SPATIAL DISTRIBUTION OF PROTEIN CONCENTRATION CAN CONTROL CELLULAR PATHWAY FLUXES

BENG 221
Hooman Hefzi and Andrew Richards
October 21, 2011

INTRODUCTION

One of the goals of systems biology is to create models for intracellular pathway fluxes, which can then be used to design new strains or to predict cellular behavior for a strain. A pathway network can be mathematically reconstructed using singular perturbation theory. Data is gathered from *in vitro* experiments in which specific pathway components are sequentially altered in some way, such as a knockdown/knockout, or perhaps by an inhibitor molecule. Step by step, the observed effect of changing individual pathway components allows investigators to reverse engineer a metabolic map[1]. The steady state solutions and response characteristics of cells can then be found and modeled computationally[2]. These models often suggest strikingly coordinated behavior, such as robust bistability and stable limit cycles[3]. There are, however, limitations on the predictive power of such models due to certain assumptions, which makes it difficult to connect computational results to the real world biology.

One such assumption concerns the spatial distribution of both enzymes within a given pathway, as well as small molecules such as pathway intermediates and signaling molecules. The concentrations of these species are often treated as spatially invariant, thereby treating the cell as a well-mixed system with homogeneous distribution of both enzymes and substrates. In reality, cells have a highly organized (as well as compartmentalized, in the case of eukaryotes), and the laws of mass transfer play a significant role in overall pathway fluxes[4].

Here we show that the spatial distribution of protein concentration can dramatically alter the flux through otherwise equivalent pathways using the same starting substrate, released in a finite burst and diffusing through the cell. One assumption that is physiologically accurate is that mass transfer in the cytoplasm is by diffusion only. The diffusion length of a small molecule is $2\sqrt{Dt}$, where D is diffusivity (typically cm^2/s) and t is time. The distance over which diffusion is effective for mass transport is therefore limited by the stability and half-life of the diffusing species. Typical cell diameters (~0.1 μm for prokaryotes, 1-10 μm for eukaryotes) are on the same order of magnitude as diffusion lengths for most small molecules.

ANALYTICAL SOLUTION

We began by modeling the cell as a rectangular slab with width L. A small molecule u is released from a plane representing an organelle in the center of the cell at x = L/2 and diffuses outwards. At the edges of the cell, the small molecule rapidly leaks into extracellular environment, fixing the concentration at zero for x = 0 and x = L. We modeled the initial concentration U(x,0) as a fourth-power sine function to approximate a finite pulse of the molecule (Note: We use U to refer to both the small molecule itself as well as its concentration). We then considered two protein pathways which use our small molecule as a starting substrate, Pathways A and Pathway B. As we were concerned only with the consumption of small molecule U and which pathway consumes it, only the starting proteins in each pathway were relevant to our model; we refer to these as Protein A and Protein B, respectively. We first considered the case in which A and B are both distributed homogenously (constant concentration). Furthermore, we set the total amount of A and B equal, and assumed that their binding affinities and rate constants were the same. The consumption of U was therefore modeled as a driving source term,

$$Q = R(A + B)U,$$

in the differential equation. Finally, we assumed no convective terms, such that all mass transport within the cell was due to random-walk diffusion only, as described by Fick's second law of diffusion. The partial differential describing the dynamic behavior of our system was therefore

$$\frac{\partial U}{\partial t} = D\frac{\partial^2 U}{\partial x^2} - R(A+B)U, \qquad U = U(x,t), \qquad 0 \le x \le L$$

$$BC: \begin{cases} U(0,t) = 0 \\ U(L,t) = 0 \end{cases} \qquad IC: U(x,0) = sin^4(\pi x)$$

We used separation of variables to arrive at general form for our solution:

$$U(x,t) = F(x)G(t)$$

Plugging in for U:

$$\frac{\partial U}{\partial t} = D\frac{\partial^2 U}{\partial x^2} - R(A+B)U \rightarrow F\frac{dG}{dt} = DG\frac{d^2U}{d^2x} - R(A+B)FG$$

Rearrange:

$$\frac{1}{DG}\frac{dT}{dt} + \frac{R(A+B)}{D} = \frac{1}{F}\frac{d^2F}{dx^2} = -\lambda$$

Assuming an exponential function, we solved for G:

$$G = G_0 e^{[-\lambda - \frac{R(A+B)}{D}]Dt}$$

Solution for F where $\lambda = 0$:

$$\frac{d^2F}{dx^2} = 0 \rightarrow \frac{dF}{dx} = C1 \rightarrow F = C1x + C2$$

$$U(0,t) = 0 \rightarrow F(0) = C1 \cdot 0 + C2 = 0 \rightarrow C2 = 0$$

$$U(L,t) = 0 \rightarrow F(L) = C1 \cdot L + 0 = 0 \rightarrow C1 = 0$$

$\lambda = 0$ therefore results in a trivial solution. Assuming $\lambda > 0$:

$$F = Acos\left(\sqrt{\lambda}x\right) + Bsin\left(\sqrt{\lambda}x\right)$$

$$U(0,t) = 0 \rightarrow F(0) = Acos(0) + Bsin(0) = 0 \rightarrow A = 0$$

$$U(L,t) = 0 \rightarrow F(L) = 0 + Bsin\left(\sqrt{\lambda}L\right) = 0 \rightarrow B = 0 \; or \; sin\left(\sqrt{\lambda}L\right) = 0$$

