I. PJKJ10 MODEL

The model is

$$\begin{aligned}\frac{dD}{dt} &= \psi_D D \\ \frac{dR}{dt} &= \psi_R R - (\gamma_D D + \gamma_T T) R \\ \frac{dT}{dt} &= \psi_T T + (\gamma_D D + \gamma_T T) R \\ \frac{dC}{dt} &= -(\epsilon_D \psi_D D + \epsilon_R \psi_R R + \epsilon_T \psi_T T)\end{aligned}$$

The details of the coefficients (ψ, γ, ϵ) are given in the reference paper.[1] In general these are nonlinear functions of time and the substrate, C. From these coefficient functions there are **6 parameters** which must be estimated.

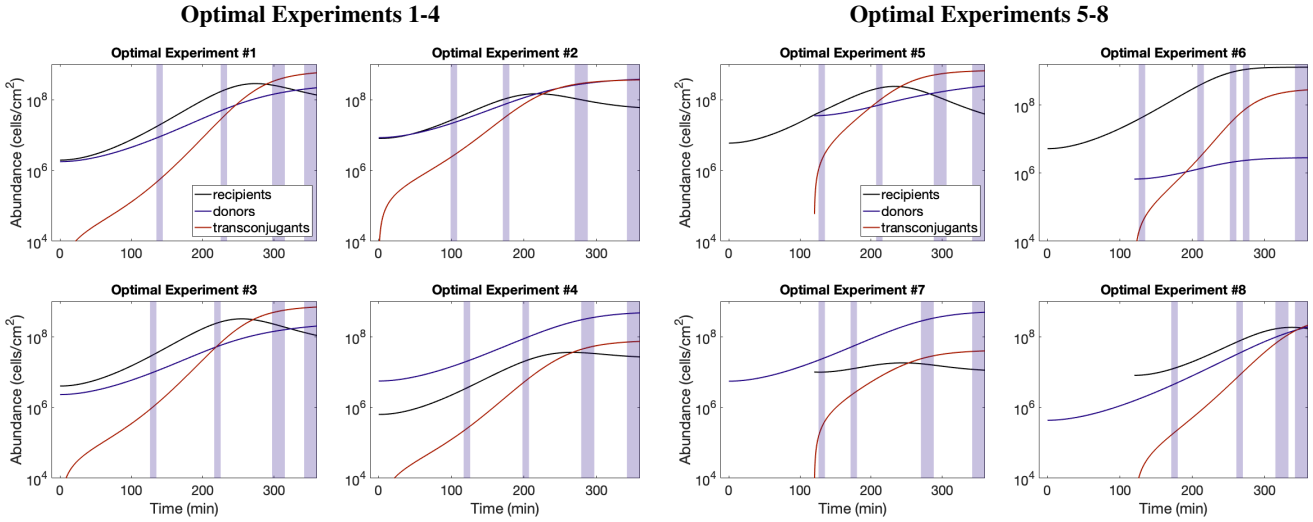


FIG. 1. Reproduction of plots presented in the paper[1] using Euler's method. Shows how the system evolves with various initial conditions. Shaded regions are optimal sampling regions, 6 samples over 9 minute intervals. Code used to generate each plot is saved in the code folder. Sensitivities are tabulated in the sensitivites.txt file.

II. OPTIMAL EXPERIMENTAL DESIGN

Given a limited amount of samples, c_{max}, over some time interval, T, we would like to select the sampling schedule, $\boldsymbol{w}$, which maximizes the D-optimality score. **Each sample is a measurement of D, R, and T simultaneously**. C is never measured.

We also wish to find the optimal initial conditions for the state variables, D and R. The initial conditions of T and C are set to be 0 and 1 respectively so that the abundance scale is $C(0)$.

The FIM can also be written in discrete form as[1]

$$\mathbf{I} = \boldsymbol{S}^T \boldsymbol{w} \boldsymbol{S}$$

Where $\boldsymbol{S}$ is the matrix of sensitivities at all times.

