PeerJ Preprints NOT PEER-REVIEWED

Personalised models for human – gut microbiota interaction (extended abstract)

David Gilbert[1], Monika Heiner[2], Leila Ghanbar[1]

[1]Brunel University London, UK
[2]Brandenburg Technical University, Cottbus, Germany

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

Introduction

It is now becoming feasible to determine the composition of an individual gut microbiota (gut microflora), as well as the individual genome.

In addition, whole genome scale metabolic models (GEMs) exist for a range of bacteria, and also for human. These models have typically been developed for Flux Balance Analysis (FBA) and are constraint-based, lacking any quantitative data (concentrations and kinetic information), and are often pretty large. For example, the K-12 strain of *E. coli (Escherichia coli)* has over 4000 genes, of which some 1400 are involved in metabolism [13]. When taking into account the compartmental structure of the organism (periplasm, cytosol, extracellular space), this results in a model comprising around 3000 reactions, and about 1200 unique metabolites yield about 2300 metabolites respecting the compartmental structure. The BiGG database [11] contains 80 bacterial GEMs including 51 strains of *E. coli*, 8 strains of *(Shigella)* and models for 15 other bacteria species, plus mouse and human (RECON1). The Biomodels database [10] also contains several GEMs, including RECON1 and RECON2 for human.

In principle this enables us to build models for gut microbiota by aggregating strain-specific models and also place this within the human context, and to make predictions on a personalised basis of the influence of gut microbiota on human metabolism, and how the interactions between these microbiota and also the human may evolve. Such aggregation, however, raises several challenges including:

- How to realise such aggregated models?
- Can we model the interactions between members of an aggregated model?
- What does such interaction mean and entail? For example, do these bacteria communicate (e.g. by quorum sensing), or compete for resources?

Coping with these challenges will enable us to contribute to the personalised medicine by generating aggregated models for individual or sub-populations of patients, exploiting the data characterising gut microbiota population composition for individuals whose individual genomes can also be determined. This will yield a personalised human GEM together with a personalised gut microbiota profile.

In addition, there is a need to derive dynamic models from constraint-based models, because these do not naturally permit the composition of interacting non-homogeneous component models. Furthermore, the dynamic approach enables the analysis of transient as well as steady state behaviour.

One strand of our current research is to develop methodologies to derive dynamic GEMs from their constraint-based counterparts. We have also implemented a suite of modelling and analysis tools which permit the exploration of the dynamic behaviour of the very large metabolic models which could be applied to populations of models in the context human – gut microbiota. Our approach permits the description of the dynamic behavioural interaction between different bacterial strains and their human host on a personalised level within one aggregated model. We exploit model checking techniques to analyse the huge amounts of data generated by the dynamic simulation of these very large and complex models.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3267v2 | CC BY 4.0 Open Access | rec: 26 Sep 2017, publ: 26 Sep 2017

PeerJ Preprints NOT PEER-REVIEWED

Methods

Modelling. Our modelling approach builds on the following components: *Petri nets* generated out of SBML; *Quantitative Petri nets* (dynamic – deterministic or stochastic kinetics); and *Coloured Petri nets* (aggregated models; structured colours, . . .).

The models that we consider in this project comprise networks of biochemical reactions, which are often exchanged via the Systems Biology Markup Language (SBML) [9]. We use a subset of the information contained in these models comprising compartments, metabolites including boundary conditions and reactions. Kinetic information is not included in these original GEMs.

Such biochemical reaction networks can be considered as classical bipartite graphs, with the two distinct types of nodes representing reactions and metabolites as transitions and places, respectively. Hence they can be immediately encoded as Petri nets, and thus their analysis can benefit from the rich set of Petri net techniques and tools, covering both qualitative and (stochastic and deterministic) quantitative aspects [4].

Coloured Petri nets offer a modelling concept entirely orthogonal to the engineering principles of compositionality and hierarchical structuring. They can be considered as a short-hand notation of (plain) Petri nets. The reduction in the visible model size is achieved by folding similar net components into one component, which are then technically distinguished by coloured tokens (instead of simply having black tokens). This just requires to borrow a few concepts from standard high-level programming languages. Hierarchical information can be encoded using colour tuples which in effect describe the branches from root to leaves of a hierarchical tree. In order to model the gut microbiota we exploit this hierarchical facility to represent species, strain and substrain relationships, which have a common component plus sets of differences. Copy numbers of individual strains are given by the number of corresponding coloured tokens. Indeed we can include the human GEM in the set of coloured variants. The personalised human aspect would be colours for the stratification of patients into types of sub-populations.

Coloured Petri nets can always be automatically unfolded into the underlying (plain) Petri nets, as long as we confine ourselves to finite colour sets. In return for this concession, we can do everything with coloured Petri nets what we are able to to with (plain) Petri nets (animation, simulation, analysis).