Because A = B = 0 is a trivial solution, we assumed that $sin(\sqrt{\lambda}L) = 0$:

$$sin(\sqrt{\lambda}L) = 0 \rightarrow \lambda = \left(\frac{n\pi x}{L}\right)^2, \quad n = 1,2,3 \ldots$$

F can therefore be represented by an infinite series on the integer n:

$$\therefore F(x) = \sum_{n=1}^{\infty} B_n \sin\left(\frac{n\pi x}{L}\right)$$

Our solution for U combines F and G by principle of superposition:

$$U(x,t) = \sum_{n=1}^{\infty} B_n \sin\left(\frac{n\pi x}{L}\right) e^{[-\left(\frac{n\pi}{L}\right)^2 - \frac{R(A+B)}{D}]Dt}$$

where B_n is an constant coefficient for every n. We solve for B_n by noting that at time t = 0, our solution is represented by a Fourier series. By multiplying each side by $\sin\left(\frac{m\pi x}{L}\right)$ and taking the integral from 0 to L, we can arrive at a general expression for B_n:

$$\int_0^L F(x) \sin\left(\frac{m\pi x}{L}\right) dx = \sum_{n=1}^{\infty} \int_0^L B_n \sin\left(\frac{n\pi x}{L}\right) \sin\left(\frac{m\pi x}{L}\right) dx$$

Noting the orthogonality of sines,

$$\int_0^L \sin\left(\frac{n\pi x}{L}\right) \sin\left(\frac{m\pi x}{L}\right) dx = \begin{cases} 0 \text{ for } m \neq n \\ \frac{L}{2} \text{ for } m = n \end{cases},$$

we can see that only those terms for which m=n contribute to our sum. Our expression therefore simplifies to give us an expression for B_n:

$$\int_0^L F(x) \sin\left(\frac{n\pi x}{L}\right) dx = \int_0^L B_n \sin^2\left(\frac{n\pi x}{L}\right) dx$$

$$B_n = \frac{\int_0^L F(x) \sin\left(\frac{n\pi x}{L}\right) dx}{\int_0^L B_n \sin^2\left(\frac{n\pi x}{L}\right) dx} = \frac{\int_0^L F(x) \sin\left(\frac{n\pi x}{L}\right) dx}{\frac{L}{2}} = \frac{L}{2} \int_0^L F(x) \sin\left(\frac{n\pi x}{L}\right) dx$$

Our analytical solution can now be expressed in terms of x, t, L, D, R, A, and B:

$$U(x,t) = \sum_{n=1}^{\infty} \frac{L}{2} \int_0^L F(x) \sin\left(\frac{n\pi x}{L}\right) dx \sin\left(\frac{n\pi x}{L}\right) e^{\left[-\left(\frac{n\pi}{L}\right)^2 - \frac{R(A+B)}{D}\right]Dt}$$

RESULTS

Figure 1 shows the surface plot for the analytical solution using 20 terms for the Fourier series.

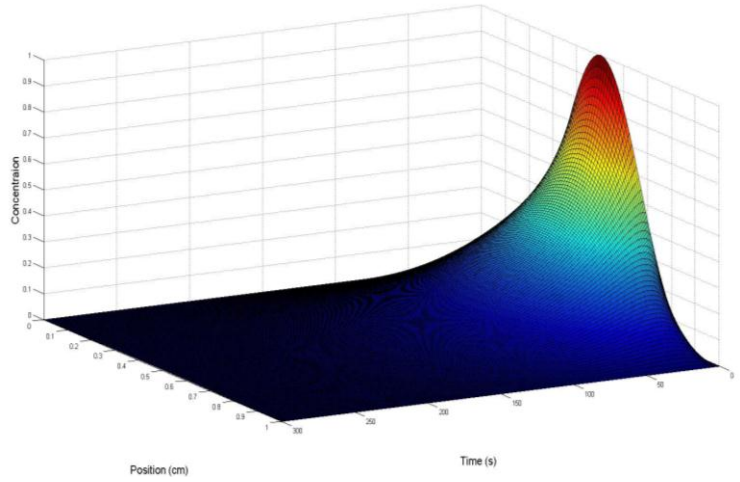

Figure 1: Surface plot of analytical solution

Four different protein distributions were then numerically simulated using MATLABs *pdepe* solver. Each case is discussed in turn.

