
products	*M_adp_c*	59.81	*M_akg_c*	4.1182
	M_coa_c	3.7478	*M_h_c*	59.81
	M_nadh_c	3.547	*M_nadp_c*	13.0279
	M_pi_c	59.81		

Workflow of key analytical steps



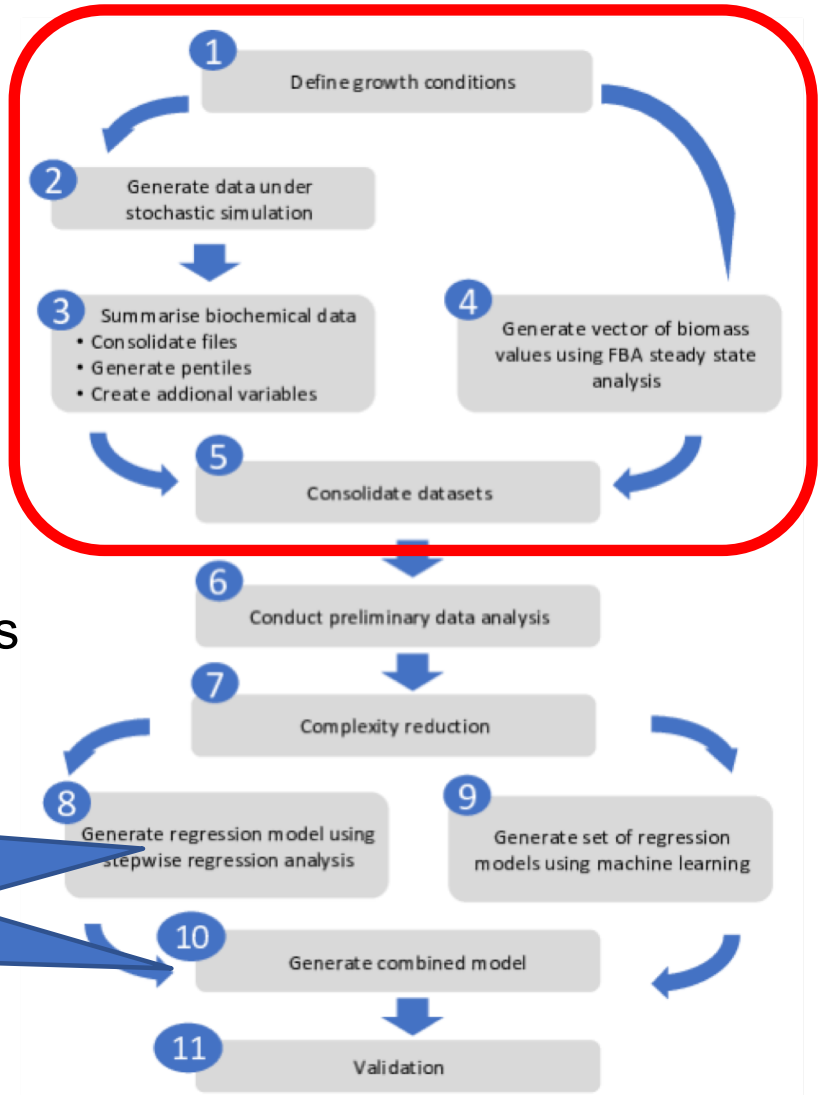
Workflow of key analytical steps

Data collection

Dataset was created for analysis with

- 300 variables (metabolites & reactions
- 26 single growth condition