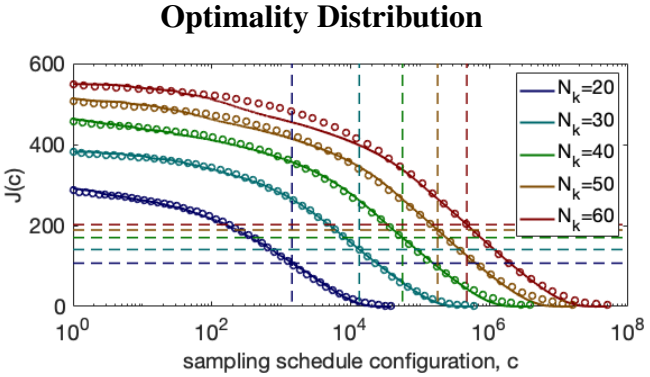
$$\boldsymbol{S} = \left[\frac{d\bar{\boldsymbol{x}}}{d\boldsymbol{\theta}}^T (t_0) \;\; \frac{d\bar{\boldsymbol{x}}}{d\boldsymbol{\theta}}^T (t_1) \;\; \ldots \;\; \frac{d\bar{\boldsymbol{x}}}{d\boldsymbol{\theta}}^T (t_N) \right]^T$$

Therefore optimizing the sampling schedule (with fixed initial conditions) only requires calculating $\boldsymbol{S}$ once. $\boldsymbol{S}$ can then be stored in memory while $\boldsymbol{w}$ is varied. $\boldsymbol{w}$ is a very sparse matrix so objective function computations will be fast.

A more natural representation of $\boldsymbol{w}$ is the decomposition $\boldsymbol{w} = \boldsymbol{m}^T \boldsymbol{m}$, where $\boldsymbol{m}$ is defined as the measurement matrix which has the effect of partially collapsing the sensitivity matrix, $\boldsymbol{S}$. That is, if there are N_s state variables, N_o observable state variables, N_t

time points, N_p parameters, and N_m measurements, then $\boldsymbol{S}$ is a $N_s N_t \times N_p$ matrix and $\boldsymbol{m}$ is a $N_o N_m \times N_s N_t$ matrix, so that $\boldsymbol{mS}$ is a $N_o N_m \times N_p$ matrix which is a significant reduction in size.

If the dimensions of $\boldsymbol{w}$ are small enough then the optimality score, J, can be computed exhaustively for all possible sampling schedules. For example, if the time interval is discretized into 25 sampling intervals and 6 samples are taken over the full time interval then there are 25 choose $6 \approx 10^5$ possible sampling schedules. In practice the sensitivities are calculated on a much finer grid and the mean value is computed over sampling intervals.



N_k	N_c	J^*	c_r/N_c	β	gof	cpu time (s)
20	4×10^4	291.2	0.037	0.50	tbd	1.389
30	6×10^5	382.1	0.023	0.45	tbd	16.79
40	4×10^6	463.1	0.015	0.38	tbd	107.7
50	2×10^7	513.2	0.011	0.35	tbd	455.7
60	5×10^7	549.3	0.009	0.35	tbd	1483

FIG. 2. Direct calculation of the D-optimality score for all possible sampling schedules for various sampling interval sizes where 6 samples are taken over 360 minutes. There are $N_c = (N_k$ choose 6) possible configurations for each sampling interval size. The sampling schedules are sorted after calculation. Dashed lines intersect at the points $(c_r, J(c_r))$. o-markers are a stretched exponential fit (by eye). Initial conditions are that of experiment #1 in FIG. 1. As N_k increases, a noticeable gap forms at $c \approx 10^2$ between the highest optimality states and the next highest.

The distribution of J is a stretched exponential

$$J(c) = J^* e^{-(c/c_r)^\beta}.$$

If $N_k = 60$, $c = c_r$, then $J(c_r) = J^*/e \approx 202$. Only $\sim 0.9\%$ of all possible sampling schedules have an optimality score greater than $J = 202$ (left of the dashed red line). The vast majority of the other $\sim 99.1\%$ of possible experiments have an optimality score close to zero.

$\beta \in [0, 1]$ is a measure of how fast the optimality score drops over the set of configurations. Smaller values of β mean there are more experiments which are reasonably informative.

