Patterns of Life HMW 6 Francesco Pancaldi
E. Coli Transcriptional Network 11/03/2014

Introduction

Many living organism functions depend on production of certain chemical polymers called proteins. The recipe for these protein is contained in the cell genetic code, called DNA, and is transmitted by m-RNA to the ribosome "protein-factories" and assembled through t-RNA.
The copy of a portion of DNA by m-RNA is initiated through the attachment of a molecule called transcription factor to a portion of DNA next to the site encoding a specific gene.
The role of the transcription factors is to integrate signals from inside and outside the cells and consequently enable the transcription of a specific portion of DNA.
Moreover, transcription factors can target other transcription factors. This connections are collected in the so called Transcription Network of the cell, containing all the interaction between different Tfs, both activators and repressors, where the first induces production of the element its acting on while the second inhibits it.
In this homework we will provide one possible implementation of the E. Coli transcription Network as provided in [1] and will observe the behavior of the solution with respect to different initial conditions and perturbation of the steady state.

Results and Discussion

A transcription network consist of all the interactions between different TFs. Usually these are visualized through graphs in which each node correspond to one specific TF and the directed edges between nodes indicate that the TF at the origin of the edge acts on the one at the end of it. Moreover, as previously stated, the interaction between TFs can either be positive or negative. A positive interaction of A on B will increase the quantity of B and A will be called an activator, while a negative interaction of A on B will decrease the quantity of B and A will be called a repressor. Furthermore, some TFs can act both as repressor and activators in this case we will say that A as a dual action on B. The strength of the connection usually varies from connection to connection, but since in the data provided in [1] no weight were included we will considered each connection to have the same strength (as an absolute value) and will only distinguish between activation, repression and dual effect.
An other important element of these interactions is the fact that the interaction are significant only if the quantity of A is big enough (after a threshold K). We can think of this either us an Hill function or a logic circuit in which the value of the function is close to zero (or zero) before a value K, and increases sharply after that value (or has a jump) to a non zero value (positive for activation negative for repression). In our implementation we will use logic functions with the same K value for all the TFs, and for both activation and repression we have 0 before K and 1 or -1 after, while for the dual case we have -1 before K and 1 after (similar results are obtain using 1 before K and -1 after).
The data provided in [1] consist of three column vectors where the first column carry the operon number the second the transiction factor number and the last the type of interaction activation, repression or dual. There are 578 interaction for 423 TFs in the database.
Our matlab code first converts the 578x3 matrix given by [1] in a 423x423 matrix in which each row represent one operon and the entries in the column represent the interaction of that specific columnTF on the row operon, with 0 no connection, 1 activation, 2 repression, 3 dual.
Then we derive a set of 423 ODEs (one for each operno) in 423 variables (one for each TF) using the connection matrix to determine the rate change for the concentration of each operon as the sum of all the interaction with the TFs (through switch functions as previously described) and the degradation of the operon (this is kept constant and the same for everyone).

Finally we are solving the ODEs system using ode45 with the “NonNegative” option ON (this keeps our solution non negative since we are working with molecule concentrations) and a random initial condition.
We can see in the following picture an example of solution.