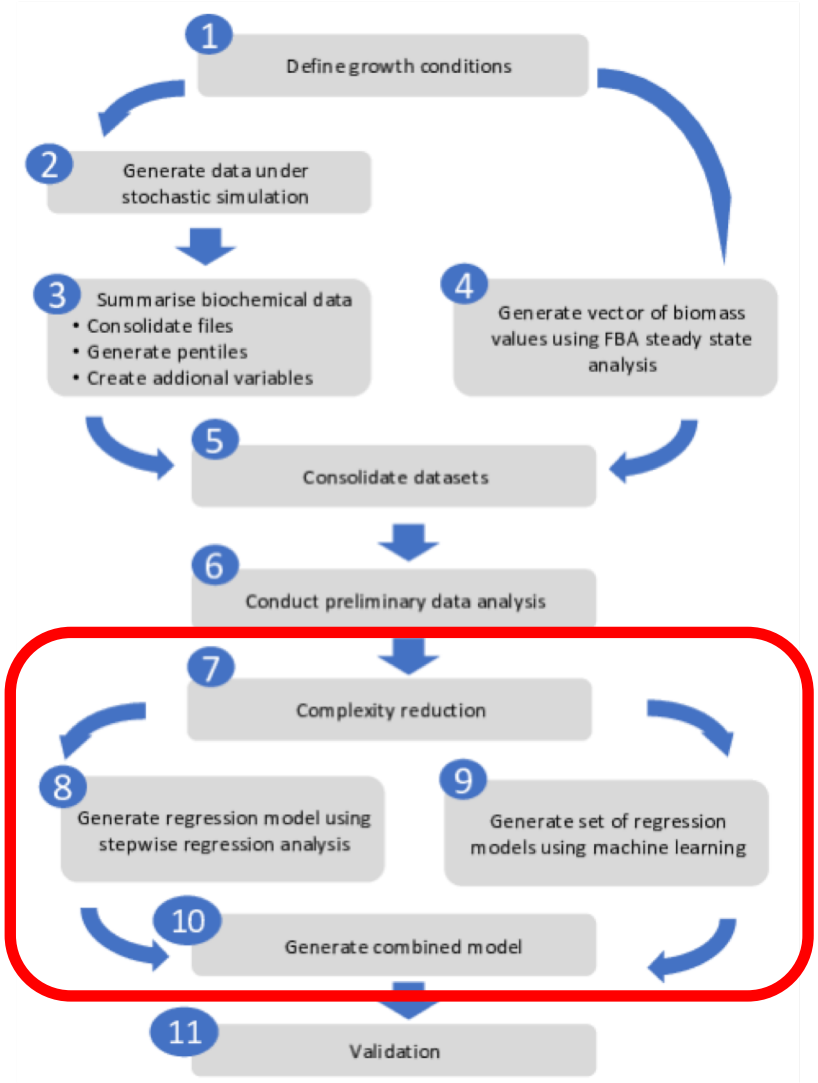
Not enough data!
so extended by 156 pairwise growth conditions



Workflow of key analytical steps

Data analytics techniques

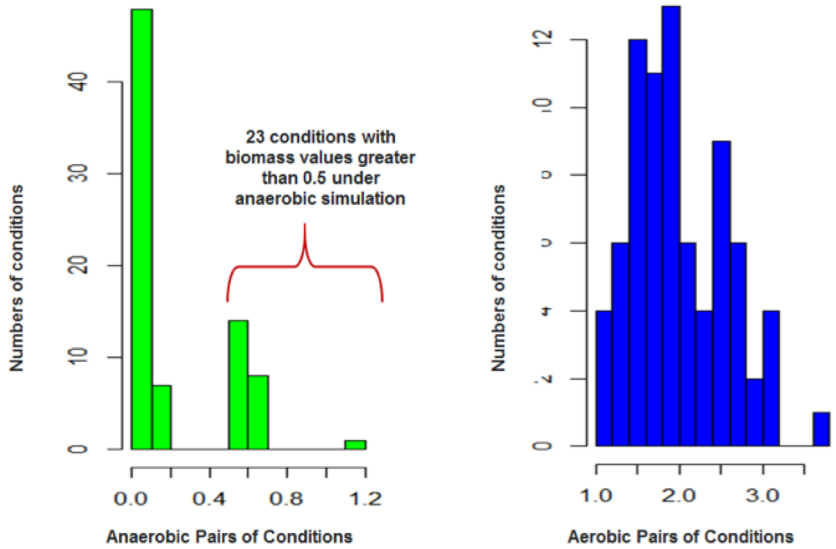
- Complexity reduction
- Stepwise regression analysis
- Machine learning based algorithmic approach to regression (random forest – ensemble over decision trees)



