

A stochastic model for the evolution of metabolic networks with neighbor dependence

Aziz Mithani^{1,*}, Gail Preston² and Jotun Hein¹

Supplementary Material

S1 Time Reversibility of Neighbor Dependent Model

The time reversibility of the Markov process with neighbor dependence can be proved using Kolmogorov's criterion which states that *a positive continuous-time chain is reversible if and only if the product of the transition rates corresponding to any loop is equal to the product of the transition rates for the same loop traversed in the reverse direction* [Wolff, 1989]. As all networks are reachable from one another, it is trivial to see that the process is positive. Thus we need to prove

$$\gamma_{12}\gamma_{23}\cdots\gamma_{n1} = \gamma_{1n}\gamma_{n(n-1)}\cdots\gamma_{21} \quad (1)$$

where γ_{ij} is the rate of going from state i to j and is equal to the base rate, q_{ij} , from the independent edge model, times the neighborhood weight calculated using the network at state i , f_i , i.e.

$$\gamma_{ij} = q_{ij}f_i \quad (2)$$

Using (2) the Kolmogorov's criterion in (1) can be written as

$$q_{12}f_1q_{23}f_2\cdots q_{n1}f_n = q_{1n}f_1q_{n(n-1)}f_n\cdots q_{21}f_2$$

Rearranging the terms, we can write

$$q_{12}q_{23}\cdots q_{n1}f_1f_2\cdots f_n = q_{1n}q_{n(n-1)}\cdots q_{21}f_1f_2\cdots f_n.$$

Since the neighborhood weights are same on both sides we just need to prove that the following equality holds.

$$q_{12}q_{23}\cdots q_{n1} = q_{1n}q_{n(n-1)}\cdots q_{21} \quad (3)$$

In our model, only two types of events are possible: insertion and deletion of edges. It is, therefore, easy to see that starting from a network it requires an even number of events to return to the same network, as each extra edge which is inserted must be removed and an edge which is removed must be added later. In other words, (3) contains k pairs of complementary events where $k = n/2$. Thus, we can write the left hand side as

$$q_{12}q_{23}\cdots q_{n1} = (\lambda\mu)^k; \quad k = n/2.$$

Similarly, the right hand side can be written in the same form, i.e.

$$q_{1n}q_{n(n-1)}\cdots q_{21} = (\lambda\mu)^k; \quad k = n/2$$

completing the proof.

*to whom correspondence should be addressed: mithani@stats.ox.ac.uk,

¹Department of Statistics and ²Department of Plant Sciences, University of Oxford, Oxford, UK

S2 Example: Calculation of neighborhood component and rate matrix

We illustrate the calculation of neighborhood component $F(x_i, \Psi(x_i))$ and the rate matrix entry $\gamma(x'_i; x_i, \Psi(x_i))$ for the hyperedge labeled '7' for networks H_1 and H_2 shown in Figure 1 in the paper. First consider network H_1 . To calculate the neighborhood component, we need the number of neighboring hyperedges present/absent from the network. Hyperedge '7' has a total of 7 neighbors (hyperedges 3, 4, 5, 6, 8, 9 and 10) in the reference network, out of which 5 hyperedges (hyperedges 3, 5, 8, 9 and 10) are present in network H_1 . To calculate the neighborhood component, we also need the total number of hyperedges present in network H_1 . The number of hyperedges present in H_1 is 7 (hyperedge '7' is not taken into account). This results in the following neighborhood weights.

$$F_{H_1}(x_7, \Psi(x_7)) = \frac{5}{7} = 0.714.$$

The entry $\gamma(x'_7; x_7, \Psi(x_7))$ in the rate matrix Γ for the neighbor-dependent model is given as

$$\gamma(x'_7; x_7, \Psi(x_7)) = q(x_7, x'_7)F(x_7, \Psi(x_7))$$

where the entry $q(x_7, x'_7)$ corresponds to the appropriate entry in the rate matrix Q for the independent edge model. Using the rate parameters $(\lambda, \mu) = (0.05, 0.03)$, the matrix Q is given as

$$Q = \begin{bmatrix} -0.05 & 0.05 \\ 0.03 & -0.03 \end{bmatrix}.$$

Since the hyperedge '7' is present in network H_1 , we take the entry 0.03 from Q and multiply it by the neighborhood component $F_{H_1}(x_7, \Psi(x_7))$, i.e. the entry $\gamma(x'_7; x_7, \Psi(x_7))$ is calculated as follows.

$$\gamma(x'_7; x_7, \Psi(x_7)) = 0.03 \times 0.714 = 0.02142$$

Now consider network H_2 . Hyperedge '7' has 6 neighbors (hyperedges 3, 5, 6, 8, 9 and 10) present out of a total of 8 hyperedges present in the network. This results in the following neighborhood weight.

$$F_{H_2}(x_7, \Psi(x_7)) = \frac{6}{8} = 0.75$$

The hyperedge '7' is absent from network H_2 and, therefore, needs to be added to the network. Using the entry $q(x_7, x'_7) = 0.05$, the entry $\gamma(x'_7; x_7, \Psi(x_7))$ is given as

$$\gamma(x'_7; x_7, \Psi(x_7)) = 0.05 \times 0.75 = 0.0375.$$

S3 Equilibrium Probability

The stationary distribution π , also called the equilibrium distribution, of the Markov process can be calculated by using the following equation.

$$\pi Q = 0 \tag{4}$$

The i -th entry in vector π corresponds to the equilibrium probability of the i -th state in the system. In our case, this entry corresponds to the equilibrium probability of the network $P(H_i)$, i.e. the probability of observing the network H_i during the course of evolution. As mentioned earlier, when reaction neighborhood is taken into account the rate matrix is a $2^M \times 2^M$ matrix, where M is the total number of hyperedges in the system, making the explicit enumeration of the rate matrix impossible even for moderately sized networks. This in turn implies that (4) cannot be solved to obtain the equilibrium probabilities. An approximation is, therefore, required to obtain the equilibrium probabilities $P(H_i)$.