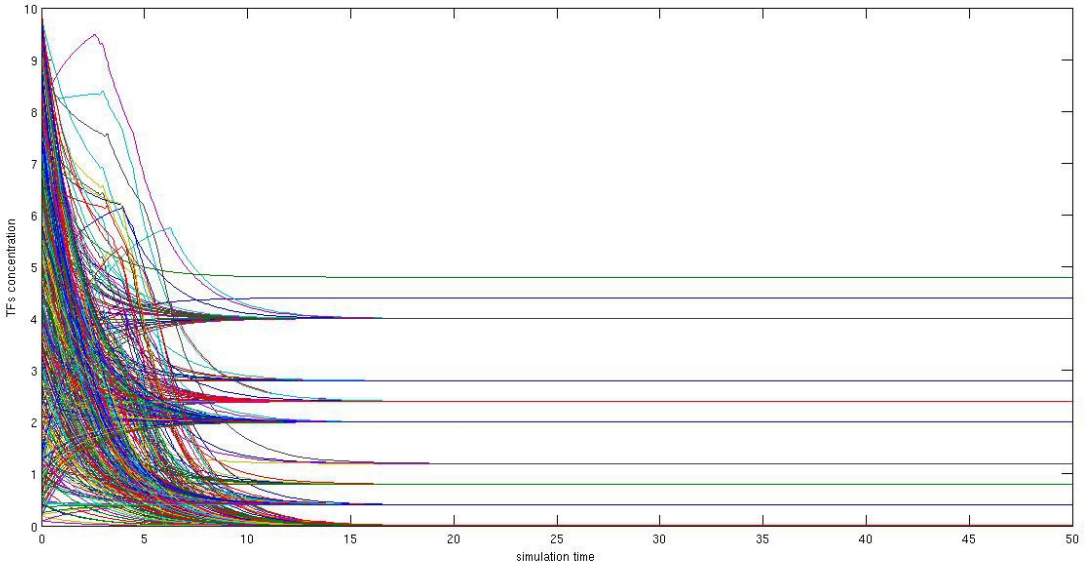

Illustration 1: Network simulation: k=1.2, alpha=0.5 (degradation rate), beta=1 (connection strength), initial condition random from uniform distribution between 0 and 10.

Repeating the simulation with different random initial condition does not seem to change significantly the qualitative behavior of the system (see Fig. 2). There seem to be an almost fixed number of steady state whose location does not change. However, repeating the simulation it is possible to find a relatively small number of this state appearing or disappearing.

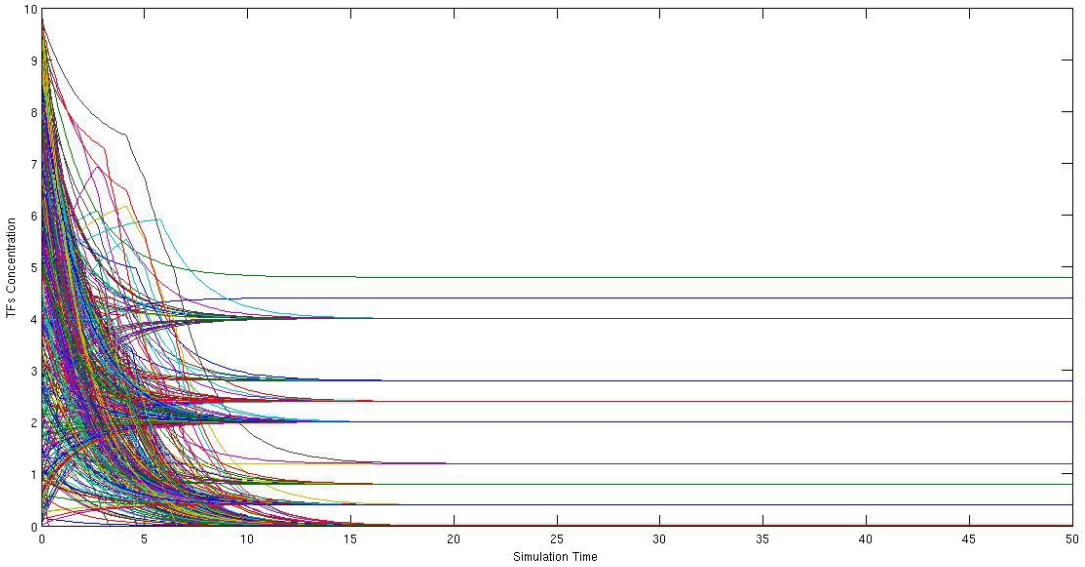

Illustration 2: Network simulation: all parameters as in Fig.1, but different random initial condition

Now we want to observe how perturbation alters the solution of our network. To see this we will apply first a small amount of noise to all the final concentrations at time t=50 and let the system evolve to a new steady solution, and secondly apply a significantly big change to only one of the concentration at t=50 and observe the resulting solution.

For the first case we decided to apply a noise from a random uniform distribution between -err and +err.
In the following figure we are showing the result for err=1.