Homogenous Distribution of Protein A/Homogenous Distribution of Protein B

Without biological cues, the distribution of proteins within a cell will approach a homogenous mixture due to simple Brownian motion. Setting the (linear) concentration of Protein A and Protein B to 0.375 for all x and t served as the baseline for calculating metabolic efficiency. **Figure 2** shows the concentration of metabolite U as a function of time and space while **Figure 3** shows the cumulative protein product as t approaches 300 s. Of note are the identical values for production of A and B (0.0905) as is expected when concentration of the proteins is identical at all points in time and space. The total amount of U consumed, derived by summing the total amount of A and B produced from t=0 to t=300, was 0.181.

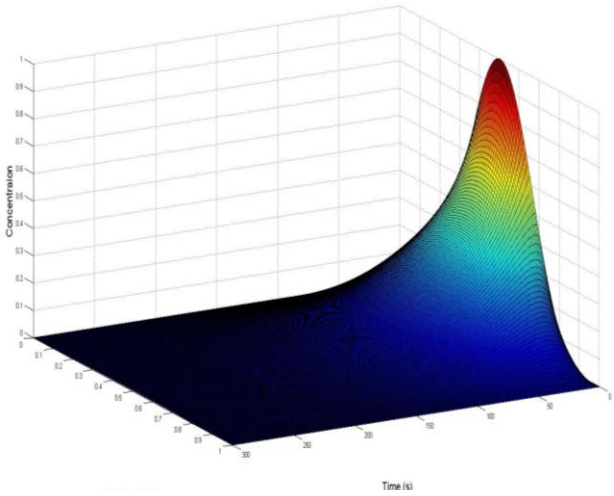

Figure 2: Spatiotemporal distribution of metabolite U with homogenous distribution of both Protein A and Protein B.

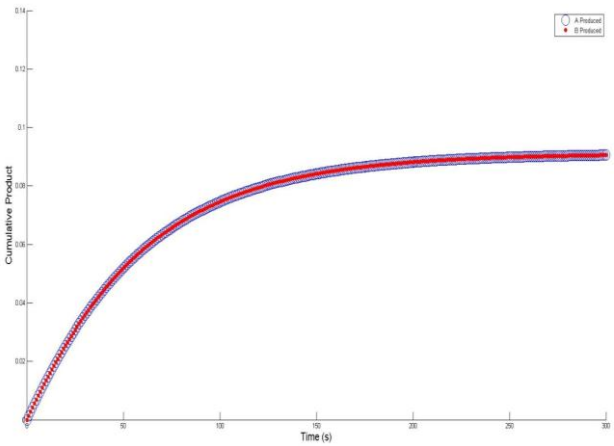

Figure 3: Cumulative flux through reactions catalyzed by Proteins A and B for the metabolie distrubtion given by figure 1.

Comparing the analytical and numerical solutions for the homogeneous/homogeneous distribution model, we found that 89.6% of points have an average difference of 0.15%, with most of the variation located at the edges.

Variable Distribution of Protein A/Homogenous Distribution of Protein B

Assuming directed genetic engineering could alter the localization of proteins within the cell, a logical target would be to cluster the protein catalyzing the reaction of interest near its substrate. A time independent concentration profile for Protein A was set such that [Protein A]=$\sin^4(\pi x)$ for all t. This allowed the total amount of Protein A to be equal to the total amount of Protein B for all t. Protein B levels were kept at 0.375 for all x and t. **Figure 4** depicts the protein distribution. **Figure 5** shows the concentration profile of metabolite U for this protein distribution. **Figure 6** illustrates the resulting cumulative flux split for the cell. The total amount of U consumed was 0.1967. Of this, 0.1176 was converted to A and 0.0791 to B.

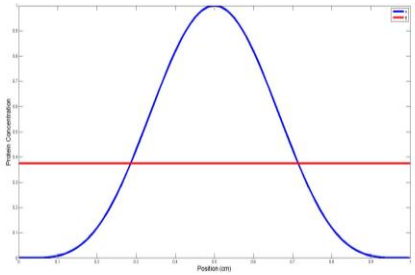

Figure 4: Protein distribution when Protein A has been localized and Protein B remains homogenous. Total areas under each curve are equal.

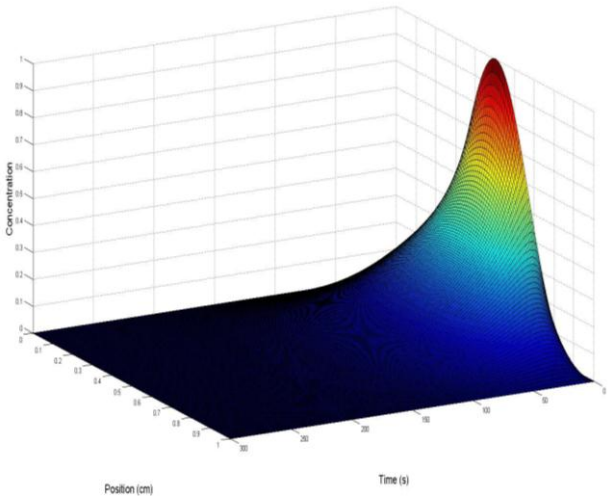

Figure 5: Spatiotemporal distribution of metabolite U with distribution of proteins in accordance with figure 3.