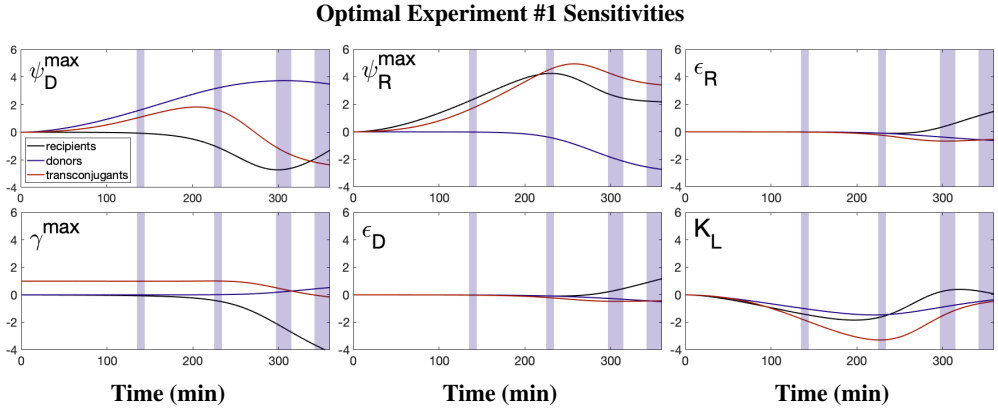


FIG. 3. Sensitivities calculated by finite differences for experiment #1. The shaded regions are the optimal sampling intervals.

The optimal sampling schedule balances the parameter-uncertainty information received by all state measurements so that samples are taken when state variables sensitive to changes in parameters (not necessarily independenty).

III. OPTIMAL INITIAL CONDITIONS

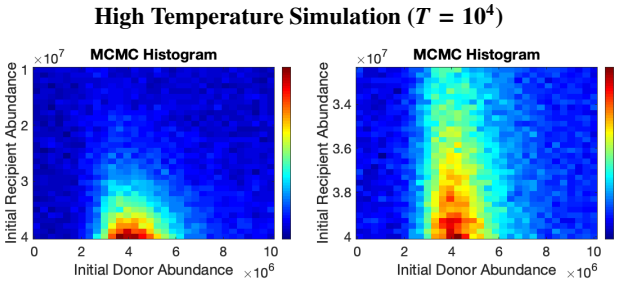


FIG. 4. MCMC histogram of the model with variable initial conditions and variable sampling schedule. The simulated temperature is $T = 1000$, and 10^5 Monte Carlo samples are taken. At least one Monte Carlo sweep is performed for each of the two different types of updates. If there are any local minima, they would be visible in the histogram given the Markov chain is sufficiently ergodic.

IV. SIMULATED ANNEALING

A. Local Update

A Simulated annealing algorithm uses a Markov-chain Monte Carlo method to simulate thermodynamics (i.e. create a Boltzmann distribution). The transition probability for a system to undergo an update from initial state, i, to updated state, j, in the Metropolis algorithm is

$$W_{ij} = \begin{Bmatrix} 1 & \text{if } J_j - J_i < 0 \\ \exp(-(J_j - J_i)/T) & \text{if } J_j - J_i > 0 \end{Bmatrix} \begin{matrix} \textit{cooling step} \\ \textit{fluctuations} \end{matrix}$$

The update is defined as swapping a current sampling interval with a non-sampling interval. This is a *local* update since it only involves updating **one** sampling interval. More computationally intensive *global* updates should be defined if local updates lead to non-ergodic behaviour in the Markov chain.

Let $\boldsymbol{o}$ represent the old measurement matrix before updating and $\boldsymbol{n}$ represent the new updated measurement matrix. If a measurement interval is updated then the change in the objective function after this update is

$$J_{\boldsymbol{n}} - J_{\boldsymbol{o}} = \det\left((\boldsymbol{nS})^T(\boldsymbol{nS})\right) - \det\left((\boldsymbol{oS})^T(\boldsymbol{oS})\right)$$

Since the measurements are taken simultaneously for all observable state variables, the measurement matrix can be represented as,

$$\boldsymbol{m} = \left[\boldsymbol{m}^{(1)}\boldsymbol{m}^{(2)}\ldots\boldsymbol{m}^{(N_t)}\right]$$

where $\boldsymbol{m}^{(i)}$ (unnormalized) is a vector of size N_o which is all ones if a measurement is taken on the i-th interval or all zeros otherwise.

In this notation the Metropolis algorithm is

Choose (random) $r_1 \in [1, N_t]$ such that $\boldsymbol{m}^{(r_1)} = \mathbf{1}$.
Choose (random) $r_0 \in [1, N_t]$ such that $\boldsymbol{m}^{(r_0)} = \mathbf{0}$.
Define $\boldsymbol{n} = \boldsymbol{m}$ set $\boldsymbol{n}^{(r_1)} = \mathbf{0}$ and $\boldsymbol{n}^{(r_0)} = \mathbf{1}$.
Compute $\Delta J = \det\left((\boldsymbol{nS})^T(\boldsymbol{nS})\right) - J_0$.
Update with $W_{ij} = \begin{Bmatrix} 1 & \text{if } \Delta J < 0 \\ \exp(-\Delta J/T) & \text{if } \Delta J > 0 \end{Bmatrix}$.