S3.1 Approximation of equilibrium probability

The equilibrium probability $P(H_i)$ of observing the network H_i can be approximated by dividing the network into smaller subnetworks and multiplying the equilibrium probabilities of the individual subnetworks. Let $P^*(H_i)$ denote the approximated value of the equilibrium probability of the network H_i . We write

$$P^*(H_i) = \prod_j P(H_i^j)$$

where $P(H_i^j)$ is the equilibrium probability of subnetwork H_i^j of the network H_i . The procedure to obtain the subnetworks H_i^j is outlined below.

- Choose the hyperedge with highest number of neighbors and consider its top $N - 1$ connected neighbors where N is the maximum allowed size of a subnetwork.
- If the number of selected edges is less than N then add the hyperedge with highest number of connections out of the remaining ones and consider its neighbors.
- Repeat the above steps until a subnetwork of size N is obtained.
- Remove the hyperedges used in the current subnetwork from the network and calculate the number of neighbors based on the remaining hyperedges in the network.
- Repeat the above procedure until the network is exhausted.

We used the above procedure to approximate the equilibrium probability of the toy network H_1 (shown in Figure 1 of the main paper) for different subnetwork sizes, and compared it against the equilibrium probability obtained by using the full network. The results are listed in Table 5.

S4 Path proposal

A path proposal can be generated from the current path by applying one of the following operations.

Add Events Given a path between two networks, we can obtain a new path by adding two events involving an edge at any two positions in the path. The addition of the events does not affect the final output of the path as the two events cancel the effects of each other. Formally, given a path z_k , two positions i, j and a hyperedge $e (e \in \mathcal{E}')$, a new path z'_k , where $k' = k + 2$ is given as

$$\text{addEvents}(z_k, i, j, e) = [e_1, \dots, e_{i-1}, e, e_i, \dots, e_{j-1}, e, e_j, \dots, e_k].$$

Delete Events Deletion of events from a path can be defined in a similar manner. However, we need to keep in mind that events can only be removed from a path if they are redundant. An event that occurs only once is not redundant and therefore cannot be removed. Given a path z_k such that $k > d$, where d is the number of differences between the two networks, and two positions i, j ($i \neq j$) such that $e_i = e_j$, a new path z'_k , where $k' = k - 2$ is given as

$$\text{deleteEvents}(z_k, i, j) = [e_1, \dots, e_{i-1}, e_{i+1}, \dots, e_{j-1}, e_{j+1}, \dots, e_k].$$

Permute Events Finally, we define an operation to permute events in a path. This operation is motivated by the fact that multiple paths of the same length from the starting network lead to the desired network. Formally, given a path z_k , a new path $z'_{k'}$ where $k' = k$ is given by permuting the events in the path z_k .

$$\text{permuteEvents}(z_k) = \text{permute}(e_1, \dots, e_k)$$

It must be noted that the new path length k' can take only following values.

$$k' = \begin{cases} k + 2 & \text{events are added} \\ k - 2 & \text{events are deleted} \\ k & \text{events are permuted} \end{cases}$$

Example Consider the following path consisting of 4 events from the network H_1 to the network H_2 (shown in Figure 1 in the main paper).

$$z_4 = [7, 4, 6, 4]$$

The above path consists of two redundant events relating to hyperedge 4 since during the course of evolution, hyperedge 4 is first added to the network and later removed, thus not affecting the final outcome. Using the operations defined above, the proposal for a new path can be generated. Following are some examples of the paths obtained by applying above operations.

Add Events $k' = k + 2$:

$$\begin{aligned} z'_6 &= [7, 4, \mathbf{7}, 6, \mathbf{7}, 4] \\ z'_6 &= [\mathbf{1}, 7, 4, 6, 4, \mathbf{1}] \\ z'_6 &= [\mathbf{4}, \mathbf{4}, 7, 4, 6, 4] \\ z'_6 &= \dots \end{aligned}$$

Delete Events $k' = k - 2$:

$$z'_2 = [7, 6]$$

Permute Events $k' = k$:

$$\begin{aligned} z'_4 &= [4, 6, 4, 7] \\ z'_4 &= [4, 4, 6, 7] \\ z'_4 &= \dots \end{aligned}$$

Note that only one path can be proposed by delete events whereas multiple paths are possible using add and permute operations in this case.

S5 Probability of selecting a path length

Let γ_a , γ_d and γ_p denote probabilities of selecting operations ‘add events’, ‘delete events’ and ‘permute events’ respectively. The probability of selecting a new path length is then given as follows.

- $k = 1$ (Only addition of events is allowed)

$$p(k'|k) = \begin{cases} 1 & k' = k + 2 \\ 0 & \text{otherwise} \end{cases}$$

- $k = d$ (Only addition and permutation of events is allowed)

$$p(k'|k) = \begin{cases} \gamma_a & k' = k + 2 \\ \gamma_p & k' = k \\ 0 & \text{otherwise} \end{cases} \quad (\gamma_a + \gamma_p = 1)$$

- $k > d$ (All three operations are allowed)

$$p(k'|k) = \begin{cases} \gamma_a & k' = k + 2 \\ \gamma_d & k' = k - 2 \\ \gamma_p & k' = k \\ 0 & \text{otherwise} \end{cases} \quad (\gamma_a + \gamma_d + \gamma_p = 1)$$

In the current analysis, all allowed operations are taken to be equally probable, i.e. when path length equals the number of differences between the two networks ($k = d$) the deletion of events is not allowed and each of the two allowed operations (`addEvents` and `permuteEvents`) has a probability of a half, and in the case when $k > d$ all three operations have a probability of one third each.