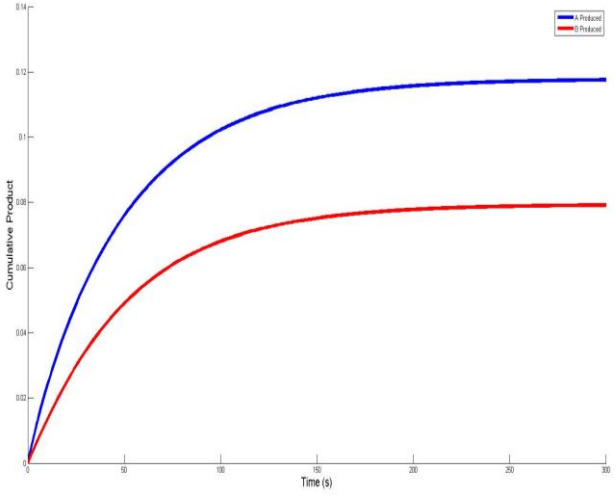

Figure 6: Cumulative flux through reactions catalyzed by Proteins A and B for the metabolie distrubtion given by figure 4.

Variable Distribution of Protein A/'Inverse'Distribution of Protein B

The third set of distributions considered probed instances where the transcription and translation of Protein A inhibited that of Protein B. Biologically this would occur if Protein A were in an operon that contained an enzyme that negatively regulated Protein B. The resultant protein concentration profiles (again keeping the total amount of each protein invariant) were [Protein A]=$\sin^4(\pi x)$ and [Protein B]=0.6*[1-$\sin^4(\pi x)$] as seen in **Figure 7**. The accompanying concentration profile for U and flux split through the proteins are shown in **Figure 8** and **Figure 9**, respectively. The total amount of U consumed was 0.1881. Of this, 0.1282 was converted to A and 0.0599 to B.

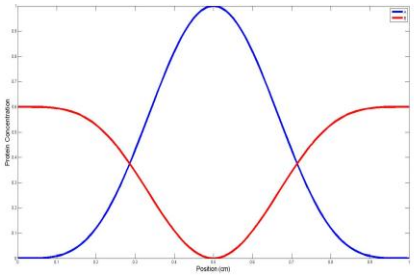

Figure 7: Protein distribution when Protein A has been localized and Protein B concentration is inversely proportional to that of Protein A.

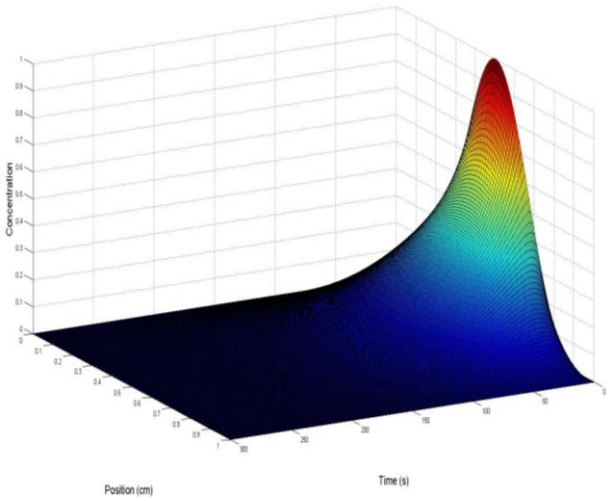

Figure 8: Spatiotemporal distribution of metabolite U with distribution of proteins in accordance with figure 6.

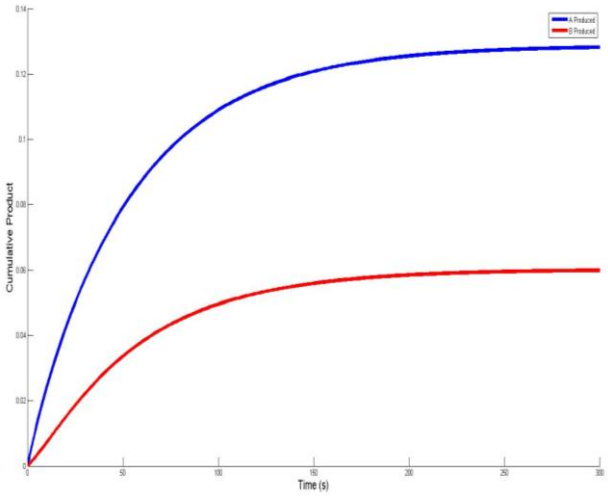

Figure 9: Cumulative flux through reactions catalyzed by Proteins A and B for the metabolie distrubtion given by figure 7.

