***Modelling Gene Expression 1***

**INVESTIGATION USING A DETERMINISTIC MODEL**

In the following exercise we will construct and simulate a deterministic model of gene expression, and explore the effect of adding negative feedback into the system.

The first step will be to construct our model of gene expression. To do this we model the rates and reactions that occur in the following processes:

|  |  |  |
| --- | --- | --- |
| Process | Description | Rate |
|  | Gene in active state switches to gene in inactive state. |  |
|  | Gene in inactive state switches to gene in active state. |  |
|  | RNAP binds to active gene and transcribes the gene to mRNA |  |
|  | Ribosome binds to mRNA and translates the mRNA to proteins |  |
|  | Degradation/dilution of mRNA |  |
|  | Degradation/dilution of protein |  |

Initially let’s assume we are using the model to represent the expression rates in a large population of identical cells each containing a single copy of the gene.

In this case the variables in the model can be described in the following way:

the fraction of cells in which the gene is in the active state

the fraction of cells in which the gene is in the inactive state

the average copy number of mRNA transcripts from the gene in each cell

P the average copy number of the translated protein P in the cell

In this model we do not explicitly model the effect of cell division, but instead include its effect on protein and mRNA numbers through modelling a degradation/dilution process.

***Construction of the model***

*1) Use the rate expressions to complete the table below:*

|  |  |
| --- | --- |
| Species | Rate equation |
|  |  |
|  |  |
|  |  |
|  |  |

***Analysis of steady states***

*2) Analyse the equations you found to identify the steady state conditions for:*

*a) the concentration of mRNA in the system [mRNA]*

[mRNA] at steady state when:

*b) the concentration of protein in the system [P]*

[P] at steady state when

In this model the gene either exists in a fully-active or fully-inactive state. This leads to the following conservation law:

*3a) Use this equation to rewrite the rate equation for in terms of only.*

simplifying by collecting terms in

*b) Analyse the resulting expression to find the steady state value for equivalent to the fraction of genes in the active state.*

at steady state when

***Initial Conditions and Parameters***

We will use the following parameters as our initial parameters and rate constants:

|  |  |
| --- | --- |
| Model parameter | Value (seconds-1) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| Species | Initial condition |
|  | 0 |
|  | 1 |
|  | 0 |
|  | 0 |

***Analysis of model timescales***

In the model we have only assumed processes can be modelled using 1st order mass action dynamics. The rate of each process is defined by its “*k*” parameter or *rate constant*.

We can use standard mathematical relations for these to extract biologically relevant information about our system from these values.

Firstly we note that for a population that is declining with rate constant *k* the half-life can be calculated as:

half-life = ln(2)/*k*

*4) Use the parameter values provided to calculate the half-lives (in minutes) associated with:*

*a) protein degradation/dilution*

*[P]*

*b) mRNA degradation/dilution*

*[mRNA]*

*c) Comment on the timescales that you found with respect to the likely biological processes that dominate the rate in each case.*

Half-life for protein is on same order as cell generation time, indicating dilution through cellular expansion/division processes most important.

Half life for mRNA is much shorter than cell generation time indicating that the mRNA degradation dominates over dilution.

If the gene in its active state switches to its inactive state with rate *koff* the average time interval over which it remains active can be shown to be 1/*koff*.

*5 a) Use this relationship to calculate the average time interval that the gene will spend in the active state.*

*koff* =0.5 s implying average time gene remains active before swutching to inactive is 1/ *koff* = 2s

*b) A similar argument can be made for the time for the inactive gene to switch on. Use this to find an estimate for the average time interval that the gene spends in its inactive state.*

Following the same argument if with rate constant *kon* then average time taken for an inactive gene to switch on is 1/*kon* =1 / 0.1 = 10s

*c) Compare your answers to a) and b). Comment on how they relate to the steady state value for gon,*

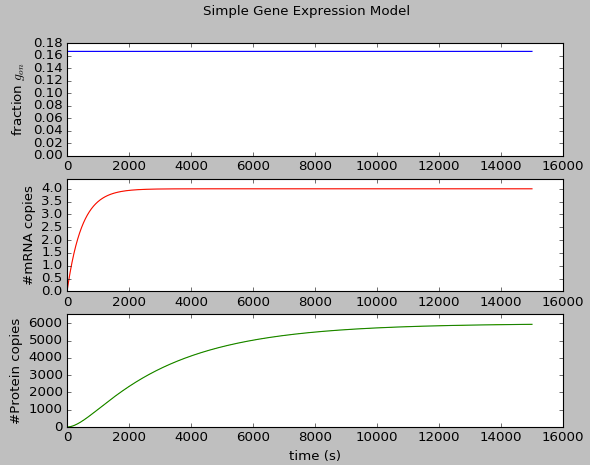
If gene remains in active state for an average of 2s and inactive state for average time of 10s then over an “average” cycle of 12s it spends 2s active and 10s inactive, i.e. fraction of time each gene is active averages to 2/12 = 0.1667.

