Mikrobiokosmos 2021

Intro Slide

Good evening everyone!

Many thanks to Mikrobiokosmos for this great conference and for the chance to share with you some piece of our research.

Slide 1: Stoichiometric matrix

A genome-scale metabolic model (GEM) is a mathematical representation of the metabolism of an organism and is strongly related to its corresponding stoichiometric matrix (S).

S is a matrix with the **metabolites** of the cell as rows and its **reactions** as columns meaning the values of a column are the **stoichiometric coefficients** of each metabolite in the corresponding reaction.

Once the model is built, the question is **how much** a reaction occurs under certain circumstances.

The rate of turnover of molecules through a reaction is called **flux**.

The stoichiometric relationshipns impose a series of **mass balance constraints**.

(meaning you cannot use a metabolite more than it is actually available)

while the lower and upper bounds of each flux impose certain **capacity constraints**.

When applied to the network these constraints define its **allowable solution space**.

Constraint based modeling, allows us to calculate the fluxes of the different reactions of a model.

The widely known Flux Balance Analysis that has been proven extremely useful, is strongly dependent on an objective function and that is not a straight-forward task. But most importantly, it returns a **sole flux vector** out of infinite.

Slide 2: Sampling definition

So what if we could have the distribution of the flux values of each reaction instead of a single value without the need of an objective function?

If we sample a sufficiently large number of uniformly distributed points, we can generate the probability distribution of each flux.

In flux sampling can be used **both** with and without an objective function.

Slide 3 - MMCS

The geometic representation of the solution space is called **polytope**.

As we move to models with more reactions and therefore to polytopes of higher dimension, sampling gets challenging.

Polytopes, derived from GEMs are usually rather skinny making uniform sampling even harder.

To address this challenge, we developed a Multiphase Monte Carlo Sampling algorithm which runs at phases, transforming the polytope from phase to phase.

At each phase, we sample several points using the current polytope.

We perform a rounding step on the current polytope to obtain the polytope of the next phase.

To this end, we compute a linear transformation, that puts the sample we just built into isotropic position and then we apply this transformation on the current polytope.

Once the algorithm converges, the samples found are mapped back in the initial polytope.

Slide 4: Renz et al.

In 2020, Renz et al. built the biomass function of SARS-CoV-2 and they integrated it, into the metabolic model of a human alveolar macrophage, along with a host biomass maintenance function.

Using FBA they optimized their model first for the host biomass maintenance function and then for the virus biomass function.

This way they were able to find that the GK1 reaction is a potential antiviral target.

Slide 5: MMCS on the Renz et al. model

To demonstrate the potential of flux sampling, we used this model and ran our MMCS algorithm:
a. first by maximizing the maintenance function b. and then by maximizing the virus biomass function

As you see the flux distribution of Tyramine Sulfotransferase, is exactly the same in both cases, while in case of GK1, the flux increases dramaticcaly when the virus biomass function is maximized.

Slide 6: Further applications of flux sampling

The potential applications of flux sampling are numerous!

Preparing this presentation I found out about a "fragrant" (μοιρωράδτος) study that was published only a couple of months ago, where flux sampling was used in developing or selecting optimal aroma-producing yeast stains for winemaking.

My future plans are to focus on microbial communities and using flux sampling on such models, investigate microbial interactions.

Slide 7: `dingo` library

Our MMCS algorithm, initially written in C++, is now available as a Python library and you may find a tutorial on how to use it either throught its GitHub repo or this GCollab notebook.

Thank you slide

That's all from me. Thank you for your attention and best wishes for the rest of the conference.