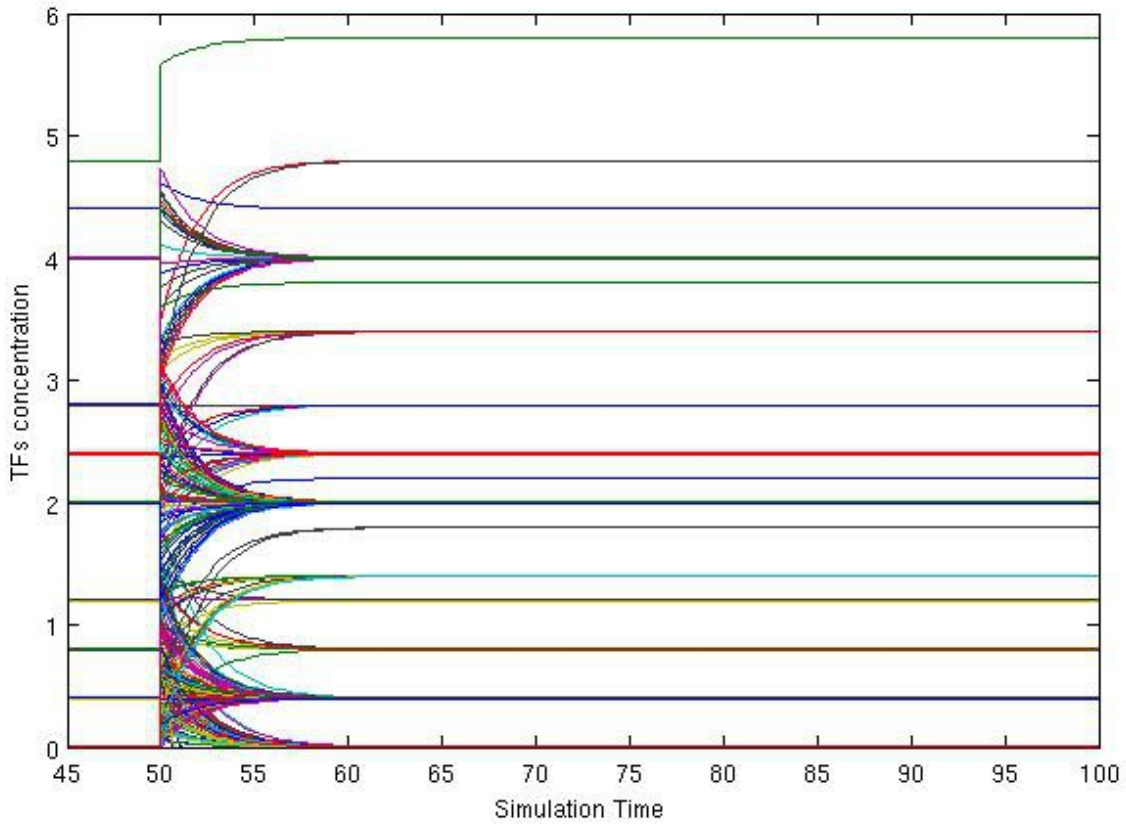

Illustration 3: Perturbation of all solution values with uniform distribution between -1 and +1
We can easily observe that this type of perturbation results in a significant change in the final solution. Some steady state are shifted to a completely different value (either already present or not) and some of them completely disappear (see top green line), while other after being perturbed quickly reset to their original value (first blue line from the top).
This behavior is preserved also when the strength of the noise is reduced up to several order of magnitude as low as err=10^-11 as seen in Fig.4, while lower values seem to have no effect on the final solution. However, when reducing the magnitude we see less jumps to different steady states.

Secondly we applied noise to a single TFs solution decreasing its concentration to zero (we select one TF with steady state solution significantly different from 0).
This does not seem to generally have much effect on the steady state solution as can be seen in Fig. 5. However, some selected node produce a behavior similar to the one observed in Fig.3 and 4 (see Christopher Eschb [4] for TF 66). We think that this is due to the high number of outward connections of this specific nodes, and/or the type of connections.

