Supplementary Material (ESI) for Molecular BioSystems

Systems biology meets synthetic biology: case study of metabolic effects of synthetic rewiring Supplementary information

Josselin Noirel[1] Saw Yen Ow[1] Guido Sanguinetti[1,2]
Phillip C. Wright[1,*]

18th May 2009

[1] ChELSI Research Institute, The Sheffield Bioincubator, Department of Chemical and Process Engineering, University of Sheffield, Mappin St, S1 3JD Sheffield, United Kingdom.

[2] Department of Computer Science, 211 Portobello St, University of Sheffield, S1 4DP Sheffield, United Kingdom.

* To whom correspondence should be addressed: p.c.wright@sheffield.ac.uk.

1 Supplementary information

1.1 Maximal-flux derivation at steady state

We take again the simple toy pathway from Figure 2 (main text). We showed that if steady state is achieved,

$$[B]_\infty = \frac{A^T E_1^T k_1^{(1)} k_2^{(1)} (k_{-1}^{(2)} + k_2^{(2)})}{k_1^{(2)} \Big(E_2^T (k_{-1}^{(1)} + k_2^{(1)}) k_2^{(2)} - A^T k_1^{(1)} (E_1^T k_2^{(1)} - E_2^T k_2^{(2)}) \Big)} \tag{S1}$$

Similarly, one can derive the flux $d[C]/dt$:

$$\frac{d[C]}{dt} = E_1^T \frac{k_1^{(1)} k_2^{(1)} A^T}{k_1^{(1)} A^T + k_{-1}^{(1)} + k_2^{(1)}} \tag{S2}$$

Interestingly, it does not depend on E_2^T (or any of the second enzyme's parameters, for that matter) but does on E_1^T and if steady state could always be achieved, maximal flux would be reached when E_1^T is maximal. If we keep the total amount of enzymes constant:

$$E_1^T + E_2^T = E^T$$

we would predict that maximal flux is attained when $E_1^T = E^T$ and $E_2^T = 0$. Obviously, when $E_2^T = 0$, the overall flux must be zero. What happens is that when E_2^T gets close to

zero, steady state cannot be attained because the threshold A^* tends towards zero and therefore the input A^T ends exceeding A^*. As soon as $A^T \geq A^*$, B accumulates (as seen in the section 'Ensuring flux balance') and $[B]_\infty = +\infty$. Since the substrate of the second enzyme is hyperconcentrated, the flux $d[C]/dt$ is the $v^{(2)}_{\max}$ of the second enzyme, that is,

$$k_2^{(2)}(E^T - E_1^T) \tag{S3}$$

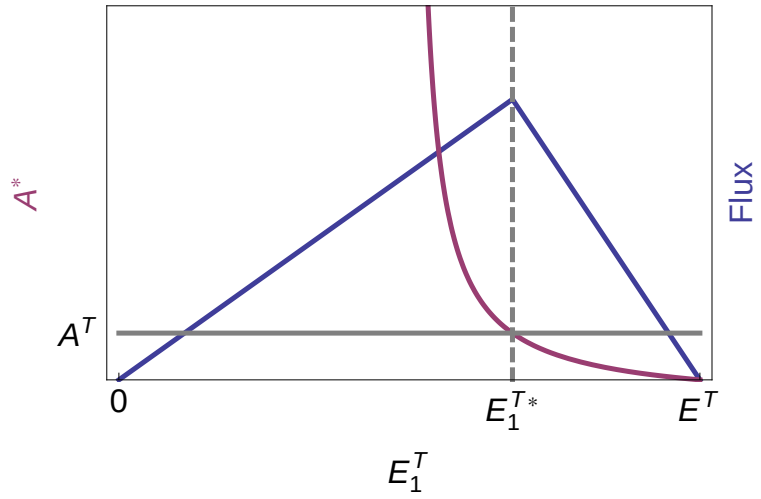
Maximum flux is therefore attained when $A^T = A^*$. So we can solve the equation

$$A^T = \frac{(E^T - E_1^T)(k_{-1}^{(1)} + k_2^{(1)})k_2^{(2)}}{k_1^{(1)}(E_1^T k_2^{(1)} - (E^T - E_1^T)k_2^{(2)})} \tag{S4}$$

