A. Lane

A gene regulatory network can be described as a system of equations in which any gene , where is a set of all genes in a network, arbitrarily depends on gene such that for non-self-regulating genes. As such, a matrix can be used to describe the set of interdependencies between the sets of genes. This such matrix will be denoted as , with dimensions and is defined as such:

We will denote as the gene expression level of gene, where is a function of all. Furthermore, a steady state of a regulatory network corresponding to can be described as a set of functions. The steady state of said network, when expressed as a system of linear equations, can be described as matrix multiplication. We’ll denote matrix with same dimensions as, so that (where), and gene expression vector such that , where is a nonzero vector. For clarity, a nonzero value in indicates that the expression level for gene depends on gene by a factor of and implies. The following proof can be induced to find the steady state. Assume is an identity matrix.

However, for many purposes the regulatory network is unknown and thus must be found. One such way of finding is through a series of perturbations unto. Doing so will yield new steady states which can be observed to estimate. For the case of linear systems of equations, can also be found (or estimated, if noise is present). will be denoted as the steady state after perturbing gene by amount such that

.

The following proof will be used to show that:

Asserting that gene is perturbed by amount so that , and that is the only possible nonzero element of row of , and asserting for the following, the perturbation of gene can be described as:

This concludes the proof of and that the above equation holds for the perturbation when gene is perturbed by for the equation:

It will next be shown that genes will also be perturbed using a similar equation, without the constraints: , lest being singular.

We’ll denote: , ,

This concludes the proof of all perturbations in a linear regulatory network.

Once the perturbations on all genes have been performed and their deviations from the steady state measured, they can then be used to recover. We’ll denote these deviations as an matrix where is the initial steady state.

(proof of correctness needed)

For nonlinear networks however, linear algebra is not sufficient enough to find the steady-state of and new methods must be used. The following set of nonlinear functions will be used to describe the steady state of a nonlinear network.

Where denotes a set of indices of all nonzero values of row of matrix , denotes the power set of , denotes the relative activation of and is within the domain , and , where K and N are matrices who’s nonzero elements correspond to the nonzero elements of matrix . The values of and are the dissociation constants and Hill coefficients, respectively. Here, a steady state can be found by converging an arbitrary vector toward by using the property: such that:

For the nonlinear case, can also be estimated by inducing perturbations on all genes as with the linear case. What differs, however, is the method of their induction. We’ll again denote as the steady state of the expression levels of the regulatory network when perturbing gene by . The perturbations can be simulated by asserting . We can take advantage of the nonlinear convergence property to find: such that: for all iterations . As with the linear case, the deviations are: . They can be used to again, apply . This, however, is a naïve approach due to the nonlinear regulatory network being expressed as a linear system of equations. However, the perturbations may imply that . This can be used to estimate , but not with high precision (especially with cases of cycles in the regulatory network) by using the contrapositive:

We’ll next take into account (for both linear and nonlinear cases) a level of molecular noise, which can be described by a Gaussian random variable of a normal distribution with mean 0 and standard deviation .

This will cause fluctuations in to (yielding in ) which in can yield. As such, precision may be lost due to the noise in the process of recovering in addition to the loss of precision due to the nonlinear function’s naïve approach to estimate .