S6 Probability of proposing a new path

Assuming each path has equal chances of being proposed, the probability $q(z'_{k'}|z_k, k')$ of generating a new path, is given as

$$q(z'_{k'}|z_k, k') = \frac{1}{\Omega_{k'}}$$

where $\Omega_{k'}$ is the total number of possible paths of length k' and depends on the operation applied to the path. The number $\Omega_{k'}$ is calculated as follows.

S6.1 Events are added

First, consider the scenario when events are added. The total number of possible paths $\Omega_{k'}$ is given as

$$\Omega_{k'} = \sum_{e \in \mathcal{E}'} \binom{k - k_e + 2}{2}$$

where k_e is the number of times edge e is present in the path z_k , and \mathcal{E}' is the set of alterable hyperedges. The above expression can be understood as follows. There are three cases for addition of an edge. First, the edge may not be present in the path, i.e. $k_e = 0$. In this case, the first event relating to the edge can be added in $k + 1$ positions and the second event can be added in $k + 2$ positions. This results in $(k + 2)(k + 1)/2$ distinct paths. The second case is when the edge being added to the path is present once ($k_e = 1$). Assume that the edge is present at position i in the path. Now the first event can be added only in k positions because adding an event at $i - 1$ will lead to the same path as $i + 1$. Similarly, the second event relating to the edge can be added at $k + 1$ positions resulting in $(k + 1)(k)/2$ distinct paths. Finally, when an edge is present more than once in the current path then the two events can be added at $k - k_e + 1$ and $k - k_e + 2$ positions respectively resulting in $(k - k_e + 2)(k - k_e + 1)/2$ distinct paths.

S6.2 Events are deleted

The number of possible paths when events are deleted from the current path is given as

$$\Omega_{k'} = \sum_{e \in z_k} \binom{s_e^1}{2} + s_e^2$$

where s_e^i is the number of distinct stretches of length $\geq i$ of edge e in the path z_k . The first term in the above expression considers the cases when events are deleted from two different stretches in the path whereas the second term corresponds to the cases when two events are removed from the same stretch.

S6.3 Events are permuted

Finally, when events are permuted in the path z_k the total number of paths is given as

$$\Omega_{k'} = (k-1)! \sum_{e \in z_k} (k - k_e).$$

Events relating to edge e can be permuted to only those places which have different edges resulting in the factor $k - k_e$ whereas the remaining events can be permuted to any of the $k - 1$ positions.

S7 Performance of the MCMC for likelihood calculation

To test the performance of the MCMC for the estimation of likelihood value, we ran the path sampler with parameters $(\lambda t, \mu t) = (0.7, 0.9)$ on networks which differed from the starting network H_1 by 1 to 5 hyperedges. These networks are shown in Figure 1. The sampler was run for an increasing number of iterations and the likelihood value was averaged across 3 different runs. The results of the experiment are shown in Table 3. Also shown are the average CPU time taken in seconds and the acceptance percentage of the sampler. From the table it can be seen that the estimated value of the likelihood of the evolution of H_1 to H'_i conditioned on H_1 gets closer to the true likelihood value calculated using exponentiation of the rate matrix as the number of iterations are increased.

We further tested the sampler to see if the following condition was satisfied.

$$P(H_1)P(H_2|H_1) = P(H_2)P(H_1|H_2)$$

Here $P(H_i)$ is the equilibrium probability of the network H_i and was calculated by solving the equation

$$\pi \Gamma = 0$$

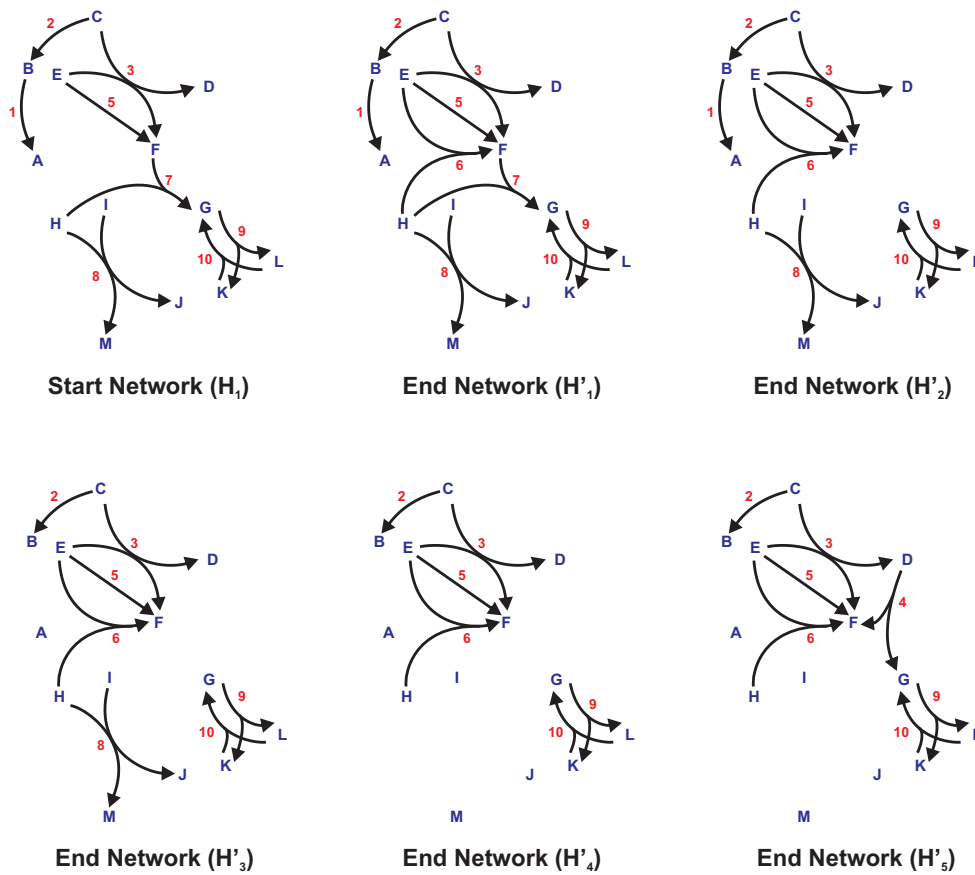
where Γ is the 1024×1024 rate matrix. The sampler was run for different parameter ratios and several parameter values. The results for the test are shown in Table 4. It can be seen from the table that the values $P(H_1)P(H_2|H_1)$ and $P(H_2)P(H_1|H_2)$ calculated by the sampler are very close to each other.

