Global Optimization of Equilibrium-Dynamic Models to Fit Time Process Data

With Applications to Ribosomal RNA folding reactions

Stanford University iCME

October 13, 2010

1 Model

Suppose we have some time process data of an equilibrating system that is a linear combination of the ensemble-averaged states of the system. We wish to compute the optimal discrete-state continuous-time equilibrium-dynamic model or "master equation" to fit these observations. Let there be m discrete states of the system and the consider a time-dependent ensemble-average probability vector $p(t) \in [0,1]^m$, representing the probability of existing in states $i = 1, \ldots, m$ at time t, subject to the system of first-order ODEs

$$\frac{d}{dt}p = Dp$$

for some negative semidefinite $D \in \mathbb{R}^{m\times m}$; D usually has one zero eigenvalue whose corresponding right-eigenvector represents the equilibrium probability distribution at $t = \infty$.

2 Objective Function (LS)

Let $K \in \mathbb{R}^{m\times m}_{\geqslant 0}$ represent the off-diagonal elements of D; without loss of generality we can assume $k_{ii} = 0 \ \forall i = 1, \ldots, m$. Non-negativity of each $p_i(t)$ under time evolution requires that $k_{ij} \geqslant 0, i \neq j$. Compute $D \in \mathbb{R}^{m\times m}$ such that

$$d_{ij} = \left\{ \begin{array}{cc} k_{ij}, & i \neq j \\ -\sum_{i\neq j} k_{ij}, & i = j \end{array} \right. .$$

Note that

$$0 = e^T D,$$

i.e., e^T is a left-eigenvector of D with eigenvalue 0.

$$\begin{aligned} \sum_i p_i = 1 &\Longleftrightarrow \\ e^T p = 1 &\Longrightarrow \\ e^T(p + Dp) = 1, \end{aligned}$$

telling us that no net probability is lost or gained from any state under this process as required to maintain the probability distribution requirements.

Solve for $\{\lambda_n : \lambda_n \in \mathbb{C}\}_{n=1}^m, \{\varphi_n : \varphi_n \in \mathbb{C}^m\}_{n=1}^m$, the right-eigensystem. Let $X(t)$ represent the vector of solutions to the system of equations,

$$X(t) = \sum_{n=1}^{m} c_n e^{-\lambda_n t} \varphi_n.$$

For our application, at $t = 0$ the ensemble is in the state $p_i = \delta_{i1}$ and at $t = \infty$ the ensemble is in the state $p_i = \delta_{im}$, although this is not a general constraint. From

$$X(0) = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = e_1,$$

solve for $\{c_n\}_{n=1}^m$ in

$$X(t) = \sum_{n=1}^{m} c_n e^{-\lambda_n t} \varphi_n.$$

I.e., solve the system of linear equations

$$(\varphi_1 \cdots \varphi_m) \begin{pmatrix} c_1 \\ \vdots \\ c_m \end{pmatrix} = e_1.$$

Form the matrix A defining the LS problem such that

$$A \equiv \begin{pmatrix} X_1^T \\ \vdots \\ X_N^T \end{pmatrix}$$

where N is the number of time points and $X_k \equiv X(t_k)$, the value of the X vector at measurement $k = 1, \ldots, N$; t_k is the time that the measurement was taken.

Since we require our experimental data to be representable as linear combinations of $\{X_i\}_{i=1}^m$, we want to find the linear combination of the X curves to best fit our measurements. Assume we have M time-process curves to fit; for each curve C_i with $i = 1, \ldots, M$, solve the variable-constrained linear least squares problem

$$f_i \equiv \min_{0 \leqslant y \leqslant 1} ||Ay - C_i||$$

and let

$$f \equiv \sum_{i=1}^{M} f_i$$

where

$$A = \begin{pmatrix} X_1^T \\ \vdots \\ X_N^T \end{pmatrix}.$$

The following Matlab code solves this variable-constrained linear least-squares problem:

```
options = optimset('LargeScale', 'off', 'Display', 'off') ;
sumres = 0 ;
for k = 1 : p % loop over experimental curves
  y(:, k) = lsqlin(A, C(k, :)', -eye(m), zeros(m, 1), [], [], zeros(m, 1), ones(m, 1), [], options) ;
  sumres = sumres + norm(A * y(:, k) - C(k, :)') ; % objective function value is a sum of residuals
end
```

Repeat for each curve and sum the errors to get the total error of the fit. This represents the objective function calculation which can be optimized over all possible K.