We arrive at:

$$\frac{E_1^{T*}}{E^T} = \frac{k_2^{(2)}(k_1^{(1)}A^T + k_{-1}^{(1)} + k_2^{(1)})}{k_2^{(2)}(k_{-1}^{(1)} + k_2^{(1)}) + k_1^{(1)}A^T(k_2^{(1)} + k_2^{(2)})} \tag{S5}$$

The maximal flux depends on many parameters, including the input value A^T. We can summarise this result with the following plot:



As E_1^T moves from zero to E_1^{T*} (the value for which A^T equates the threshold A^*) the flux augments linearly and steady state is mathematically achievable. Above this value, E_2 becomes limiting. The maximal flux is obtained at $E_1^T = E_1^{T*}$, which will always be greater than 1/2 if $k_2^{(1)} = k_2^{(2)}$.

1.2 Supplementary figures

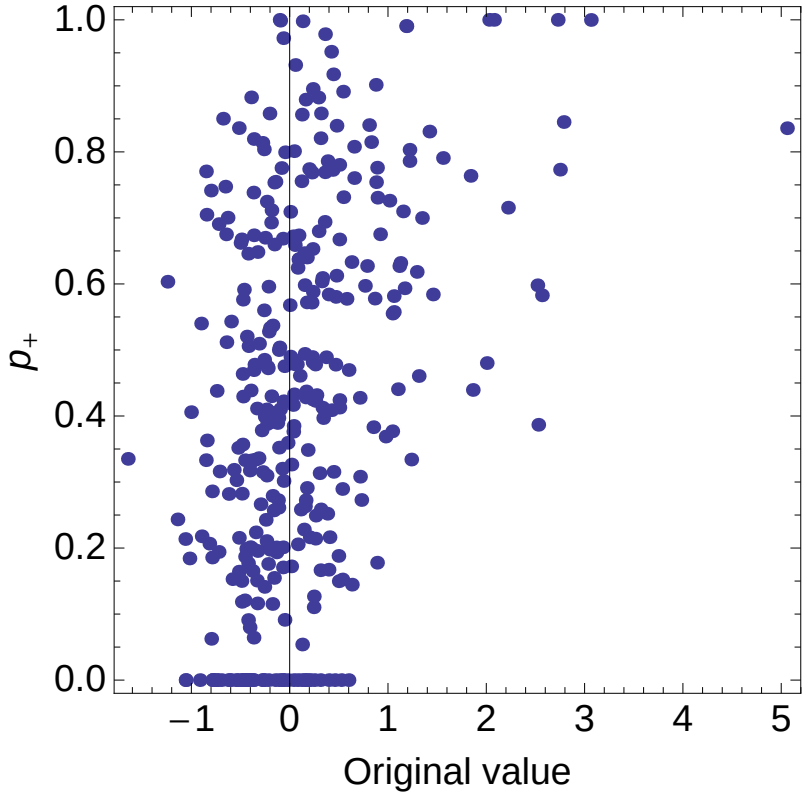


Figure S1: Newman et al.'s 2006 study presents a proteomic dataset obtained by measuring GFP fluorescence. The authors compared yeast cells grown in (a) rich and (b) minimal media. The dataset was made available as supplementary material and was used to conduct a validation of MMG [20]. The dataset was copied and modified 350 times into 'mutant datasets' (MD). For each MD, a datapoint x was removed so as to create 350 different MDs. MMG was run independently on each MD and used to determine the probabilities p_-, p_+, and p_0 for the missing datapoint to be down-regulated, up-regulated, and unchanged, respectively. These probabilities were then compared to the original value x. Here is represented p_+ as a function of x (log-ratio), it shows that high probabilities p_+ of being up-regulated are associated with higher measurements x. This shows that genes predicted to up-regulated are likelier to be associated with high measurements in the original dataset.