S8 Performance of the Gibbs sampler for parameter estimation

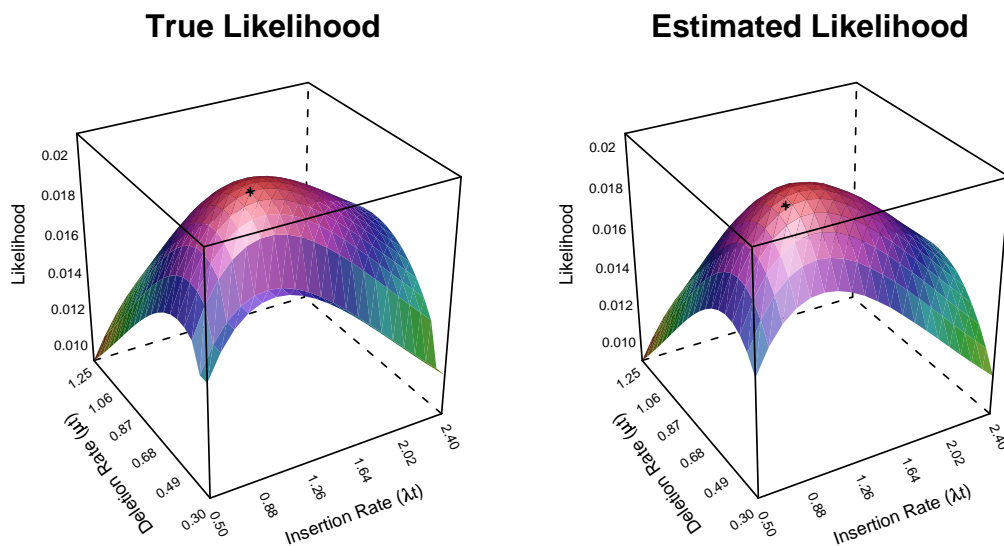
Starting from the toy network H_1 shown in Figure 1 of the main paper, we simulated network evolution with parameters $(\lambda t, \mu t) = (0.06, 0.03)$. At each step, the insertion and deletion rates were calculated based on the neighbor dependent model and were normalized to get probability values. An edge was then selected based on these probability values and was inserted if absent from the current network and deleted otherwise. A total of 50,000 iterations were used as burn-in period and a further 25,000 iterations were run to obtain networks from the stationary distribution. Start and end networks were sampled at every ten iterations and the Gibbs sampler was used to estimate the parameters. As the samples were taken from the stationary distribution, it was expected to get the ratio back instead of the actual parameter values since the behavior of the system in the equilibrium condition is driven by the ratio (λ/μ) and not the actual parameters. The density plot of the ratio (λ/μ) of estimated parameters for 2500 samples is shown in Figure 5. The posterior mean of the ratio of the estimated parameters was calculated as 2.17 compared to the true ratio of 2.0.

References

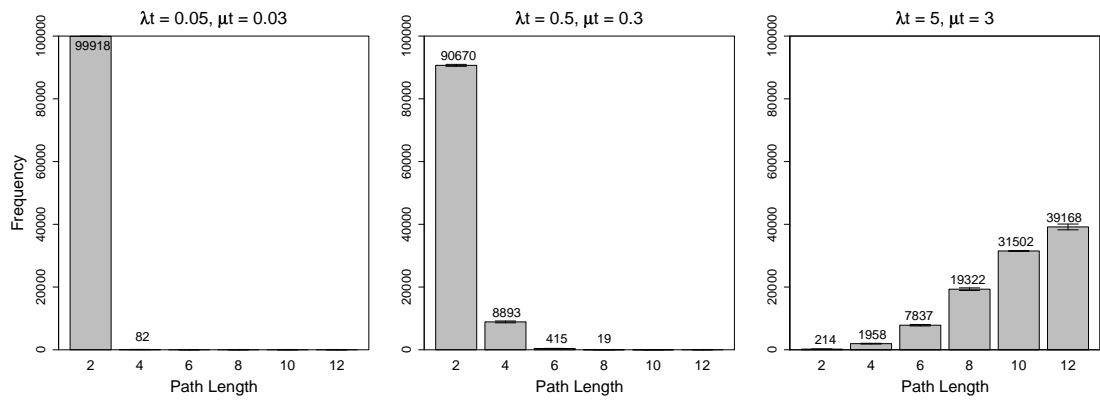
- [Kanehisa *et al.*, 2006] Kanehisa, M. *et al.* (2006) From genomics to chemical genomics: new developments in KEGG, *Nucleic Acids Res.*, **34**, D354–357.
- [Sarkar and Guttman, 2004] Sarkar, S.F., and Guttman, D.S. (2004) Evolution of the core genome of *Pseudomonas syringae*, a highly clonal, endemic plant pathogen, *Appl. Environ. Microbiol.*, **70**, 1999–2012.
- [Wolff, 1989] Wolff, R.W. (1989) *Stochastic modeling and the theory of queues*, Prentice-Hall International. p-311.