Variable Distribution of Protein A/Variable Distribution of Protein B

The final distribution simulated was that of forced localization of both proteins. [Protein A] remained equal to $\sin^4(\pi x)$ while [Protein B]=$1.875*(e^{-100x}+e^{100(x-1)})$, mimicking a protein localized as far away (e.g. the membrane) from the source of metabolite U as possible (**Figure 10**). The metabolite profile is shown in **Figure 11** and the flux split in **Figure 12**. The total amount of U consumed was 0.1734. Of this, 0.1360 was converted to A and 0.0374 to B.

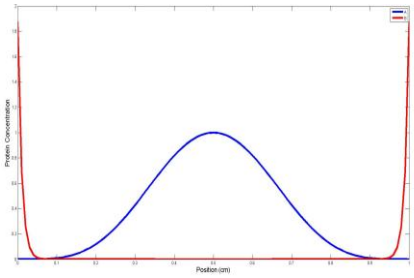

Figure 10: Protein distribution when Protein A has been localized near the source of U and Protein B has been localized away from the source of U.

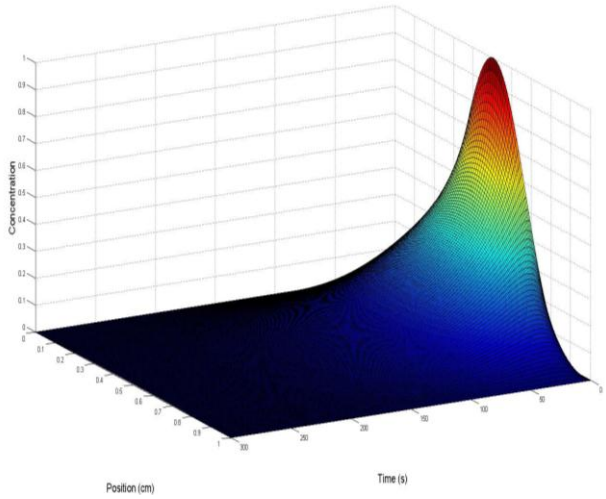

Figure 11: Spatiotemporal distribution of metabolite U with distribution of proteins in accordance with figure 9.

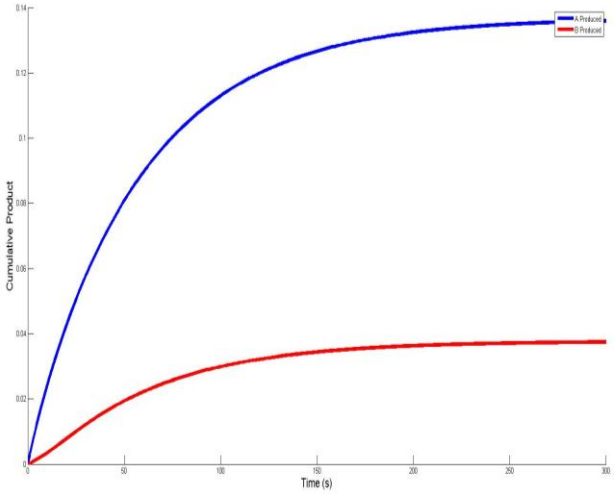

Figure 12: Cumulative flux through reactions catalyzed by Proteins A and B for the metabolie distrubtion given by figure 10.

DISCUSSION

Two important trends stand out. The first is the relationship between protein distribution and total production of A (**Figure 13**). As expected, the amount of A produced increases when Protein A is localized near the source of U and Protein B is localized away from the source of U. The most extreme localization (**Figure 10**), produces 1.5 times more A than a system where the proteins are homogenously distributed.

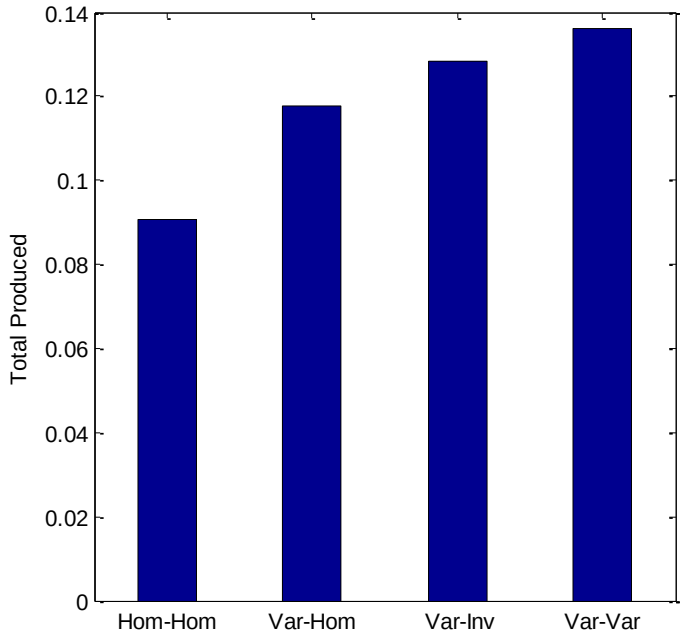

Figure 13: Total amount of A produced for various protein distributions.