The folding of Petri nets into coloured Petri nets requires a partitioning of the set of places and transitions. Each partition of (uncoloured) places makes a coloured place, and the uncoloured places become its colours; likewise for transitions. Thus, whether the folding can be done automatically in reasonable time depends on the effort one has to spend to identify the partitioning. Despite the complexity of the general subgraph isomorphism problem, it is often feasible for special cases [12]. We are able to automatically fold the *E. coli* models to give a coloured (aggregated) model, due to the consistent naming convention in the BiGG database models, because they have all been derived from one ancestor model *E. coli* K-12.

Analysis. GEMs typically have an infinite state space, which precludes the use of exact analysis methods which build on an exhaustive description of the state space [7]. An obvious choice is thus dynamic simulation, i.e. the generation of a representative (finite) set of finite traces through the infinite state space. Our tool suite enables us to perform either stochastic, deterministic or hybrid simulation based on the same model.

The time series output generated by the simulators comprise many individual traces of metabolites or reactions, which cannot be easily evaluated by eye. Model checking permits us to determine if a model fulfils given properties specified in temporal logics, e.g. probabilistic linear-time logic PLTL [2]. We use simulative model checking over time series traces of metabolite and reaction behaviours, exploiting libraries of typical property patterns which can be automatically applied to very large sets of traces. In addition there is a hierarchical structure over sets of traces which is implied by the colour tuples; this enables us to make behavioural analyses disentangling components of this hierarchy. An advantage of simulative model checking on coloured traces is that we can relate the predicted behaviour generated by these complex models with real-life observations of bacterial mixtures.

We can observe individual reactions at the unfolded level, or coloured reactions giving the differential activity of different species and strains, for example detecting when one strain is dead or dying by monitoring the biomass reaction. Likewise we can observe individual metabolites at the unfolded or coloured levels, the latter giving us the ability to dissect admixtures.

PeerJ Preprints NOT PEER-REVIEWED

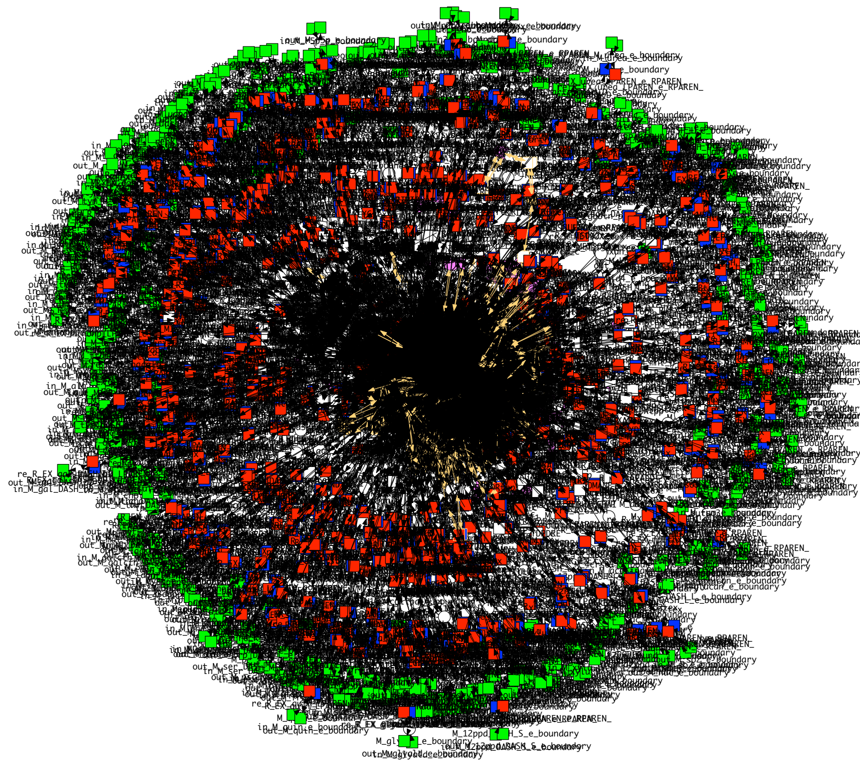

Figure 1: Petri net representation of an *E. coli* K-12 genome scale metabolic model (GEM) from [13]; layout generated with Snoopy [5]. Colour code: green: generated boundary transitions, blue: reversible reactions, red: generated reverse direction for reversible reactions, yellow: P-invariants.

Results

In this paper we present techniques and supporting tools which permit the development of personlised models for human – gut microbiota interaction. As a proof of principle we have developed an initial coloured quantitative Petri net model of quorum-sensing driven biofilm formation in *E. coli*.

The construction of such models is supported by a suite of modelling and analysis tools which permit the exploration of the dynamic behaviour of the very large metabolic models [3], comprising Snoopy [5], Charlie [8], Prolog [1], MC2 [2], and Marcie [6]. In the following we sketch the core functionalities of each tool employed, and their interrelation.

Snoopy is a platform-independent Petri net editor and simulator, reading and writing – among others – the exchange formats SBML and CANDL (Coloured Abstract Net Description Language) [6]. Reading SBML may involve the generation of input and output transitions for all boundary conditions, and the generation of reverse transitions for reversible reactions. Snoopy also reads some of Charlie's result files, which helps to visualise subnets induced by P-invariants or (bad) siphons, and to initialise the places belonging to a P-invariant for simulation. Snoopy supports a variety of simulation algorithms, including approximative and exact stochastic simulation, and permits the visualisation of simulation traces within user-defined result viewers.