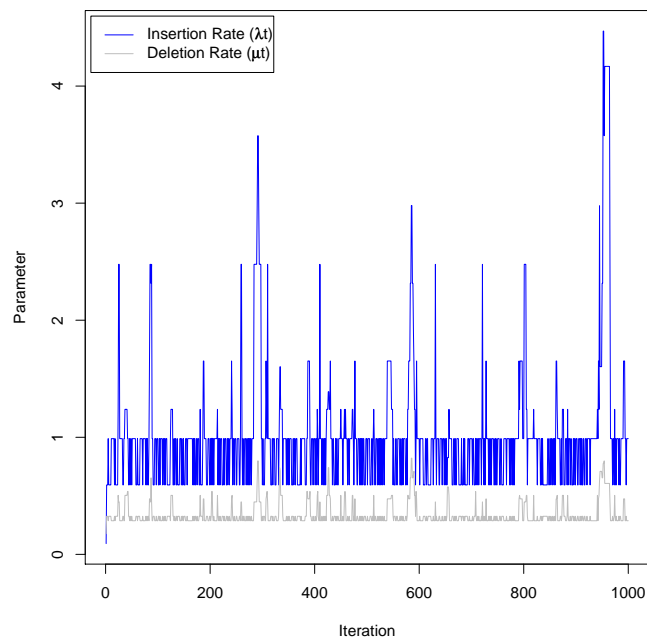
Supplementary Figure 1: Toy networks used for MCMC profiling. The network H'_i differs from the starting network H_1 by i hyperedges.



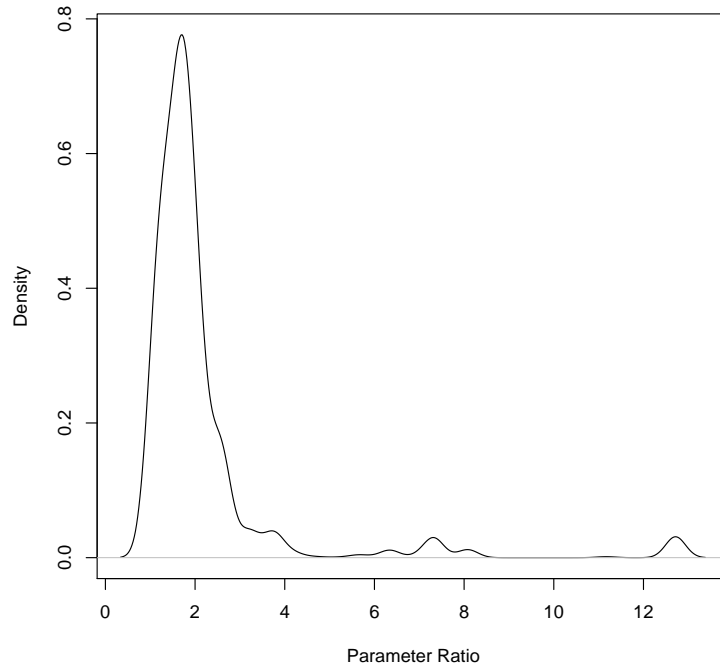
Supplementary Figure 2: Likelihood surfaces calculated by matrix exponentiation (True Likelihood) and by using the MCMC (Estimated Likelihood) for different insertion and deletion rates for the toy networks shown in Figure 1 in the main paper. The true and estimated maximum likelihood values are marked with asterisks. The maximum likelihood value was estimated using the Gibbs sampler described in the main paper.



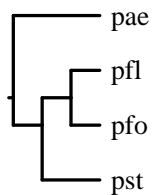
Supplementary Figure 3: Distribution of path lengths for different parameter combinations (λt : insertion rate, μt : deletion rate) averaged over three MCMC runs for the toy networks shown in Figure 1 in the main paper.



Supplementary Figure 4: An example MCMC trace showing the parameter values for the first 1,000 iterations of the Gibbs sampler for the toy networks shown in Figure 1 in the main paper.



Supplementary Figure 5: The density plot of the ratio (λ/μ) of estimated parameters for 2500 samples using the Gibbs sampler for the toy networks shown in Figure 1 in the main paper. The posterior mean of the ratio of the estimated parameters was calculated as 2.17 compared to the true ratio of 2.0.



Supplementary Figure 6: A phylogeny obtained using a multilocus sequence typing approach [Sarkar and Guttman, 2004] relating the four species (pae: *P. aeruginosa* PAO1, pfl: *P. fluorescens* Pf-5, pfo: *P. fluorescens* PFO-1, pst: *P. syringae* pv. tomato DC3000).

Supplementary Table 1: Basic information for the metabolic networks used in the analysis. Non-current metabolites (Mtbs) are metabolites excluding H₂O, CO₂, ATP and other ubiquitous metabolites. The codes MAP_{xxxxx} correspond to the respective KEGG pathway codes. [Kanehisa *et al.*, 2006].