Hom-Hom: Homogenous distributions of both proteins

Var-Hom: Localized distribution of Protein A, homogenous distribution of Protein B.

Var-Inv: Localized distribution of Protein A, 'inverse' distribution of Protein B.

Var-Var: Localized distribution of Protein A (near source of U), localized distribution of Protein B (away from source of U).

This points to the viability of using protein localization as a tool in metabolic engineering efforts to bypass the shunt caused by forks in metabolic pathways. Likewise, this production comes at the cost of production of B, thus care must be taken to not limit flux through the reaction catalyzed by Protein B to such an extent that the cell is not viable.

The second notable trend is the change in total U consumed (given by summing total production of A and B) with the various protein distributions. With the parameters chosen for simulation (r and [Protein] less than U for some x and t, U=0 at the boundaries), some metabolite is lost as it diffuses to x=0 or x=L without being consumed. Physiologically this corresponds to export of metabolite via diffusion. **Figure 14** shows the quantitative difference protein distribution can have on total consumption of metabolite. Of note is that maximum use of U is achieved with Protein A localized and Protein B homogenously distributed. If metabolic frugality is a goal, care must be taken to ensure that the overall metabolic capability of the cell is not sacrificed in the name of focused production of a single metabolite.

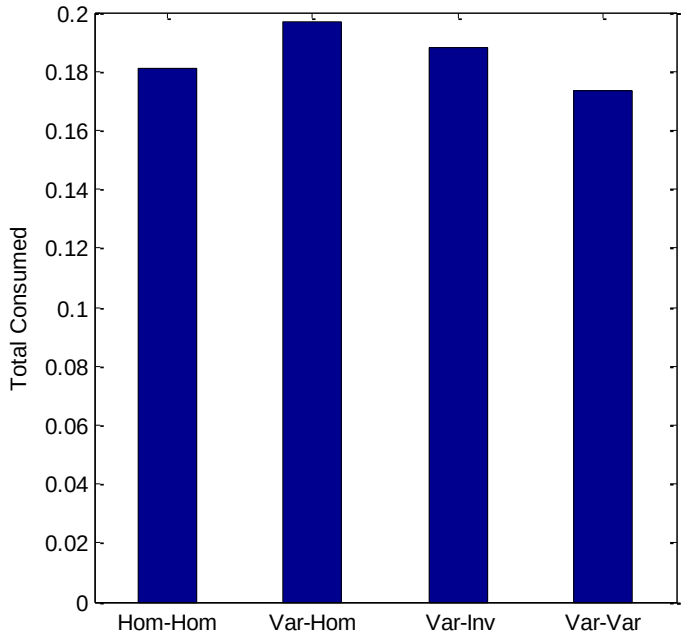

Figure 14: Total consumption of U produced for various protein distributions.

Hom-Hom: Homogenous distributions of both proteins

Var-Hom: Localized distribution of Protein A, homogenous distribution of Protein B.

Var-Inv: Localized distribution of Protein A, 'inverse' distribution of Protein B.

Var-Var: Localized distribution of Protein A (near source of U), localized distribution of Protein B (away from source of U).

CONCLUSION

By incorporating spatially dependent terms for bothmetabolite and enzyme concentrations, rather than assuming well-mixed conditions inside the cell, we hope that existing systems models can be refined to better predict cell pathway fluxes. While our parameters are not biologically accurate (e.g. R being only 1 order of magnitude larger than D rather than 4-5, the value for L, etc., etc.), our findings serve as a proof-of concept showing the utility of targeted modification of protein localization.

REFERENCES

1. *Yeung MKS, Tegner J and Collins JJ. Reverse engineering gene networks using singular value decomposition and robust regression.* Proceedings of the National Academy of Sciences USA *99: 6163-6168 (2002).*
2. *Covert MW, Schilling CH, Famili I, Edwards JS, Goryanin II, Selkov E, and Palsson BØ. Metabolic modeling of microbial strains "in silico". Trends in Biochemical Sciences 26*: 179-186 (2001).