Therefore the calculation for the steady state value of [gon] and analysis of the fraction of time each individual gene stays on give the same answer (as might be expected.

***Simulating the gene expression model***

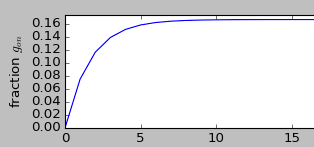
*6) Simulate the model using the template file* gene\_exp1.py

*a) Create separate plots showing the behaviour of gon, mRNA, P, and extend the simulation time until you see the whole system reach steady state.*



*Comment on the behaviour you observe, and whether it is biologically sensible.*

The fraction of active gene starts at 0 but within 10s is close to its steady state level.



As a fraction of gene is in the active state mRNA starts to be transcribed, the level starts to increase, followed by an increase in the level of protein as expected once mRNA has been generated.

Therefore qualitatively the plots follow the simple model gene in active state gets transcribed to mRNA, mRNA leads to protein production.

Furthermore the system reaches a steady state with copy numbers that do not seem unreasonable.

*b) Examine the figures produced to read off the steady states reached by the system, and compare these values to the predictions made from your analysis of the model equations.*

Steady states:

g\_on ~ 0.17

mRNA ~ 4

protein P ~ 5970

From calculations:

*gon* at steady state should be 1/6 = 0.1667 (agrees!)

mRNA at steady state when:

protein *P* at steady state when:

***Modifying the model to include negative feedback***

Suppose we want to model a situation where the protein produced acts cooperatively as a repressor for its own expression.

This can be done by introducing a Hill function term into the transcription rate expression:

|  |  |  |
| --- | --- | --- |
|  | RNAP binds to active gene and transcribes the gene to mRNA, with repression from protein P |  |

*7a) Explain how the action of this term is to model protein P as a repressor for the transcription process. e.g. explain the rate behaviour for different P levels.*

When P is zero the rate of transcription is

If P increases increases the term decreases leading to a decrease in the rate. When P = K the term so the rate is half of that observed in the absence of P.

If P is further increased the rate continues to reduce tending to zero in the limit of large P (>>K).

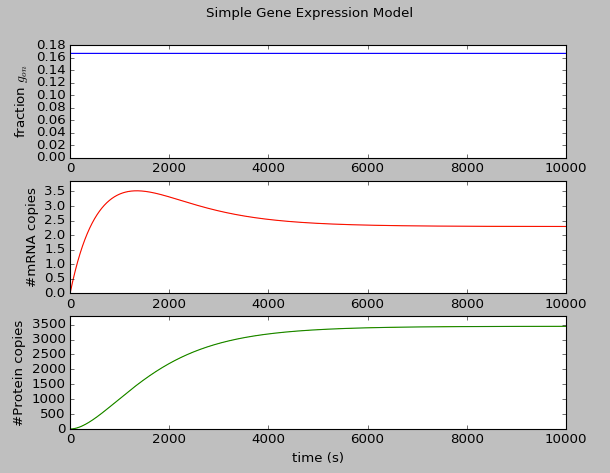
Therefore when P is present the transcription rate is reduced so P is a repressor

*b) Suggest a mechanism by which P may repress transcription.*

P could either bind to the gene and block RNAP from transcription or occupy a site on the gene that block a necessary activator from binding. Alternatively it could bind to an activating factor necessary for transcription.

*c) Make a copy of the rate function (e.g. call it* sdot\_with\_repression*) and edit it to include the repressor term into your model.*

*Simulate the behaviour using* n=2 *and* K=4000*.*

**

*Comment on how has the steady-state behaviour changed and whether this is in line with your expectations*

We expect the action of the new term is to reduce the rate of mRNA production, therefore we expect the steady state level of mRNA to be reduced once the term is added. This is observed as the new steady state is ~2.5. If the steady state level of mRNA is reduced we expect a corresponding decrease in the steady state protein level. This is also seen (it falls from ~6000 to ~3500).

In addition we might note that the mRNA now overshoots its steady state value, peaking ~30% higher at 1300s then decreasing, eventually closing near to its steady state value after about 2500s.

The effect on the protein is that it rises more rapidly, and nears its steady state value after around 7000s

***Measuring the reaction times of the system***

An important feature of gene expression networks is how quickly they react to a signal.

Suppose we assume that in addition to the processes modelled the system requires the presence of an activating signal molecule to induce transcription.

We will assume that when the signal is not present transcription cannot occur and *ktranscription* = 0.