Initial analysis

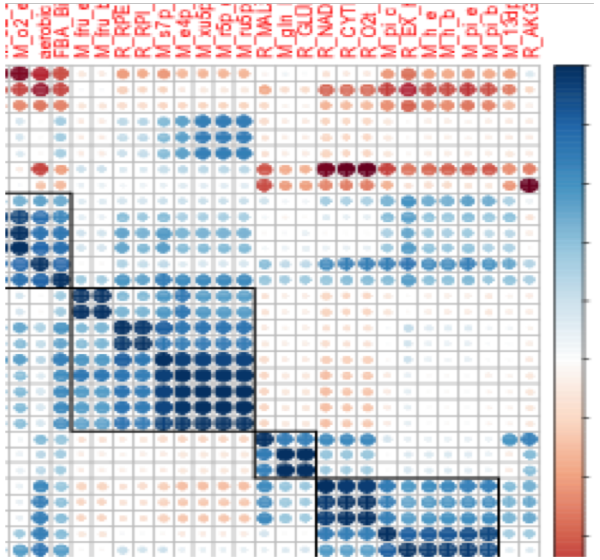
- Anaerobic conditions: zero inflated distribution (many values ≃ 0)
- Aerobic conditions: normal distribution

→ Dichotomous (binary) variable



Complexity reduction

- Reduce the large number of predictors
- Some variables perfectly correlated (Pearson correlation coefficient of 1)
- Hierarchical clustering to identify groups of highly correlated variables



Stepwise regression

- To produce ‘explanatory’ results (instead of black box)
- Stepwise – terms added or removed.

Method	R-squared value
Single conditions	0.91
Paired conditions	0.83
Paired conditions & dichotomous variable	0.976

Machine learning

- Random forest:
  ensemble over decision trees
- 10-fold cross validation
- 2^{14} (16,384) models

- Adjusted R-squared (higher better)
- AIC - Akaike's information criterion (lower better)
- BIC – Bayesian AIC (lower better)
- P-values - highly statistically significant <0.001
- K-Fold Cross validation mean squared error (lower better)

Formula	Adj_R_Sq	AIC	BIC	Model p-value	10 fold CV MSE
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + M_adp_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.830	155.3	185.0	2.74E-50	0.182
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.832	154.3	184.1	1.78E-50	0.178
FBA_Bio ~ M_o2_c + R_NADTRHD + M_h_b + M_glc_D_e + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.831	155.4	188.1	1.13E-49	0.181
FBA_Bio ~ M_o2_c + R_NADTRHD + M_pi_c + M_glc_D_e + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.831	155.6	188.3	1.22E-49	0.183
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + re_R_G6PDH2r + M_adp_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.831	156.1	188.8	1.53E-49	0.178
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + re_R_G6PDH2r + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.830	156.2	188.9	1.63E-49	0.174
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + M_adp_c + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.833	154.0	186.8	6.06E-50	0.180
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_EX_glu_L_e + R_AKGt2r	0.831	155.9	188.6	1.41E-49	0.178
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_GLNS + R_AKGt2r	0.830	156.3	189.0	1.69E-49	0.180
FBA_Bio ~ M_o2_c + R_NADTRHD + M_h_b + M_glc_D_e + re_R_G6PDH2r + M_adp_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.831	156.2	192.0	6.13E-49	0.180
FBA_Bio ~ M_o2_c + R_NADTRHD + M_h_b + M_glc_D_e + M_adp_c + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.834	154.4	190.1	2.61E-49	0.182
FBA_Bio ~ M_o2_c + R_NADTRHD + M_pi_c + M_glc_D_e + re_R_G6PDH2r + M_adp_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.834	154.0	189.7	2.17E-49	0.181
FBA_Bio ~ M_o2_c + R_NADTRHD + M_pi_c + M_glc_D_e + M_adp_c + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.836	152.1	187.9	9.46E-50	0.180
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + re_R_G6PDH2r + M_adp_c + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_AKGt2r	0.832	155.9	191.6	5.20E-49	0.177
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + M_adp_c + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_EX_glu_L_e + R_AKGt2r	0.833	155.3	191.0	3.92E-49	0.181
FBA_Bio ~ M_o2_c + R_NADTRHD + M_glc_D_e + M_adp_c + M_13dpg_c + R_EX_fru_e + M_h2o_e + R_GLUN + R_GLNS + R_AKGt2r	0.832	155.8	191.5	5.02E-49	0.182