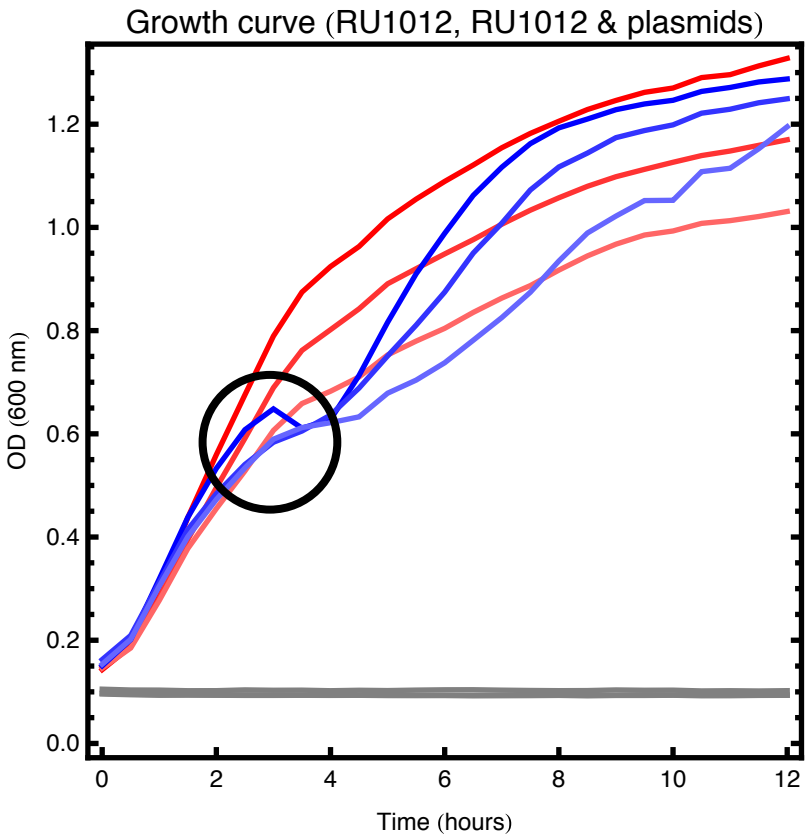


Figure S2: Growth curves of the strain RU1012 (red) and the engineered strain (blue). The time when the cells were harvested is indicated by a black circle.

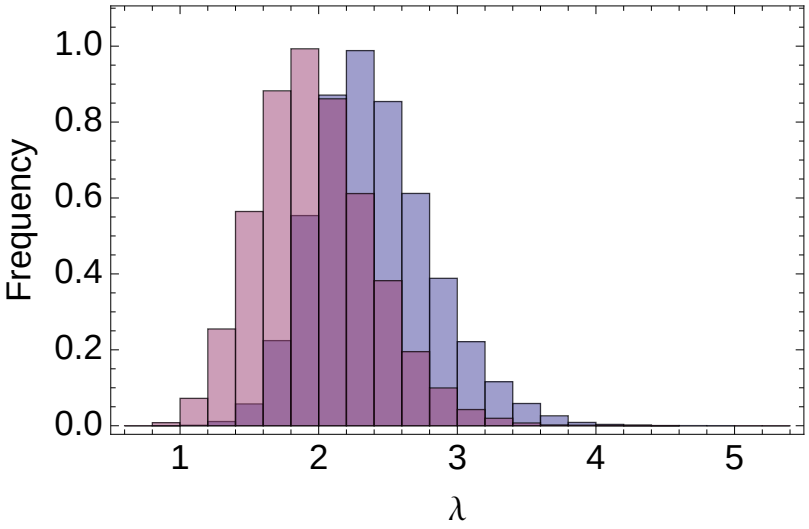


Figure S3: MMG assumes that differential expression can be modelled by a positive or negative exponential distribution, depending on whether one considers up- or down-regulation. The parameters of the exponential distribution were called λ_- and λ_+. Their values indicate the strength of differential expression, lower values hinting at more noticeable changes. The posterior distributions of these parameters are represented here. Blue: λ_-; magenta: λ_+.

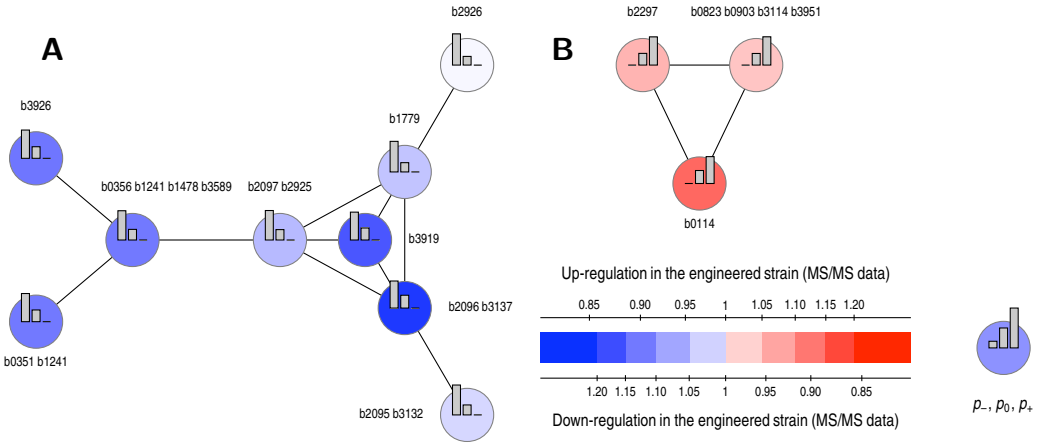


Figure S4: Up- and down-regulated networks labelled with the gene names rather than the enzyme descriptions.