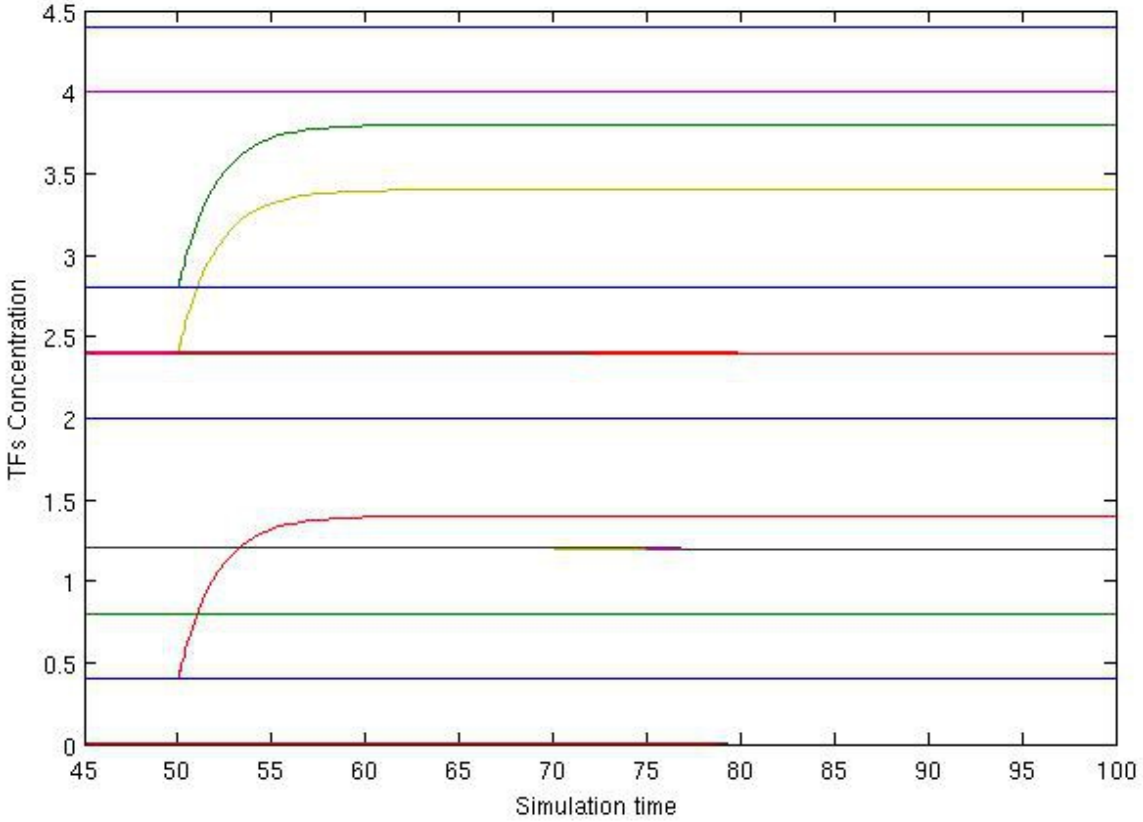

Illustration 4: Perturbation of all solution values with uniform distribution between -10^-11 and 10^-11