Combining stepwise regression & machine learning

- Added 12 predictors from top machine learning models which were absent in regression model
- Repeated stepwise regression
- Adjusted r-squared value improved from 0.976 to 0.979

Derived dynamic Biomass proxy

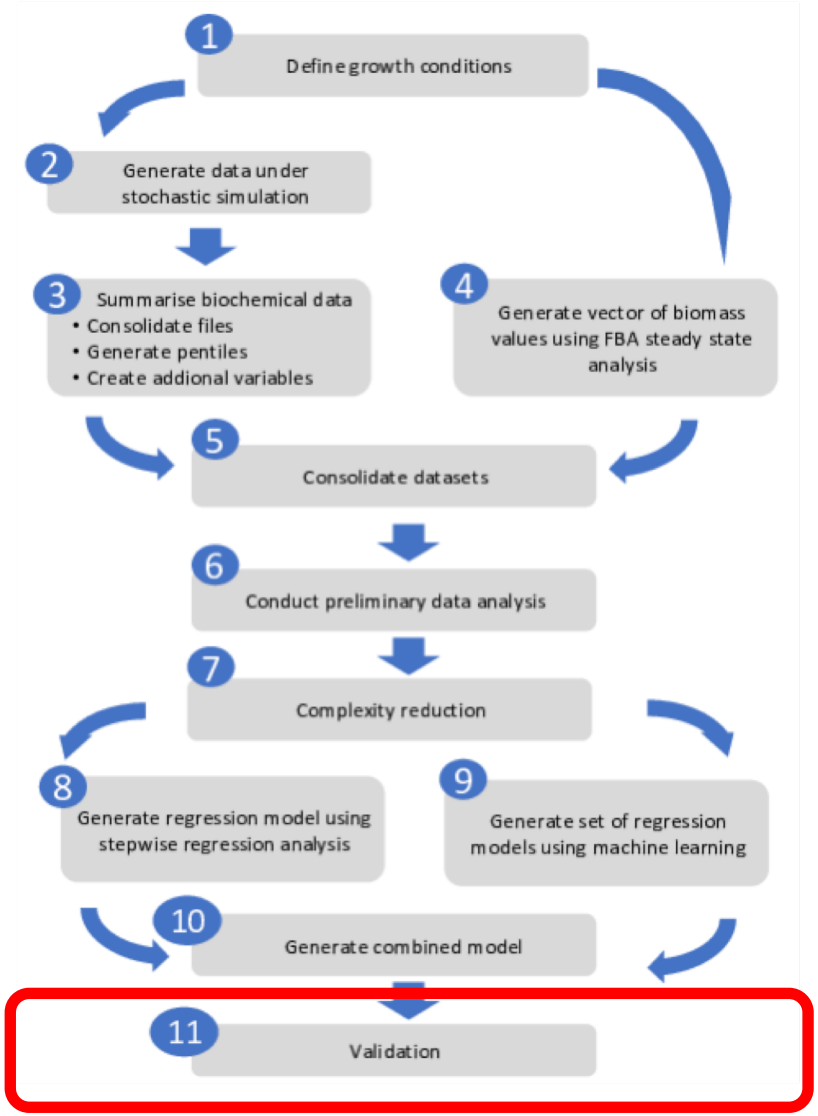
$$
\begin{aligned}
Biomass \approx\ & \\
& -14.2113 \\
& +2.1133 \cdot M\_fru\_b + 2.1744 \cdot M\_glc\_D\_b + 4.5078 \cdot M\_o2\_b \\
& +13.4913 \cdot R\_GLUN \\
& +Aerobic\,(0.7191 \cdot Pair - 0.1056 \cdot M\_h\_b \\
& \quad +1.8578 \cdot M\_fru\_b + 1.8466 \cdot M\_glc\_D\_b - 3.4306 \cdot M\_o2\_c \\
& \quad +0.8033 \cdot R\_RPI - 3.5964 \cdot R\_SUCOAS\_FwRe).
\end{aligned}
$$