Node	$p_{(-)}$	$p_{(0)}$	$p_{(+)}$	Measurement	Function
Down-regulated (in this study)					
477	0.63	0.37	0	-0.540	glycerol kinase
2	0.64	0.36	0	-0.312	alcohol dehydrogenase class III
509	0.60	0.40	0	-0.312	acetaldehyde-CoA dehydrogenase II
9	0.69	0.31	0	-0.843	fructose-bisphosphate aldolase class I/II
8	0.67	0.33	0	-0.785	triosephosphate isomerase
7	0	0.43	0.57	+0.396	glyceraldehyde-3-phosphate dehydrogenase A
99	0.35	0.33	0.31	NA	glucitol/sorbitol-specific enzyme IIC component of PTS
19	0.86	0.14	0	-1.247	phosphoglycerate kinase
98	0.34	0.33	0.33	NA	D-tagatose 1,6-bisphosphate aldolase
Up-regulated (in this study)					
400	0.56	0.44	0	-0.124	phosphate acetyltransferase
503	0	0.45	0.55	+0.382	predicted pyruvate formate lyase
3	0	0.54	0.46	+0.560	pyruvate dehydrogenase

Table S5: Re-analysis of the *cis*-1,2-dichloroethylene-degrading strain [18] based on the networks identified in this study. The posterior probabilities are reported $p_{(-)}$, $p_{(0)}$, and $p_{(+)}$. The measurements from the 2006 study are reported in $\log_2$ ratios. 'NA' indicates that the protein was not quantified.

1.3 Supplementary files

File S6: `QFluxConstant` Mathematica notebook showing the dynamics of a toy enzymatic pathway. It can be viewed using MathematicaPlayer which may be downloaded from `http://www.wolfram.com/products/player/`.

File S7: `AllQuants` Proteins quantified in this study. The file include the protein name, the quantifications, and the error factors.

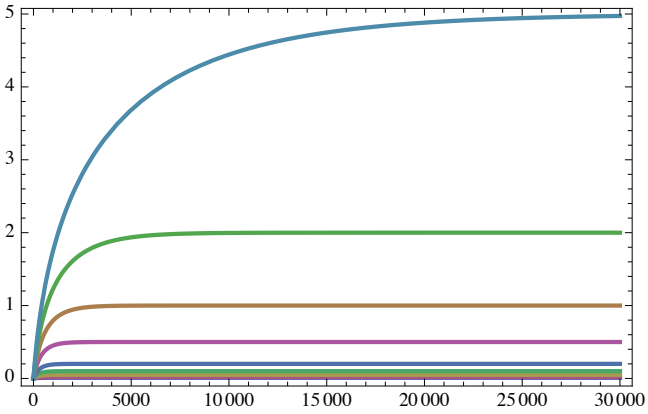
```
Needs["PlotLegends`"]
```

Varying the parameters

```
twoEquations[E1t_, E2t_, at_] := With[{k1 = 0.5, k2 = 0.5,
    l1 = 0.5, l2 = 0.5, m1 = 0.05, m2 = 0.05},
  NDSolve[{
    E1a'[t] == k1 E1[t] a[t] - (l1 + m1) E1a[t],
    E2b'[t] == k2 E2[t] b[t] - (l2 + m2) E2b[t],
    b'[t] == m1 E1a[t] - k2 E2[t] b[t] + l2 E2b[t],
    c'[t] == m2 E2b[t],
    E1a[t] + E1[t] == E1t,
    E2b[t] + E2[t] == E2t,
    b[0] == 0,
    c[0] == 0,
    E1a[0] == 0,
    E2b[0] == 0,
    a[t] == at
   }, {E1, E1a, a, E2, E2b, b, c}, {t, 0, 1 000 000}]]
```