Pathway Map	Organism	Reactions	Reversible	Hyperedges	Metabolites	Non-Current
Pentose phosphate pathway (MAP00030)	Reference	43	24	67	56	39
	<i>P. aeruginosa</i> PAO1	22	15	37	39	29
	<i>P. fluorescens</i> Pf-5	24	17	41	42	29
	<i>P. fluorescens</i> Pf0-1	26	17	43	43	30
	<i>P. syringae</i> pv. tomato DC3000	26	18	44	41	30
Lysine degradation (MAP00310)	Reference	56	20	76	81	60
	<i>P. aeruginosa</i> PAO1	7	4	11	19	11
	<i>P. fluorescens</i> Pf-5	9	5	14	25	16
	<i>P. fluorescens</i> Pf0-1	8	4	12	23	14
	<i>P. syringae</i> pv. tomato DC3000	8	4	12	24	13
Phenylalanine metabolism (MAP00360)	Reference	53	15	68	65	50
	<i>P. aeruginosa</i> PAO1	7	1	8	24	15
	<i>P. fluorescens</i> Pf-5	9	3	12	27	14
	<i>P. fluorescens</i> Pf0-1	8	1	9	26	16
	<i>P. syringae</i> pv. tomato DC3000	9	1	10	32	19

Supplementary Table 2: Hyperedge in and out degrees for the metabolic networks used in the analysis. Filtered networks are networks that do not include current metabolites such as H₂O, CO₂, ATP and other ubiquitous metabolites. The codes MAPxxxxx correspond to the respective KEGG pathway codes. [Kanehisa *et al.*, 2006].

Pathway Map	Organism	Full Network				Filtered Network			
		In-Degree		Out-Degree		In-Degree		Out-Degree	
		Avg	Max	Avg	Max	Avg	Max	Avg	Max
Pentose phosphate pathway (MAP00030)	Reference	3.284	14	3.284	10	2.239	10	2.239	6
	<i>P. aeruginosa</i> PAO1	2.757	8	2.757	8	1.919	7	1.919	6
	<i>P. fluorescens</i> Pf-5	2.512	8	2.512	7	1.854	7	1.854	6
	<i>P. fluorescens</i> Pf0-1	2.674	8	2.674	8	1.930	7	1.930	6
	<i>P. syringae</i> pv. tomato DC3000	2.864	9	2.864	8	2.136	9	2.136	6
Lysine degradation (MAP00310)	Reference	7.868	28	7.868	33	2.566	7	2.566	20
	<i>P. aeruginosa</i> PAO1	0.182	2	0.182	1	0.000	0	0.000	0
	<i>P. fluorescens</i> Pf-5	0.643	3	0.643	3	0.214	3	0.214	1
	<i>P. fluorescens</i> Pf0-1	0.250	2	0.250	1	0.083	1	0.083	1
	<i>P. syringae</i> pv. tomato DC3000	0.500	3	0.500	1	0.000	0	0.000	0
Phenylalanine metabolism (MAP00360)	Reference	7.338	20	7.338	27	2.029	12	2.029	6
	<i>P. aeruginosa</i> PAO1	0.375	3	0.375	1	0.375	3	0.375	1
	<i>P. fluorescens</i> Pf-5	1.083	5	1.083	2	0.917	3	0.917	2
	<i>P. fluorescens</i> Pf0-1	0.556	3	0.556	1	0.556	3	0.556	1
	<i>P. syringae</i> pv. tomato DC3000	1.200	3	1.200	4	0.400	3	0.400	1

Supplementary Table 3: MCMC results for paths sampling of networks which differed from the starting network H_1 shown in Figure 1 of the paper by 1 to 5 hyperedges. These tests were run to see how well the MCMC estimated likelihood values.

Differences (Network)	Iterations	True Likelihood	Estimated Likelihood		Error	CPU Time (seconds)		Acceptance Percentage	
			Average	Std Dev		Average	Std Dev	Average	Std Dev
1 (H'_1)	5000	0.022775	0.021641	2.08375×10^{-05}	1.13447×10^{-03}	13.85	1.39	41.73	0.98
	10000		0.021878	2.16488×10^{-05}	8.97562×10^{-04}	18.97	1.43	41.24	0.31
	20000		0.022105	2.50009×10^{-05}	6.70147×10^{-04}	30.44	0.94	40.50	0.25
	40000		0.022287	2.03600×10^{-05}	4.87823×10^{-04}	51.74	2.15	40.73	0.55
	80000		0.022423	6.43702×10^{-06}	3.51609×10^{-04}	88.31	3.50	40.69	0.24
2 (H'_2)	5000	0.016050	0.015322	1.33689×10^{-05}	7.28198×10^{-04}	13.48	1.07	63.31	0.70
	10000		0.015455	1.76166×10^{-05}	5.94877×10^{-04}	23.96	1.52	63.92	0.65
	20000		0.015550	2.72589×10^{-05}	5.00044×10^{-04}	39.44	2.06	63.24	0.60
	40000		0.015633	8.66006×10^{-06}	4.17699×10^{-04}	66.83	1.92	63.58	0.30
	80000		0.015717	1.11583×10^{-05}	3.33347×10^{-04}	113.53	1.01	63.46	0.14
3 (H'_3)	5000	0.002199	0.001994	1.14147×10^{-05}	2.05611×10^{-04}	24.82	2.27	59.79	1.21
	10000		0.002051	3.61673×10^{-06}	1.48477×10^{-04}	38.82	1.56	60.35	0.45
	20000		0.002079	2.41045×10^{-06}	1.20288×10^{-04}	68.71	3.72	60.46	0.03
	40000		0.002098	1.12774×10^{-06}	1.01868×10^{-04}	118.20	2.70	60.10	0.18
	80000		0.002109	6.25207×10^{-07}	9.00801×10^{-05}	209.80	2.18	60.24	0.08
4 (H'_4)	5000	0.000287	0.000222	5.84773×10^{-07}	6.52775×10^{-05}	42.94	0.92	57.14	1.00
	10000		0.000239	1.25982×10^{-06}	4.80291×10^{-05}	71.76	1.57	57.31	0.14
	20000		0.000252	5.61035×10^{-07}	3.49229×10^{-05}	113.15	2.53	57.46	0.41
	40000		0.000262	2.79423×10^{-07}	2.57824×10^{-05}	188.58	6.13	57.28	0.26
	80000		0.000267	8.93104×10^{-08}	2.03820×10^{-05}	343.90	10.92	57.17	0.16
5 (H'_5)	5000	0.000106	0.000075	7.12662×10^{-08}	3.07056×10^{-05}	73.51	2.88	55.33	1.14
	10000		0.000077	3.77709×10^{-08}	2.83951×10^{-05}	128.48	1.48	55.00	0.67
	20000		0.000081	6.44931×10^{-08}	2.48370×10^{-05}	222.92	1.77	55.37	0.30
	40000		0.000086	1.86982×10^{-07}	2.01615×10^{-05}	362.29	2.25	55.24	0.28
	80000		0.000091	1.42376×10^{-07}	1.52476×10^{-05}	602.24	8.80	55.35	0.25