Incorporates reactions rates as well as metabolite concentrations (unlike FBA biomass)

Workflow of key analytical steps

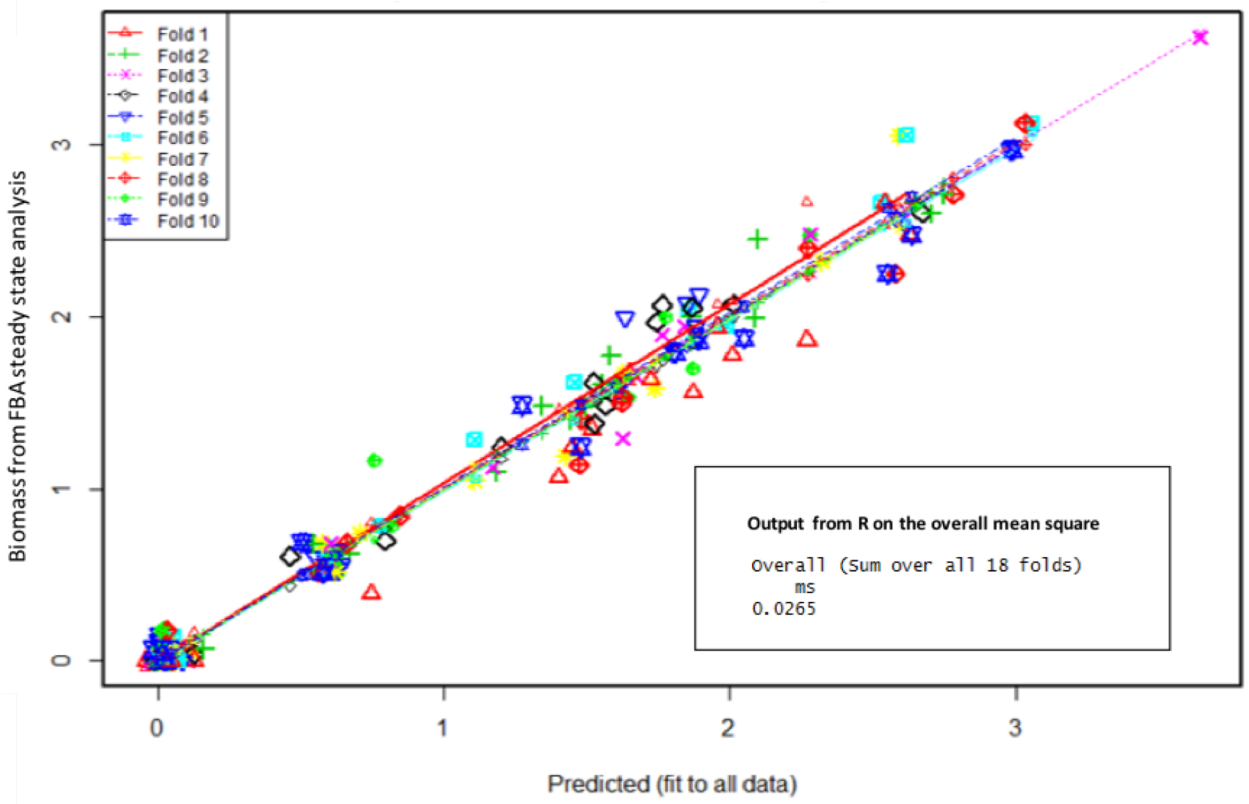
Results validation

- 10-fold Cross-validation



10-fold Cross-validation output for the final regression model

- Small symbols: predicted values
- Large symbols: actuals
- The 10 dashed lines relate to best fit line for the respective folds.