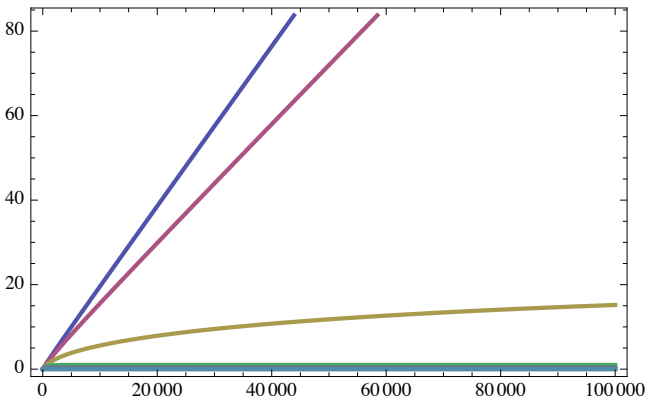
```
funcs = b[t] /. twoEquations[0.1, 0.1, #] & /@
    (0.01 {1, 2, 5}) ~Join~ (0.1 {1, 2, 5}) ~Join~ ({1, 2, 5});
```

```
Plot[funcs, {t, 0, 30 000}, PlotStyle → Thick, Frame → True]
```




```
funcs = b[t] /. twoEquations[0.1, #, 1] & /@
    (0.01 {1, 2, 5}) ~Join~ (0.1 {1, 2, 5}) ~Join~ ({1, 2, 5});
```

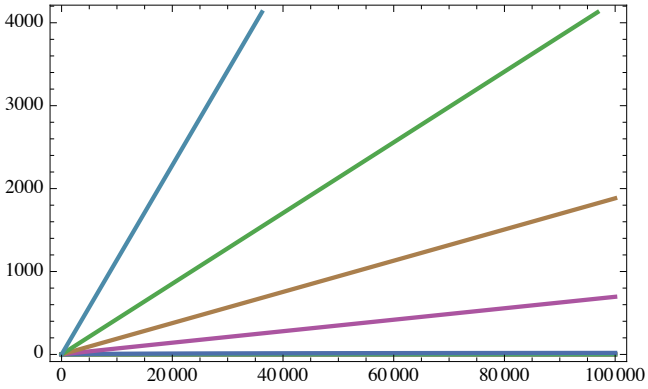
```
Plot[funcs, {t, 0, 100 000}, PlotStyle → Thick, Frame → True]
```




```
funcs = b[t] /. twoEquations[#, 0.1, 1] & /@
    (0.01 {1, 2, 5}) ~Join~ (0.1 {1, 2, 5}) ~Join~ ({1, 2, 5});
```

Supplementary Material (ESI) for Molecular BioSystems


```
Plot[funcs, {t, 0, 100 000}, PlotStyle → Thick, Frame → True]
```