Supplementary Table 4: Testing of $P(H_1)P(H_2|H_1) = P(H_2)P(H_1|H_2)$ for the toy networks shown in Figure 1 of the paper.

Case	Parameters	$P(H_1)$	$P(H_2 H_1)^*$	$P(H_1)P(H_2 H_1)$	$P(H_2)$	$P(H_1 H_2)^*$	$P(H_2)P(H_1 H_2)$
$\mu = 2\lambda$	$\lambda t = 0.1, \mu t = 0.2$	6.77404×10^{-05}	$0.003831 (\pm 7.86115 \times 10^{-07})$	2.59484×10^{-07}	6.77404×10^{-05}	$0.003830 (\pm 1.34369 \times 10^{-07})$	2.59434×10^{-07}
	$\lambda t = 0.3, \mu t = 0.6$	6.77404×10^{-05}	$0.011713 (\pm 1.03473 \times 10^{-06})$	7.93441×10^{-07}	6.77404×10^{-05}	$0.011715 (\pm 1.21154 \times 10^{-07})$	7.93546×10^{-07}
	$\lambda t = 0.5, \mu t = 1.0$	6.77404×10^{-05}	$0.012016 (\pm 2.14371 \times 10^{-06})$	8.13978×10^{-07}	6.77404×10^{-05}	$0.012017 (\pm 3.79601 \times 10^{-06})$	8.14053×10^{-07}
	$\lambda t = 0.7, \mu t = 1.4$	6.77404×10^{-05}	$0.009391 (\pm 7.93988 \times 10^{-06})$	6.36123×10^{-07}	6.77404×10^{-05}	$0.009389 (\pm 2.48905 \times 10^{-06})$	6.35983×10^{-07}
	$\lambda t = 0.9, \mu t = 1.8$	6.77404×10^{-05}	$0.006588 (\pm 7.41074 \times 10^{-06})$	4.46249×10^{-07}	6.77404×10^{-05}	$0.006582 (\pm 2.11670 \times 10^{-06})$	4.45879×10^{-07}
$\mu = \lambda$	$\lambda t = 0.1, \mu t = 0.1$	9.76562×10^{-04}	$0.002402 (\pm 2.15206 \times 10^{-07})$	2.34554×10^{-06}	9.76562×10^{-04}	$0.002401 (\pm 1.21654 \times 10^{-07})$	2.34513×10^{-06}
	$\lambda t = 0.3, \mu t = 0.3$	9.76563×10^{-04}	$0.011128 (\pm 4.66911 \times 10^{-07})$	1.08675×10^{-05}	9.76563×10^{-04}	$0.011129 (\pm 5.78902 \times 10^{-07})$	1.08685×10^{-05}
	$\lambda t = 0.5, \mu t = 0.5$	9.76562×10^{-04}	$0.016570 (\pm 1.53568 \times 10^{-06})$	1.61815×10^{-05}	9.76562×10^{-04}	$0.016569 (\pm 7.46601 \times 10^{-07})$	1.61811×10^{-05}
	$\lambda t = 0.7, \mu t = 0.7$	9.76562×10^{-04}	$0.018086 (\pm 1.28336 \times 10^{-06})$	1.76619×10^{-05}	9.76562×10^{-04}	$0.018083 (\pm 5.72389 \times 10^{-06})$	1.76591×10^{-05}
	$\lambda t = 0.9, \mu t = 0.9$	9.76562×10^{-04}	$0.017266 (\pm 7.12104 \times 10^{-06})$	1.68610×10^{-05}	9.76562×10^{-04}	$0.017246 (\pm 6.14501 \times 10^{-06})$	1.68415×10^{-05}
$\mu = \lambda/2$	$\lambda t = 0.1, \mu t = 0.05$	4.33538×10^{-03}	$0.001345 (\pm 1.48361 \times 10^{-07})$	5.83224×10^{-06}	4.33538×10^{-03}	$0.001345 (\pm 4.93860 \times 10^{-08})$	5.83218×10^{-06}
	$\lambda t = 0.3, \mu t = 0.15$	4.33538×10^{-03}	$0.007699 (\pm 2.87349 \times 10^{-07})$	3.33761×10^{-05}	4.33538×10^{-03}	$0.007696 (\pm 2.74647 \times 10^{-06})$	3.33658×10^{-05}
	$\lambda t = 0.5, \mu t = 0.25$	4.33538×10^{-03}	$0.013885 (\pm 1.23405 \times 10^{-06})$	6.01955×10^{-05}	4.33538×10^{-03}	$0.013883 (\pm 4.40639 \times 10^{-06})$	6.01881×10^{-05}
	$\lambda t = 0.7, \mu t = 0.35$	4.33538×10^{-03}	$0.018033 (\pm 2.29231 \times 10^{-06})$	7.81780×10^{-05}	4.33538×10^{-03}	$0.018034 (\pm 3.20188 \times 10^{-06})$	7.81838×10^{-05}
	$\lambda t = 0.9, \mu t = 0.45$	4.33538×10^{-03}	$0.020149 (\pm 4.19149 \times 10^{-06})$	8.73553×10^{-05}	4.33538×10^{-03}	$0.020149 (\pm 4.69874 \times 10^{-06})$	8.73547×10^{-05}

* Averaged over 3 runs

Supplementary Table 5: Equilibrium probability approximation for the toy network H_1 shown in Figure 1 of the paper. The results are shown for different subnetwork sizes.

