13C isotopomer metabolic modeling: automatic generation of the mathematical model

J. Valette[1], A. Shestov[1], K. Ugurbil[1], and P-G. Henry[1]

[1]CMRR, University of Minnesota, Minneapolis, MN, United States

Introduction

Carbon-13 MRS and metabolic modeling provide unique insight into brain metabolism. Recently a new model has been proposed to include the information from ^{13}C isotopomers in neuron-glia metabolic models of the brain [1]. Such models may comprise hundreds of equations, the handwriting of which is extremely time consuming and prone to errors. However, the isotopomer formalism can be simplified using a related concept dubbed "probabimer", which retains only the minimal information necessary to describe the fine structure of NMR spectra. In this context, the present work proposes an algorithm implemented in Matlab allowing the automatic writing of probabimer differential equations either in text format, in which the names of probabimers appear explicitly and can read by a human operator, or in Matlab format that can be directly interpreted by Matlab differential equation solvers. Moreover, the system of equations is automatically reduced for any given NMR experiment (defined by the infused substrates and the detected carbons) in order to retain a minimal amount of equations. The program is described and exemplified here for the neuron-glia two-compartment model.

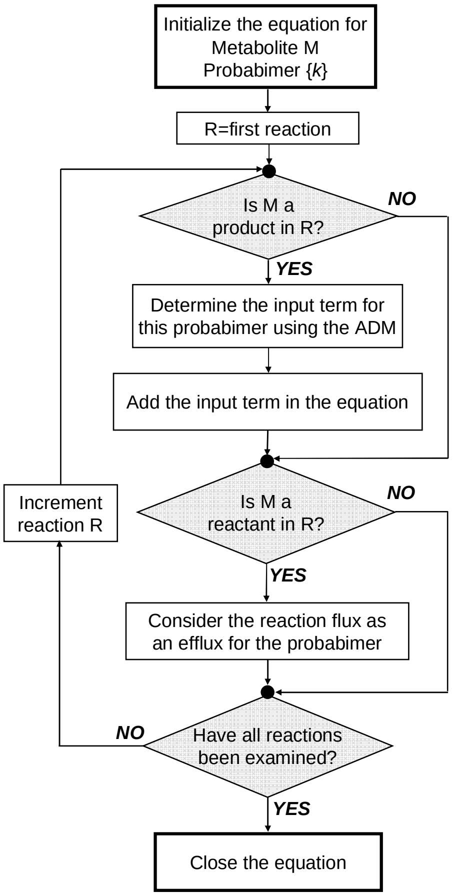
Initialize the equation for Metabolite M Probabimer {k}
R=first reaction
Is M a product in R?
NO
YES
Determine the input term for this probabimer using the ADM
Add the input term in the equation
Is M a reactant in R?
NO
YES
Consider the reaction flux as an efflux for the probabimer
Increment reaction R
Have all reactions been examined?
NO
YES
Close the equation

Figure 1. *Schematic of the algorithm analyzing the biochemical network and writing the system of differential equations.*

Methods

Background A probabimer (or cumulative isotopomer) for a metabolite M is a function $\pi_{M\{k\}}$, where $\{k\}$ is a list of indexes representing carbon positions in the carbon chain of M. The number of elements in $\{k\}$ is the *order* of the probabimer. By definition, $\pi_{M\{k\}}$ is the probability that a random M molecule is labeled at least at the carbon positions listed in $\{k\}$, whatever the label at other positions. $\pi_{M\{k\}}$ is related to the isotopomers of M accordingly: for example, if M has 4 carbons, $\pi_{M\{1,2\}}=[M_{\{1,2\}}]/[M]+[M_{\{1,2,3\}}]/[M]+[M_{\{1,2,4\}}]/[M]+[M_{\{1,2,3,4\}}]/[M]$. The fine structure of ^{13}C spectra *in vivo* is minimally though completely described by bounded probabimers (i.e. probabimers for which the elements of $\{k\}$ are consecutive) of order ≤ 3. Higher order and non-bounded probabimers are not detected by NMR since long-range ^{13}C-^{13}C scalar coupling constants are small compared to the line width *in vivo*. As an example, the intensity of the C2 doublet corresponding to the ^{13}C1-^{13}C2 coupling is proportional to $\pi_{M\{1,2\}}-\pi_{M\{1,2,3\}}$.

Biochemical network representation A biochemical reaction is represented by a flux value, the reactants, the products, and an atom distribution matrix (ADM) [2]. Molecular symmetries resulting in label scrambling for metabolites such as succinate are automatically taken into account using permutation matrices.