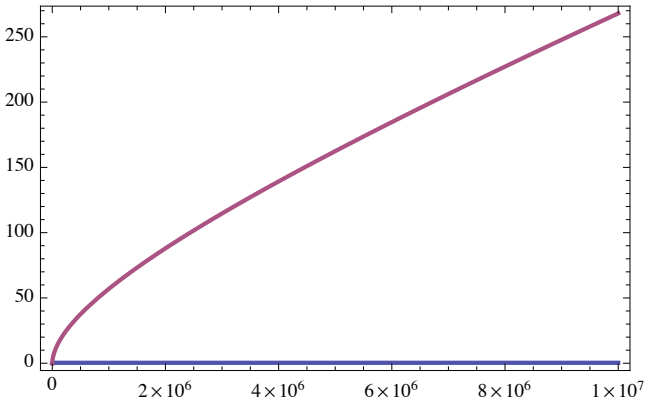
Increasing the input

■ First attempts

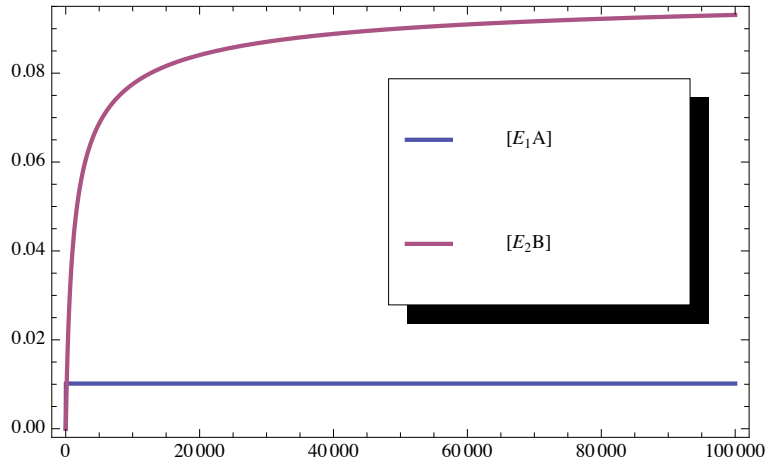
```
twoEquations[{E1t_, {k1_, l1_, m1_}}, {E2t_, {k2_, l2_, m2_}}, at_] :=
 NDSolve[{
   E1a'[t] == k1 E1[t] a[t] - (l1 + m1) E1a[t],
   E2b'[t] == k2 E2[t] b[t] - (l2 + m2) E2b[t],
   b'[t] == m1 E1a[t] - k2 E2[t] b[t] + l2 E2b[t],
   c'[t] == m2 E2b[t],
   E1a[t] + E1[t] == E1t,
   E2b[t] + E2[t] == E2t,
   b[0] == 0,
   c[0] == 0,
   E1a[0] == 0,
   E2b[0] == 0,
   a[t] == at
  }, {E1, E1a, a, E2, E2b, b, c}, {t, 0, 20 000 000}]
```

```
sols = Table[First[twoEquations[{0.05, {0.1, 0.1, 0.1}}, {0.1, {0.1, 0.1, 0.01}}, at]],
   {at, {0.1, 0.51}}];
```

```
Plot[Evaluate[b[t] /. sols], {t, 0, 10 000 000}, Frame → True, PlotStyle → Thick]
```



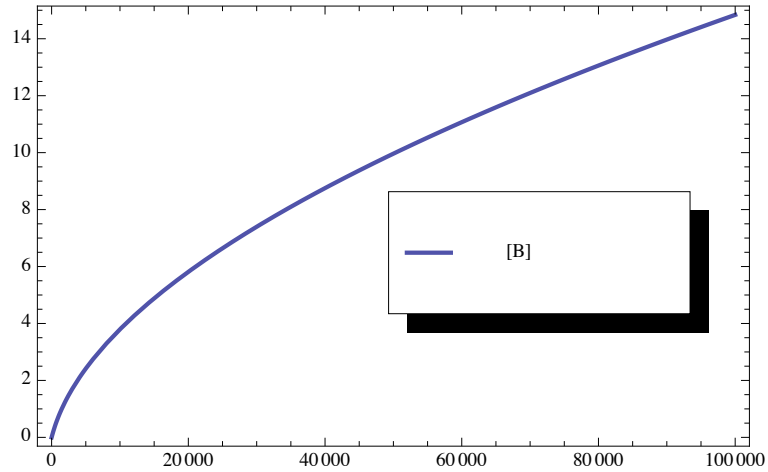
```
Plot[Evaluate[{E1a[t], E2b[t]} /. Last[sols]], {t, 0, 100 000}, Frame → True,
 PlotStyle → Thick, PlotLegend → {"[E₁A]", "[E₂B]"}, LegendPosition → {0, -0.2}]
```




```
Export["BoundEnzymes.pdf", %, "PDF"]
```

```
BoundEnzymes.pdf
```

```
Plot[Evaluate[{b[t]} /. Last[sols]], {t, 0, 100 000}, Frame → True,
 PlotStyle → Thick, PlotLegend → {"[B]"}, LegendPosition → {0, -0.2}]
```



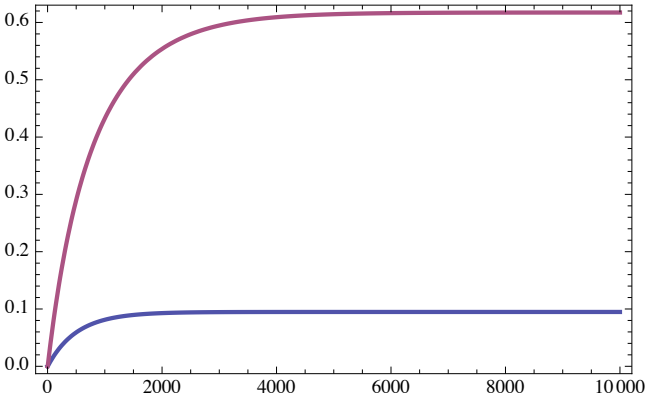

```
Export["Intermediate.pdf", %, "PDF"]
```

```
Intermediate.pdf
```

```
sols = Table[First[twoEquations[{0.05, {0.1, 0.1, 0.1}}, {0.3, {0.1, 0.1, 0.01}}, at]],
   {at, {0.1, 0.55}}];
```