Unexpected results!

Comparing the sum of aerobic single conditions with pair of conditions, part 1

Condition 1 Name	Biomass	Condition 2 Name	Biomass	Condition 1+2	Paired value	Biomass Total	Increase %
Ethanol	0.70	Glutamine	1.16	1.86	1.97	0.104	5.6%
Ethanol	0.70	Fumarate	0.79	1.49	1.58	0.097	6.5%
Ethanol	0.70	Malate	0.79	1.49	1.58	0.097	6.5%
Fructose	1.79	Glutamine	1.16	2.95	3.05	0.096	3.2%
Glucose	1.79	Glutamine	1.16	2.95	3.05	0.096	3.2%
Ethanol	0.70	Glutamate	1.24	1.94	2.04	0.096	4.9%
Glutamine	1.16	Lactate	0.74	1.90	2.00	0.092	4.9%
Ethanol	0.70	Auccinate	0.84	1.54	1.63	0.092	5.9%
Acetaldehyde	0.61	Glutamine	1.16	1.77	1.86	0.091	5.1%

Comparing the sum of aerobic single conditions with pair of conditions, part 2

Condition 1 Name	Biomass	Condition 2 Name	Biomass	Condition 1+2	Paired value	Biomass Total	Increase %
Fructose	1.79	Glutamate	1.24	3.03	3.12	0.090	3.0%
Glucose	1.79	Glutamate	1.24	3.03	3.12	0.090	3.0%
Acetaldehyde	0.61	Fumarate	0.79	1.39	1.48	0.090	6.5%
Acetaldehyde	0.61	Malate	0.79	1.39	1.48	0.090	6.5%
Acetaldehyde	0.61	Glutamate	1.24	1.85	1.94	0.090	4.8%
Acetaldehyde	0.61	Auccinate	0.84	1.45	1.53	0.087	6.0%
Glutamate	1.24	Lactate	0.74	1.98	2.07	0.085	4.3%
Ethanol	0.70	Fructose	1.79	2.49	2.57	0.083	3.3%
Ethanol	0.70	Glucose	1.79	2.49	2.57	0.083	3.3%
Fructose	1.79	fumarate	0.79	2.58	2.66	0.083	3.2%
Fructose	1.79	Malate	0.79	2.58	2.66	0.083	3.2%

Acetaldehyde rescues anaerobic conditions

Acetaldehyde paired with	FBA value for paired condition
Fumarate	0.145
Malate	0.145
Lactate	0.117
2-oxoglutarate	0.068
Glutamate	0.045
Glutamine	0.040
All other conditions	< 0.01

Pairing conditions - new insights

- Improved predictive power of the model

- Some pairs have biomass values greater than the 2 single conditions

Acetaldehyde

- does not produce biomass anaerobically as a single condition
- produces biomass when paired with a number of other conditions that do not produce biomass anaerobically → rescues them

Conclusions

- Biomass function proxy developed for reduced GEM of E.coli
- ‘Gold standard’ data from FBA analysis
- Complexity addressed by correlation & clustering analysis
- Pairing growth conditions improved predictive power and gave interesting insights
- Acetaldehyde ‘rescues’ other non-productive anaerobic conditions

Tools used

- **Cobra** :
  FBA
- **Snoopy** :
  SBML -> stochastic Petri nets
- **Marcie** :
  δ-leaping simulation of stochastic Petri nets
- **R** :
  data analytics, packages used: Mass, Boruta, car, ClustOfVar