Case	Parameters	Sub Network Size	$P(H_1)$
$\mu = 2\lambda$	$\lambda t = 0.1, \mu t = 0.2$	2	$6.77403512337 \times 10^{-05}$
		4	$6.77403512337 \times 10^{-05}$
		6	$6.77403512337 \times 10^{-05}$
		8	$6.77403512337 \times 10^{-05}$
		10*	$6.77403512343 \times 10^{-05}$
$\mu = \lambda$	$\lambda t = 0.1, \mu t = 0.1$	2	$9.76562500000 \times 10^{-04}$
		4	$9.76562500000 \times 10^{-04}$
		6	$9.76562500000 \times 10^{-04}$
		8	$9.76562500000 \times 10^{-04}$
		10*	$9.76562499994 \times 10^{-04}$
$\mu = \lambda/2$	$\lambda t = 0.1, \mu t = 0.05$	2	$4.33538247896 \times 10^{-03}$
		4	$4.33538247896 \times 10^{-03}$
		6	$4.33538247896 \times 10^{-03}$
		8	$4.33538247895 \times 10^{-03}$
		10*	$4.33538247896 \times 10^{-03}$

* Full Network

Supplementary Table 6: Effective sample size (ESS) for the estimated parameters (λt : insertion rate and μt : deletion rate) for the evolution of metabolic networks with and without core and prohibited edges. Three runs of 100,000 iterations each with additional 10,000 iterations as burn-in period were run in each case.

Pathway Map	Start Organism	End Organism	Differences	CnP	ESS_{λt}	ESS_{μt}
Pentose phosphate pathway (MAP00030)	<i>P. aeruginosa</i> PAO1	<i>P. syringae</i> pv. tomato DC3000	9 (I:8, D:1)	-	16514	16514
	<i>P. fluorescens</i> Pf-5	<i>P. fluorescens</i> Pf0-1	2 (I:2, D:0)	+	9966	8314
Lysine degradation (MAP00310)	<i>P. aeruginosa</i> PAO1	<i>P. syringae</i> pv. tomato DC3000	1 (I:1, D:0)	-	24491	14888
	<i>P. fluorescens</i> Pf-5	<i>P. fluorescens</i> Pf0-1	2 (I:0, D:2)	+	609	572
Phenylalanine metabolism (MAP00360)	<i>P. aeruginosa</i> PAO1	<i>P. syringae</i> pv. tomato DC3000	6 (I:4, D:2)	-	11487	11487
	<i>P. fluorescens</i> Pf-5	<i>P. fluorescens</i> Pf0-1	7 (I:2, D:5)	+	4164	5762
				-	14888	12435
				+	5167	5762

* I: number of insertions, and D: number of deletions going from the start organism to the end organism ** CnP: Core and prohibited hyperedges

Supplementary Table 7: Maximum likelihood values and the corresponding parameter values for the evolution of metabolic networks with and without core and prohibited edges using the Gibbs sampler. The results are averaged across three runs of 100,000 iterations each with a further 10,000 iterations as burn-in period.

Pathway Map	Start Organism	End Organism	Differences*	CnP**	λt	μt	Likelihood (St. Deviation)
Pentose phosphate pathway (MAP00030)	<i>P. aeruginosa</i> PAO1	<i>P. syringae</i> pv. tomato DC3000	9 (I:8, D:1)	-	0.32534	0.05951	7.64751×10^{-17} (7.79517×10^{-19})
	<i>P. fluorescens</i> Pf-5	<i>P. fluorescens</i> Pf0-1	2 (I:2, D:0)	+	2.11468	0.24400	7.39011×10^{-10} (1.34090×10^{-11})
Lysine degradation (MAP00310)	<i>P. aeruginosa</i> PAO1	<i>P. syringae</i> pv. tomato DC3000	1 (I:1, D:0)	-	0.08871	0.20334	2.26363×10^{-04} (1.20197×10^{-07})
			2 (I:0, D:2)	+	0.35395	1.22001	2.46143×10^{-03} (5.83379×10^{-08})
	<i>P. fluorescens</i> Pf-5	<i>P. fluorescens</i> Pf0-1	2 (I:0, D:2)	-	0.02335	0.44436	5.06807×10^{-04} (1.66351×10^{-07})
			2 (I:2, D:0)	+	0.22811	2.22180	1.48190×10^{-02} (1.04041×10^{-05})
Phenylalanine metabolism (MAP00360)	<i>P. aeruginosa</i> PAO1	<i>P. syringae</i> pv. tomato DC3000	6 (I:4, D:2)	-	0.07471	0.49355	6.39355×10^{-17} (5.15772×10^{-20})
			7 (I:2, D:5)	+	0.48594	2.01981	3.06207×10^{-12} (5.14035×10^{-14})
	<i>P. fluorescens</i> Pf-5	<i>P. fluorescens</i> Pf0-1	7 (I:2, D:5)	-	0.04325	0.78150	2.52682×10^{-10} (5.20471×10^{-13})
				+	0.35525	2.30417	1.86407×10^{-06} (7.14206×10^{-08})

* I: number of insertions, and D: number of deletions going from the start organism to the end organism
** CnP: Core and prohibited hyperedges