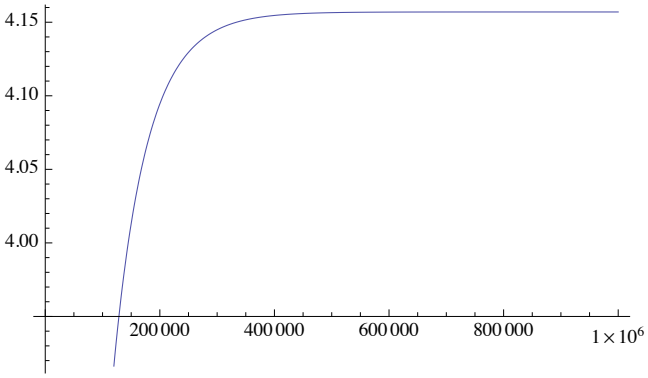
Supplementary Material (ESI) for Molecular BioSystems


```
Plot[Evaluate[b[t] /. sols], {t, 0, 10 000}, Frame → True, PlotStyle → Thick]
```

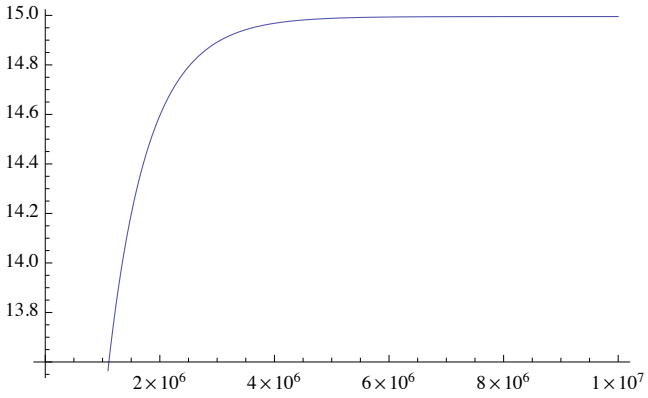


■ See 'QConstantFlux-2'

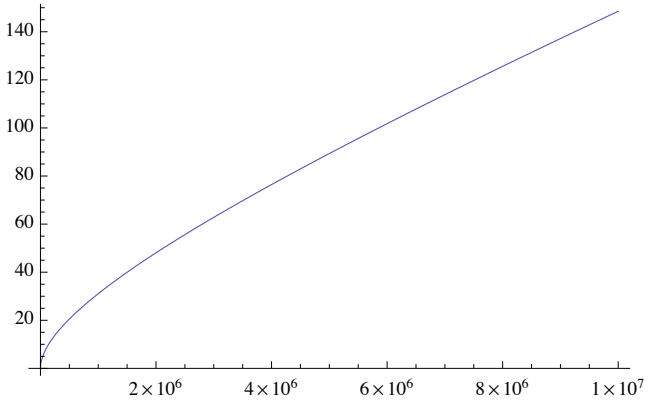
```
{e1t, e2t, k1, k2} = {0.19240115229075458`,
   0.16051415988115522`, 0.12692515175215793`, 0.14125546570683079`};
{l1, l2, m1, m2} = {0.016918527444870803`, 0.014293803605450633`,
   0.010473700443573604`, 0.01076657463893771`};

sols = twoEquations[{e1t, {k1, l1, m1}}, {e2t, {k2, l2, m2}}, 1.0];
Plot[b[t] /. First[sols], {t, 0, 1 000 000}]
```




```
sols = twoEquations[{e1t, {k1, l1, m1}}, {e2t, {k2, l2, m2}}, 1.2];
Plot[b[t] /. First[sols], {t, 0, 10 000 000}]
```



```
sols = twoEquations[{e1t, {k1, l1, m1}}, {e2t, {k2, l2, m2}}, 1.35];
Plot[b[t] /. First[sols], {t, 0, 10 000 000}]
```




```
e2t *= 1.1
```

```
0.176566
```

```
sols = twoEquations[{e1t, {k1, l1, m1}}, {e2t, {k2, l2, m2}}, 1.35];
Plot[b[t] /. First[sols], {t, 0, 100 000}]
```