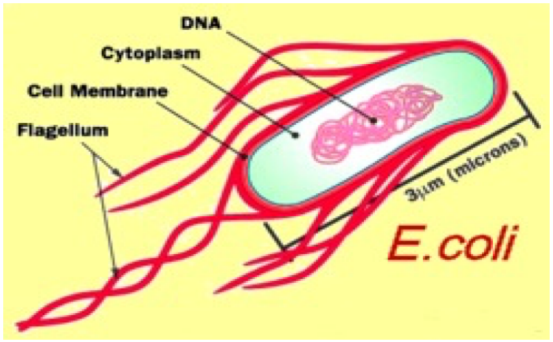
Acknowledgements

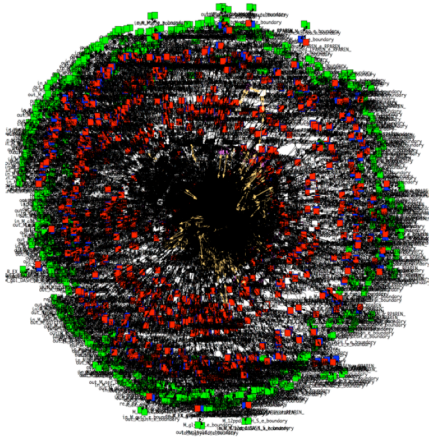
@Brunel University London

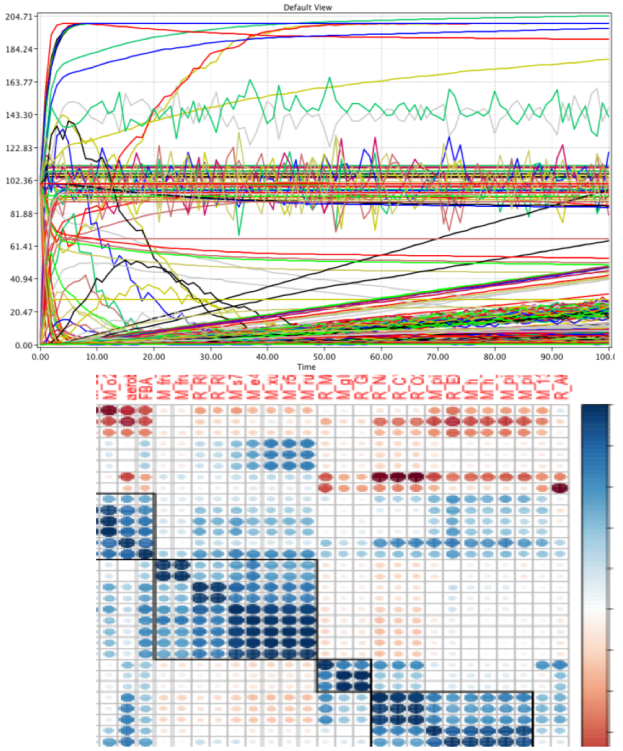
- Bello Suleiman – FBA data

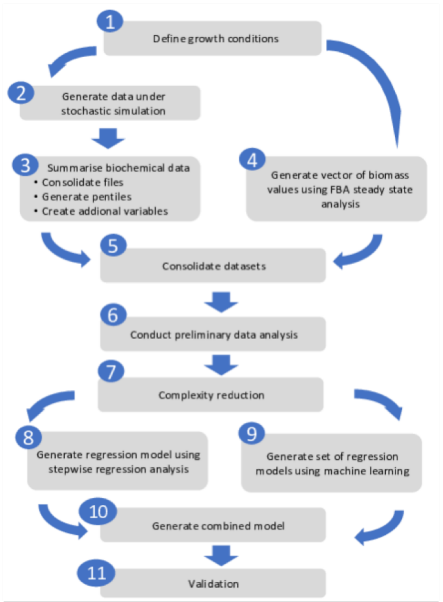
- Alessandro Pandini – data analytics / R

Self, Gilbert, Heiner, CMSB 2018









Questions?

David Gilbert & Monika Heiner
david.gilbert@brunel.ac.uk monika.heiner@b-tu.de

Supplementary materials:
www-dssz.informatik.tu-cottbus.de/DSSZ/Software/Examples