Probability balance is satisfied since updates which switch two measured intervals don't change the optimality. These types of updates can be ignored.

The computational cost of the update can be improved for systems with a high-dimensional parameter space by using the matrix determinant lemma.

B. Simulated Thermodynamics

Let c be some sampling schedule configuration and C a uniform random schedule. Then the Boltzmann distribution is defined as

$$\pi(c, T) = \Pr(C = c \mid T) = \frac{\exp(-J(c)/T)}{\sum_{c_i} \exp(-J(c_i)/T)}.$$

So that the Markov chain spends most of its time in the high optimality states.

A thermal average is a configurational average assuming a Boltzmann distribution

$$\langle J \rangle = \sum_c J(c)\pi(c, T).$$

This can be computed exactly if the optimality distribution is known. The exact computation can then be plotted with MCMC data to ensure Boltzmann sampling (FIG. 5.).

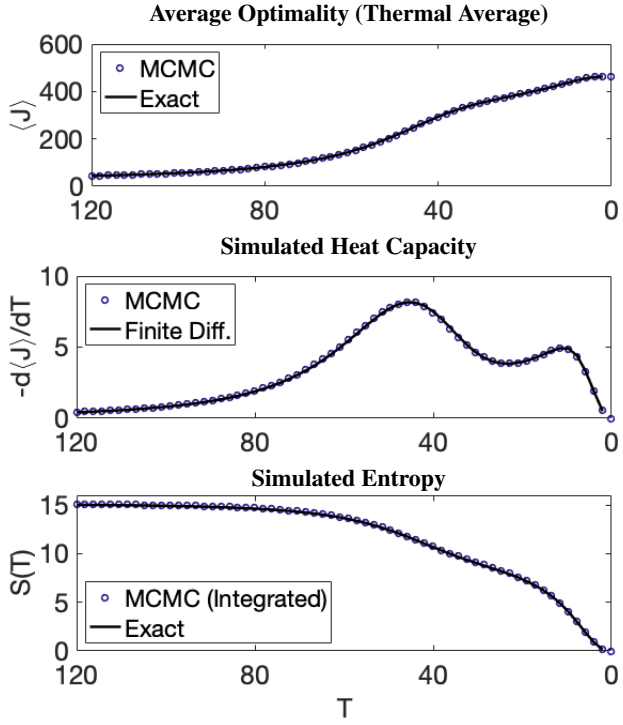


FIG. 5. Simulated thermodynamics for a high temperature state being cooled to $T = 0$ (Annealing). 10^6 MCMC samples are taken for each temperature point. The MCMC heat capacity is interpolated by cubic splines then integrated with Euler's method in the entropy calculation.

The simulated heat capacity is the change in thermal-averaged optimality with temperature. It can be computed in the MCMC simulation by using the fluctuation formula

$$-\frac{d\langle J \rangle}{dT} = -\frac{d}{dT} \sum_c J(c)\pi(c, T) = \frac{\langle J^2 \rangle - \langle J \rangle^2}{T^2}.$$

The simulated heat capacity is also a measure of the change in simulated entropy

$$S(T) = -\sum_c \pi(c, T) \log(\pi(c, T)) = -\int_0^T \frac{1}{T'} \left.\frac{d\langle J \rangle}{dT}\right|_{T=T'} dT'.$$

Peaks in the heat capacity indicate large changes in entropy (Schottky anomaly). That is, a peak appears when there is a large gap in optimality between configuration subsets. These gaps are evident in the optimality distribution plot.

Any of the above thermodynamic quantities can be checked with exact calculations to ensure Boltzmann sampling in the MCMC simulation.

C. Autocorrelations

The autocorrelation function is defined as

$$C(t) = \langle J(0)J(t)\rangle$$

where t is the amount of Monte Carlo sweeps that have past since taking a sample of J. The autocorrelation function measures how correlated the Markov chain is with past configurations. If the autocorrelation function doesn't go to zero as t increases then the algorithms isn't ergodic.

---

[1] A. Malwade, A. Nguyen, P. Sadat-Mousavi, and B. P. Ingalls, Frontiers in Microbiology **8**, 461 (2017).