3. *Hasty J, McMillen D, Isaacs F, and Collins JJ. Computational studies of gene regulatory networks: in numero molecular biology. Nature Reviews Genetics. 2(4): 268-279 (2001)*
4. *Personal communication with Nate Lewis*

```
   To generate analytical figure/data
function [output]=Analytical(iter)
    L=1;
    D=.001;
    A=.375;
    B=.375;
          r=.01;

          count1=1;
          for t=0:300
              count2=1;
              for x=0:.01:1
          for n=1:iter
              cn=-48*((-1)^n-1)/(pi*(n^5-20*n^3+64*n));
              if isnan(cn)==1
                  cn=0;
              end
              if n==1
                  val2=cn*sin(n*pi*x/L)*exp((-(n*pi/L)^2-
   (A+B)*r/D)*D*t);
              else
                  pd=cn*sin(n*pi*x/L)*exp((-(n*pi/L)^2-(A+B)*r/D)*D*t);
                  val2=val2+pd;
              end
          end
          val=val2;
          output(count1,count2)=val;
          count2=count2+1;
              end
              count1=count1+1;
          end
          tt=0:300;
          xx=0:.01:1;
          figure
          surf(tt,xx,output')
          xlabel('Time (s)','FontSize',16)
          ylabel('Position (cm)','FontSize',16)
          zlabel('Concentraion','FontSize',16)
          axis([0 300 0 1 0 1])
      end


      To generate all numerical figures/data
      m=0;
      x=linspace(0,1,101);
      t=linspace(0,300,301);
      sol_none=pdepe(m,@pdex_none,@pdexic,@pdexbc,x,t);
      %figure,surf(t,x,sol_none')
          %xlabel('Time (s)')
          %ylabel('Position (cm)')
          %zlabel('Concentration')
          %title('No Consumption')

      sol_hom=pdepe(m,@pdex_hom,@pdexic,@pdexbc,x,t);
      figure,surf(t,x,sol_hom')
          xlabel('Time (s)','FontSize',16)
          ylabel('Position (cm)','FontSize',16)
```

```
60.           zlabel('Concentraion','FontSize',16)
61.               axis([0 300 0 1 0 1])
62.
63.           %title('A=B=Homogenous')
64.
65.       for i=1:length(t)
66.       for j=1:101
67.       Aout(i,j)=.375*.01*sol_hom(i,j);
68.       Bout(i,j)=.375*.01*sol_hom(i,j);
69.       end
70.       end
71.
72.       figure
73.       %plot(t,.01*sum(Aout,2))
74.       hold
75.       %plot(t,.01*sum(Bout,2),'r')
76.       %title('hom hom')
77.       for i=1:length(t)
78.           pd1=.01*sum(Aout,2);
79.           pd2=.01*sum(Bout,2);
80.           test1(i)=trapz(pd1(1:i));
81.           test2(i)=trapz(pd2(1:i));
82.       end
83.
84.       homhoma=test1;
85.       homhomb=test2;
86.
87.       plot(t,test1,'bo','MarkerSize',16)
88.       plot(t,test2,'r.','MarkerSize',20)
89.       axis([0 300 0 .14])
90.       xlabel('Time (s)','FontSize',16)
91.       ylabel('Cumulative Product','FontSize',16)
92.       legend('A Produced','B Produced')
93.       total(1)=test1(end)+test2(end);
94.       disp(cat(2,'The total metabolite used in hom-hom was
    ',mat2str(total(1))))
95.
96.       sol_var_hom=pdepe(m,@pdex_var_hom,@pdexic,@pdexbc,x,t);
97.       figure,surf(t,x,sol_var_hom')
98.           xlabel('Time (s)','FontSize',16)
99.           ylabel('Position (cm)','FontSize',16)
100.          zlabel('Concentration','FontSize',16)
101.          %title('A var,B hom')
102.
103.      for i=1:length(t)
104.      for j=1:101
105.      Aout(i,j)=sin(pi*x(j))^4*.01*sol_var_hom(i,j);
106.      Bout(i,j)=.375*.01*sol_var_hom(i,j);
107.      end
108.      end
109.
110.      figure
111.      %plot(t,.01*sum(Aout,2))
112.      hold
113.      %plot(t,.01*sum(Bout,2),'r')
114.      %title('var hom')
115.      for i=1:length(t)
```

```
116.          pd1=.01*sum(Aout,2);
117.          pd2=.01*sum(Bout,2);
118.          test1(i)=trapz(pd1(1:i));
119.          test2(i)=trapz(pd2(1:i));
120.      end
121.
122.      varhoma=test1;
123.      varhomb=test2;
124.
125.      plot(t,test1,'b','LineWidth',5)
126.      plot(t,test2,'r','LineWidth',5)
127.      axis([0 300 0 .14])
128.      xlabel('Time (s)','FontSize',16)
129.      ylabel('Cumulative Product','FontSize',16)
130.      legend('A Produced','B Produced')