To do this we will assume the following situation:

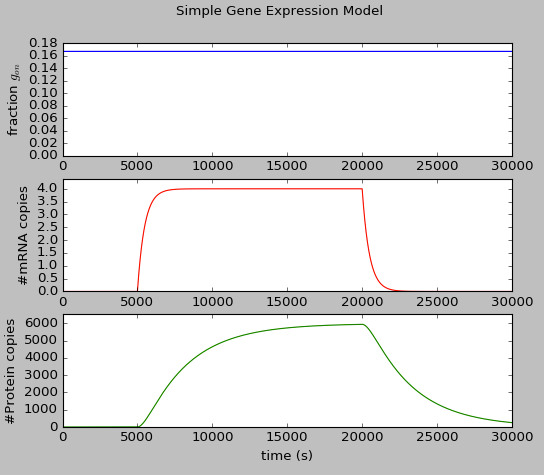
0 < t < 5000 signal off *ktranscription* = 0

5000 < t < 20000 signal on *ktranscription* = 1/20

t > 20000 signal off *ktranscription* = 0

*8a) Edit the code in the rate functions so that the value of ktranscription is dependent on the current time elapsed.*

*b) Use the original expression model (without repression term). Simulate how the gene expression system responds to the signal and examine the resulting plot of behaviour.*



The reaction time for a gene circuit measures the time taken for the system to move to its new steady-state, this is usually calculated as the time taken for the system to shift halfway to its new steady state.

*Inspect the figure to estimate the reaction time taken for the protein levels to respond to:*

*i) the signal switch on*

level rises to 3000 at approx. 7600s.

Reaction time is 7600-5000s = 2600s ( ~40min )

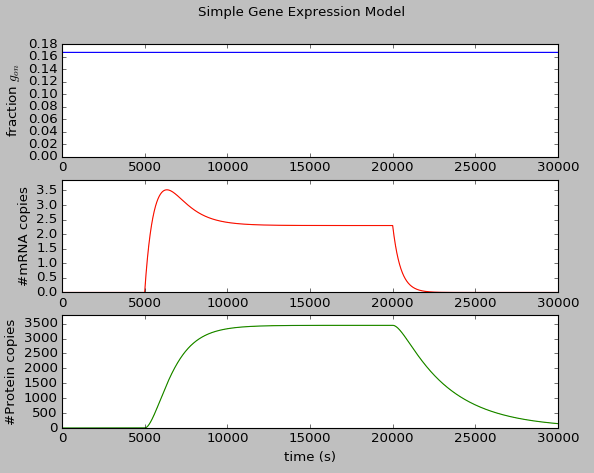
*ii) the signal switch off*

level falls to 3000 at approx. 22600

Reaction time is 22600-20000s = 2600s ( ~40min )

NB. same reaction time as for switch on

*b) Repeat the analysis for the model that includes the negative feedback via repression.*

**

*Note the system reaction time for:*

*i) the signal switch on*

Steady state at ~3460 so half this value is 1730.

Rise to 1730 at time approx 6530s

Reaction time = 6530 – 5000 = 1530s (~25min)

*ii) the signal switch off*

Falls back to 1730 after ~22600

Reaction time 22600 – 20000 = 26000s (~40min)

*c) Compare the reaction times for the two models and discuss how/why the effect could be experimentally useful to discriminate between the two types of system.*

In the simple system reaction time for protein following an action that switches-on / switches-off transcription is around 40min (a timescale similar to that of the protein degradation/dilution half life).

In the modified system the repression decreases the reaction time for the system to reach steady state following switch on (approximately by a factor of two).

However the reaction time following a switch off in transcription remains the same (40 min) This might be expected because the modification only affects the mRNA production term, and mRNA production is halted if transcription is switched off.

Experimentally we might try to measure the reaction time and detect whether the system reacts to switch-on faster than it switches-off as an indication that negative feedback could be present.

**Additional advanced analysis tasks (optional)**

**(Easy-ish)**

In your analysis of the system without negative feedback you identified steady state conditions for variables mRNA and protein P.

You should be able to combine these to write an expression for the steady state of P that is only in terms of parameters:

Check that this value is in line with the simulation output.

Our steady state conditions:

Combining the above by substitution:

This is in agreement with the previous results (effectively we did the same calculation in several stages).

**(Harder)**

You should be able to repeat this analysis for the model equations that include the negative feedback term.

In this case the steady state condition for P now involves an additional term involving parameter *K* and [P] itself!

Check that the behaviour of the system (i.e. final steady state of [P] reached) is in line with this expression.

*[mRNA] at steady state when:*

:

This leads to:

**(Even Harder)**

The above equation is difficult to solve analytically.

Write Python code to solve the equation iteratively.

# first guess is steady state level without inhibition

# we then use the expression for steady state P (that involves P)

# to calculate the next item in the sequence

# If the sequence converges to a value we know we have a solution

P=6000

for i in range(100):

print 'iteration',i,': P=',P

P\_next=6000.\*(4000.\*\*2/(4000.\*\*2+P\*\*2))

P=P\_next

Output:

iteration 0 : P= 6000

iteration 1 : P= 1846.15384615

iteration 2 : P= 4946.34146341

iteration 3 : P= 2372.34475443

…

iteration 97 : P= 3444.89610838

iteration 98 : P= 3444.89664644

iteration 99 : P= 3444.89618818

So for the model that includes repression:

[*Psteadystate*] = 3445 copies

In agreement with the simulation.