Charlie is a Java tool to analyse place/transition Petri nets and comes with a graphical and command line interface. It supports standard static and dynamic analysis techniques, including elementary structural analysis, input/output places/transitions, connectivity, P/T-invariants, (minimal) siphons/traps, and STP. The computation of siphons/traps is done in a combinatorial manner, which is only feasible for smaller Petri nets. Likewise, the Petri nets considered in this project are expected to be covered with T-invariants due to the steady state assumption, which can be shown with Charlie's implementation only for smaller nets. But Charlie does compute all P-invariants for our networks.

Prolog: We use GNU Prolog as the platform to store the network graph, perform some basic analysis

of the graph exploiting Prolog's search and pattern matching facilities due to its efficient implementation of backtracking and unification over terms, and also to edit the graph. Conversion routines were written to transform Snoopy's ANDL to Prolog and vice-versa. The overall editing environment was written in Prolog as well, which calls the other tools.

Model checking: In order to check the behaviours of the nets we use two simulative model checkers which employ Probabilistic Linear Temporal Logic with constraints. *Marcie* is able to generate simulation traces by a variety of methods, and can perform in-line model checking over individual traces. *MC2* is an off-line Monte Carlo Model Checker which can check individual and averaged traces, which can in particular be coloured traces.

Conclusions

Our tools could be applied to populations of models in the context of human - gut microbiota interactions. Our approach that we have developed permits the description of the dynamic behavioural interaction between different bacterial strains and their human host on a personalised level within one aggregated model represented as a coloured Petri net. We use simulative model checking techniques over coloured traces to analyse the huge amounts of data generated by the dynamic simulation of these very large and hierarchically structured models.

Outlook — we can incorporate into our methodology the ways in which the gut microbiota of an individual human adapt to environmental changes by genetic mutation.

References

[1] Diaz, D., Codognet, P.: The GNU prolog system and its implementation. In: Proc. 2000 ACM symposium on Applied computing-Volume 2. pp. 728–732. ACM (2000)

[2] Donaldson, R., Gilbert, D.: A model checking approach to the parameter estimation of biochemical pathways. In: Proc. CMSB. pp. 269–287. LNCS 5307, Springer (2008)

[3] Gilbert, D., Heiner, M., Jayaweera, Y., Rohr, C.: Towards dynamic genome scale models. Briefings in Bioinformatics, accepted for publication (2017)

[4] Heiner, M., Gilbert, D., Donaldson, R.: Petri nets in systems and synthetic biology. In: Schools on Formal Methods (SFM). pp. 215–264. LNCS 5016, Springer (2008)

[5] Heiner, M., Herajy, M., Liu, F., Rohr, C., Schwarick, M.: Snoopy – a unifying Petri net tool. In: Proc. Petri Nets. LNCS, vol. 7347, pp. 398–407. Springer (2012)

[6] Heiner, M., Rohr, C., Schwarick, M.: MARCIE - Model checking And Reachability analysis done effiCIEntly. In: Proc. Petri Nets. LNCS, vol. 7927, pp. 389–399. Springer (2013)

[7] Heiner, M., Rohr, C., Schwarick, M., Streif, S.: A Comparative Study of Stochastic Analysis Techniques. In: Proc. CMSB. pp. 96–106. ACM digital library 978-1-4503-0068-1/10/09 (2010)

[8] Heiner, M., Schwarick, M., Wegener, J.: Charlie – an extensible Petri net analysis tool. In: Proc. Petri Nets. LNCS, vol. 9115, pp. 200–211. Springer (June 2015)

[9] Hucka, M., Finney, A., Sauro, H., Bolouri, H., Doyle, J., et al., H.K.: The Systems Biology Markup Language (SBML): A Medium for Representation and Exchange of Biochemical Network Models. J. Bioinformatics 19, 524–531 (2003)

[10] Juty, N., Ali, R., Glont, M., Keating, S., Rodriguez, N., Swat, M.J., Wimalaratne, S.M., Hermjakob, H., Le Novère, N., Laibe, C., Chelliah, V.: BioModels: Content, Features, Functionality and Use. CPT: Pharmacometrics & Systems Pharmacology (2015)

[11] King, Z., Lu, J., Dräger, A., Miller, P., Federowicz, S., Lerman, J., Ebrahim, A., Palsson, B., Lewis, N.: BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. Nucleic acids research 44(D1), D515–D522 (2016)

[12] Liu, F., Heiner, M., Yang, M.: Representing Network Reconstruction Solutions with Colored Petri Nets. Neurocomputing 174, Part A, 483–493 (January 2016), http://www.sciencedirect.com/science/article/pii/S0925231215012473

[13] Monk, J., Charusanti, P., Azizb, R., Lermand, J., Premyodhinb, N., Orth, J., Feist, A., Palsson, B.: Genome-scale metabolic reconstructions of multiple *Escherichia coli* strains highlight strain-specific adaptations to nutritional environments. PNAS 110(50), 20338–20343 (2013)