131.      total(2)=test1(end)+test2(end);
132.      disp(cat(2,'The total metabolite used in var-hom was
    ',mat2str(total(2))))
133.
134.      sol_var_inv=pdepe(m,@pdex_var_inv,@pdexic,@pdexbc,x,t);
135.      figure,surf(t,x,sol_var_inv')
136.          xlabel('Time (s)','FontSize',16)
137.          ylabel('Position (cm)','FontSize',16)
138.          zlabel('Concentration','FontSize',16)
139.          %title('A var, B c(1-A)')
140.
141.      for i=1:length(t)
142.      for j=1:101
143.      Aout(i,j)=sin(pi*x(j))^4*.01*sol_var_inv(i,j);
144.      Bout(i,j)=.6*(1-sin(pi*x(j))^4)*.01*sol_var_inv(i,j);
145.      end
146.      end
147.
148.      figure
149.      %plot(t,.01*sum(Aout,2))
150.      hold
151.      %plot(t,.01*sum(Bout,2),'r')
152.      %title('var inv')
153.      for i=1:length(t)
154.          pd1=.01*sum(Aout,2);
155.          pd2=.01*sum(Bout,2);
156.          test1(i)=trapz(pd1(1:i));
157.          test2(i)=trapz(pd2(1:i));
158.      end
159.
160.      varinva=test1;
161.      varinvb=test2;
162.
163.      plot(t,test1,'b','LineWidth',5)
164.      plot(t,test2,'r','LineWidth',5)
165.      axis([0 300 0 .14])
166.      xlabel('Time (s)','FontSize',16)
167.      ylabel('Cumulative Product','FontSize',16)
168.      legend('A Produced','B Produced')
169.      total(3)=test1(end)+test2(end);
170.      disp(cat(2,'The total metabolite used in var-inv was
    ',mat2str(total(3))))
```

```
171.
172.      sol_var_var=pdepe(m,@pdex_var_var,@pdexic,@pdexbc,x,t);
173.      figure,surf(t,x,sol_var_var')
174.          xlabel('Time (s)','FontSize',16)
175.          ylabel('Position (cm)','FontSize',16)
176.          zlabel('Concentration','FontSize',16)
177.          %title('A var, B var')
178.
179.      for i=1:length(t)
180.      for j=1:101
181.      Aout(i,j)=sin(pi*x(j))^4*.01*sol_var_var(i,j);
182.      Bout(i,j)=1.87509*(exp(-10*x(j))+exp(10*x(j)-
   10))*.01*sol_var_var(i,j);
183.      end
184.      end
185.
186.      figure
187.      %plot(t,.01*sum(Aout,2))
188.      hold
189.      %plot(t,.01*sum(Bout,2),'r')
190.      %title('var var')
191.      for i=1:length(t)
192.          pd1=.01*sum(Aout,2);
193.          pd2=.01*sum(Bout,2);
194.          test1(i)=trapz(pd1(1:i));
195.          test2(i)=trapz(pd2(1:i));
196.      end
197.
198.      varvara=test1;
199.      varvarb=test2;
200.
201.      plot(t,test1,'b','LineWidth',5)
202.      plot(t,test2,'r','LineWidth',5)
203.      axis([0 300 0 .14])
204.      xlabel('Time (s)','FontSize',16)
205.      ylabel('Cumulative Product','FontSize',16)
206.      legend('A Produced','B Produced')
207.      total(4)=test1(end)+test2(end);
208.      disp(cat(2,'The total metabolite used in var-var was
   ',mat2str(total(4))))
209.
210.
211.      function u0=pdexic(x)
212.      u0=sin(pi*x)^4;
213.      end
214.
215.      function [p1,q1,pr,qr]=pdexbc(x1,u1,xr,ur,t)
216.      p1=u1;
217.      q1=0;
218.      pr=ur;
219.      qr=0;
220.      end
221.
222.      function [c,f,s]=pdex_var_var(x,t,u,DuDx)
223.      c=1;
224.      A=sin(pi*x)^4;
225.      B=1.87509*(exp(-10*x)+exp(10*x-10));
```

```
226.      r=.01;
227.      f=.001*DuDx;
228.      s=-1*(A+B)*r*u;
229.      end
230.
231.      function [c,f,s]=pdex_var_inv(x,t,u,DuDx)
232.      c=1;
233.      A=sin(pi*x)^4;
234.      B=.6*(1-A);
235.      r=.01;
236.      f=.001*DuDx;
237.      s=-1*(A+B)*r*u;
238.      end
239.
240.      function [c,f,s]=pdex_var_hom(x,t,u,DuDx)
241.      c=1;
242.      A=sin(pi*x)^4;
243.      B=.375;
244.      r=.01;
245.      f=.001*DuDx;
246.      s=-1*(A+B)*r*u;
247.      end
248.
249.      function [c,f,s]=pdex_none(x,t,u,DuDx)
250.      c=1;
251.      f=.001*DuDx;
252.      s=0;
253.      end
254.
255.      To generate bar graphs
256.      bar(avals,.4)
257.      set(gca,'XTickLabel',{'Hom-Hom', 'Var-Hom', 'Var-Inv', 'Var-Var'})
258.      ylabel('Total Produced')
259.
260.      bar(total,.4)
261.      set(gca,'XTickLabel',{'Hom-Hom', 'Var-Hom', 'Var-Inv', 'Var-Var'})
262.      ylabel('Total Produced')
263.      ylabel('Total Consumed')
```