General analysis algorithm The aim of the algorithm is to identify all label inputs and outputs for all probabimers of order ≤ 3. For each metabolites and each probabimers, all reactions are examined according to fig. 1. Based on this algorithm, writing the full set of differential equations is done sequentially by concatenation of character strings representing the input and output terms determined above.

Reducing the probabimer system for a given ^{13}C NMR experiment In the context of NMR spectroscopy, where only a few metabolites are detected and where label is injected in the network *via* a given set of labeled substrates, the following two kinds of probabimers can be excluded from the analysis:

- Probabimers that are not implied in the time-evolution of the measured ^{13}C NMR fine structure.
- Probabimers that remain constant throughout time, given the infused labeled substrates.

Results

The above-described algorithm was used to analyze the classical neuron-glia biochemical network [3]. The running time to write the full set of equations (total 263 equations) was ~10 sec on a work station (CPU Intel Xeon 5160 @ 3.00GHz, 16 GB RAM, Matlab 7.1 under Red Hat). Determination of the required probabimers when observing the C2, C3 and C4 resonances of glutamate and glutamine, and when infusing [1,6-^{13}C]glucose, was achieved in ~2 sec. Finally, writing the reduced set of equations (total 117 equations) was performed in ~4 sec. As an example of the two different output files (explicit and Matlab notation), the automatically generated equations are given below for $\pi_{OGn\{3,4\}}$ (OGn: neuronal 2-oxoglutarate; GLUn: neuronal glutamate; AcCoAn: neuronal acetyl-CoA; Vtca: neuronal TCA cycle; Vx: exchange between 2OGn and GLUn):

*OGn*dpi_OGn_34/dt = Vtca*pi_OAAn_2*pi_AcCoAn_2 + Vx*pi_GLUn_34 - (Vtca+Vx)*pi_OGn_34*
yp(38)=1/OGn(Vtca*y(52)*y(112) + Vx*y(13) - (Vtca+Vx)*y(38))*

Below is an example of the equation generated for the probabimer $\pi_{SUCg\{1\}}$ of glial succinate, simplified for the infusion of [1,6-^{13}C]glucose (OGg: glial OG; OAAg: glial oxaloacetate; Vg: glial TCA cycle; Vfum_g: exchange between OAAg and glial succinate/fumarate). The probabimer $\pi_{OGg\{5\}}$, which is not formed during [1,6-^{13}C]glucose infusion, is automatically replaced by the natural ^{13}C abundance *E_nat*:

*SUCg*dpi_SUCg_1/dt = 1/2*(Vg*pi_OGg_2 + Vg*E_nat + Vfum_g*pi_OAAg_1 + Vfum_g*pi_OAAg_4) - (Vfum_g+Vg)*pi_SUCg_1*
yp(111)=1/SUCg(1/2*(Vg*y(81) + Vg*E_nat + Vfum_g*y(91) + Vfum_g*y(94)) - (Vfum_g+Vg)*y(111))*

Once the system of equations has been built, solving the full set of 263 equations for a given value of the fluxes takes approximately 0.3 s using the Matlab routine *ode15s* (system solved for a 200 minutes infusion with a 1 minute time-resolution), while solving the reduced system takes only ~0.07 s. This 4-fold acceleration associated with a reduced system size is particularly beneficial when extensive Monte-Carlo simulation has to be performed.

Conclusion

To the best of our knowledge, this work is the first report of a program allowing the automatic derivation, in text format, of differential equations describing ^{13}C enrichment time-courses, although different approaches have already been proposed to numerically build and solve the system of isotopomer equations at steady-state [4-7]. Here, the use of the probabimer formalism allows reducing the number of variables and equations compared to the full isotopomer formalism. Furthermore, the system of equations can be automatically reduced by eliminating probabimers which remain constant or are not related to signals actually measured by NMR. The fact that the equations are written simultaneously in a compact and formal way presents the two following advantages: *i)* The equations can be easily assessed by a human operator; *ii)* Once the network has been analyzed and the equations have been written, solving the system is efficiently done by Matlab routines (such as *ode15s*) for any value of the fluxes or concentrations formally appearing in the text equation file. The fast execution time of such routines is especially valuable for Monte-Carlo simulations, which are very demanding in computer resources.

[1] Shestov et al., *Proc ISMRM Berlin* (2007)
[2] Muzykantov et al., V.S. *React Kinet Catal Lett* (1980)
[3] Gruetter et al., *Am J Physiol* (2001)
[4] Marx et al., *Biotechnol Bioeng* (1996)
[5] Schmidt et al., *Biotechnol Bioen* (1997)
[6] Zupke et al., *Anal Biochem* (1997)
[7] Wiechert et al., *Metab Eng* (2001)

This work was supported by NIH grants R01NS3867, P41RR08079 and P30NS057091.