3 Global Optimization (NLS)

Using the above formulation of the objective function $f(\kappa)$, we can solve the nonlinear least-squares problem over the space of models. We now want to minimize f over the space of matrices K subject to $k_{mj} = 0 \ \forall j = 1, \ldots, m$. Indexing the remaining variable upper and lower diagonal elements of K using single subscript yields a vector κ which can be used more easily in standard optimization algorithms.

$$\min_{\kappa \geqslant 0} f(\kappa)$$

We solved this NLS problem using a gradient descent method, e.g. "active set" or "interior-point". We then repeat this solution using a multidimensional grid of starting values to check for multiple local minima and hope that the best minimum overall represents the true global minimum. The following Matlab command illustrates how to solve the NLS problem for some particular value of κ.

```
[k1 f1 exitflag output] = fmincon(@(x) f(x, t, C), k0, -eye(length(k0)), zeros(length(k0), 1), [],
[], [], [], [], options) ;
```

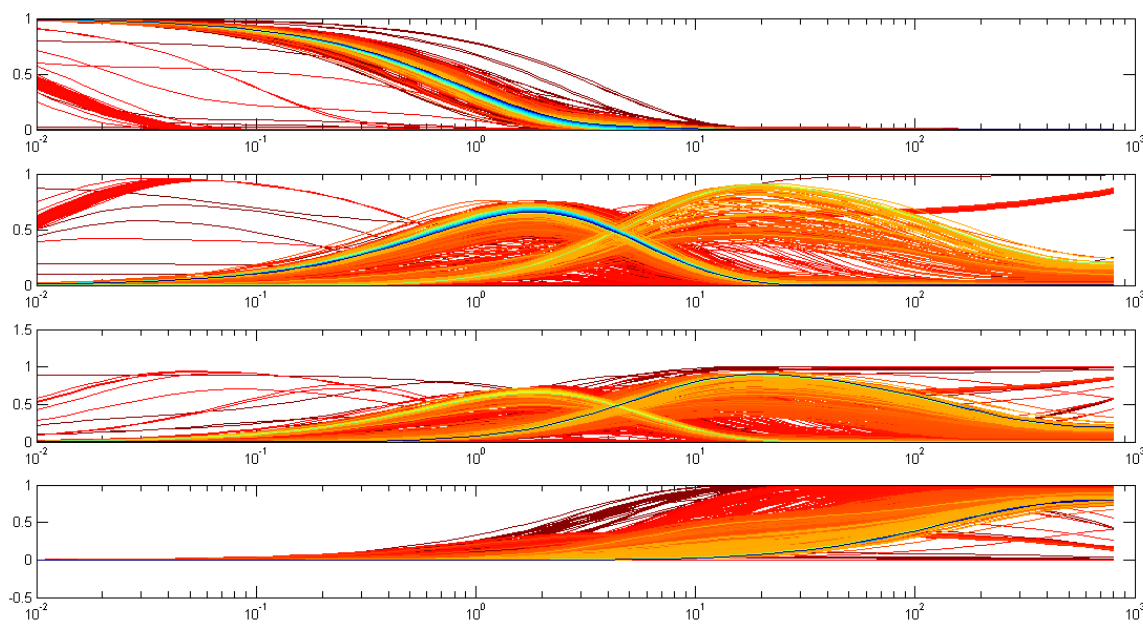

Figure 1: overlay of 15625 grid point solutions colored by objective function value index

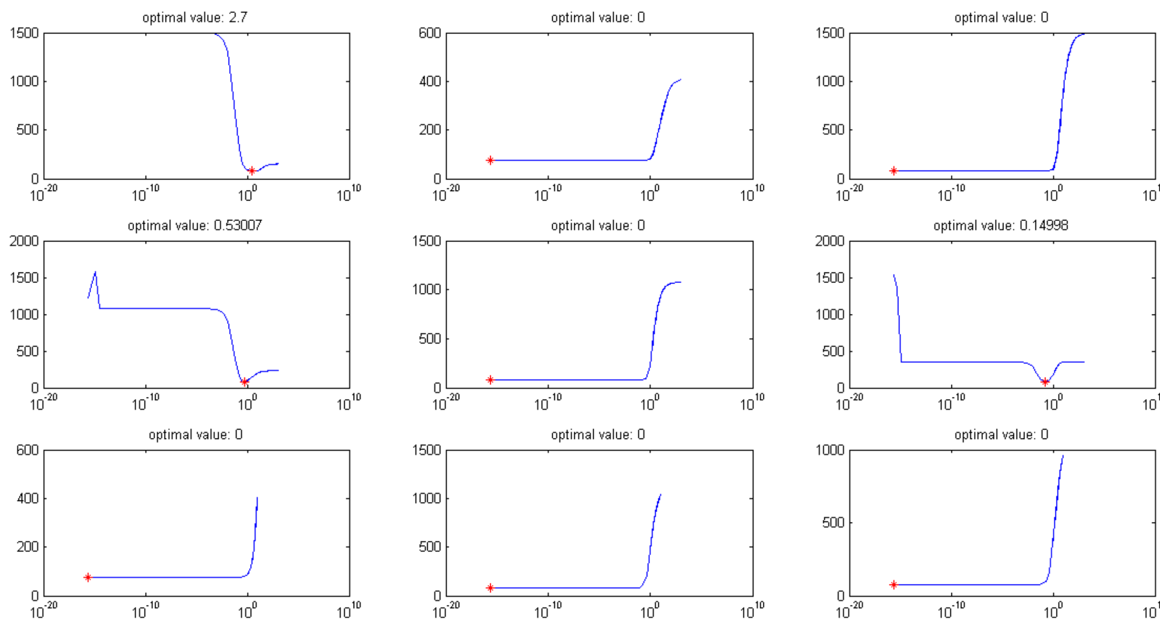

Figure 2: global optimum (243 grid points) 9-dimensional objective function profiles for ribosomal RNA folding dataset (M=522)