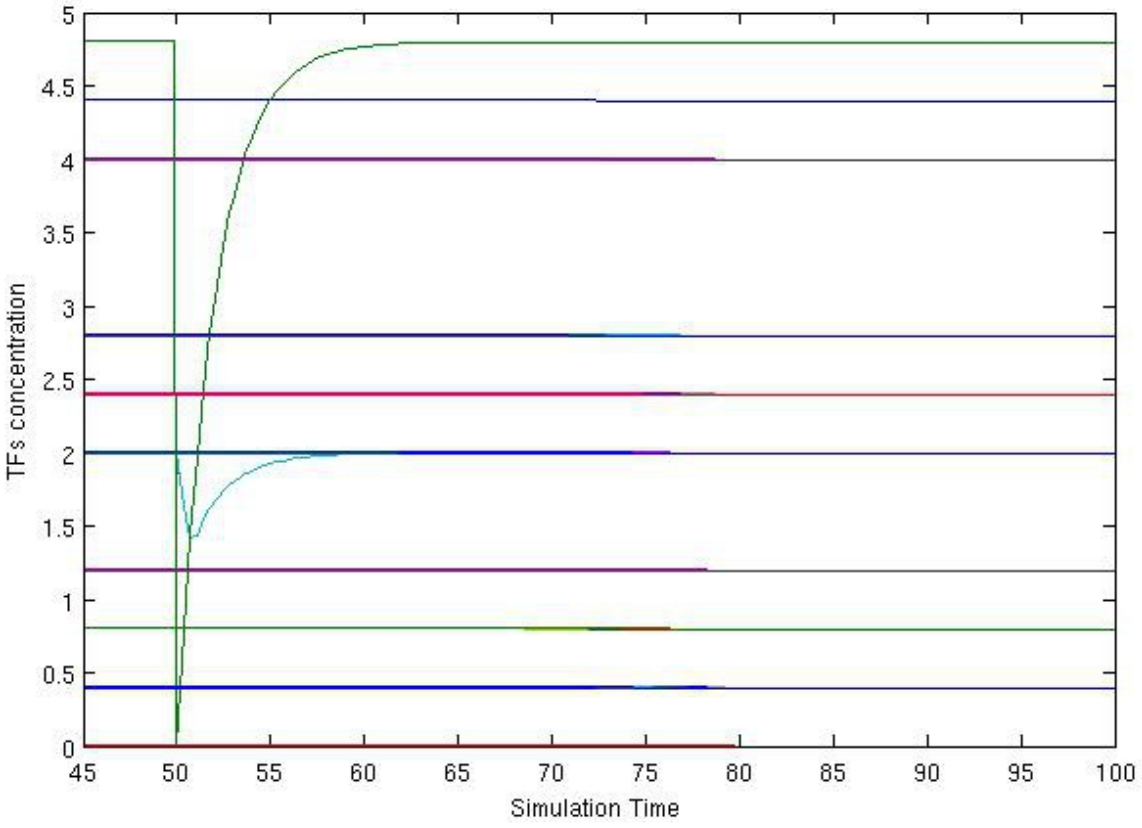

Illustration 5: Perturbation of one solution to 0

We also tried to perturb one single solution to some higher positive value (like 10 or 100) obtaining the same result as if Fig. 5 (see Fig. 6). Moreover, in this case we were not able to find exception as in the previous case.

Conclusion

Our implementation of the Transcription Network of E. Coli is very simplified, lacking diversity in strength, thresholds and degradation for each connection and Tfs, but was nevertheless able to produce some interesting computational results.
We saw how the network seems to be filtering completely unrelated random initial conditions to almost completely predetermined steady state solution. This effect is most probably due to the particular structure of the E. Coli network. Future work may try to see how this changes with a more random network.
We also observed that even a very small noise applied to all the Tfs concentration can produce significant change in the steady state, where some steady state jump to new values while others return to their original steady state. A more close analysis of different nodes with different number and type of inward and outward connections will be very interesting in this case. Also we should try to verify if random networks are as sensible as the E. Coli network.
Finally, we found that forcing a positive jump in the concentration of a single Tfs does not seem to produce significant effect. This is probably due at least partially to the choice of step functions to simulate the action of one node on another one, but the structure of the network may be also producing some stabilizing effect. Again we want to check what happens for random networks.
We also tried to remove one TF completely setting its concentration to 0. This behaved similarly to the positive jump when selecting a random TF, but seem to behave more like the diffused noise for some specific nodes (either with a lot of outward connection or with some specific type of connection with loosely connected nodes, see [4] for more details).
Applying this same model to different networks, either random or from other organism, would be interesting to understand more the inherent qualities of the model itself and the characteristic emerging from a naturally evolved network.

This worked was partially produced in collaboration with Christopher Eschb and [4] can be considered a complement to it.

References

[1] Nature Genetics 31, 64 - 68 (2002)
Published online: 22 April 2002; | doi:10.1038/ng881
Network motifs in the transcriptional regulation network of Escherichia coli

Shai S. Shen-Orr1, Ron Milo2, Shmoolik Mangan1 & Uri Alon1, 2

[2] Traffic. Author manuscript; available in PMC Aug 26, 2008.
Published in final edited form as:
Traffic. Jan 2005; 6(1): 18–31.
doi: 10.1111/j.1600-0854.2004.00251.x
PMCID: PMC2523259
NIHMSID: NIHMS38408
Covert Operations of Uropathogenic Escherichia coli within the Urinary Tract
Jean M. Bower,¶ Danelle S. Eto,¶ and Matthew A. Mulvey*

[3]Nature. 2004 May 6;429(6987):92-6.
Integrating high-throughput and computational data elucidates bacterial networks.
Covert MW1, Knight EM, Reed JL, Herrgard MJ, Palsson BO.
[4] Christopher Eschb Homework 6 on E. Coli Transcriptional Network.
[5] Lectures and Discussion with Dr. Vural 11/03/2014