where $N_c = \frac{1}{}$ and $\eta\ N(0,\delta)$ is the white noise with correlator $<\eta(\tau)\eta(\tau')>=\delta(\tau-\tau')$. Note that it is the span of the master equation in powers of the inverse population size N^{-1} re-scaling with $\tau = 2**t/N$, and $z = x_1 - x_2$ ranges over the interval [-1, 1][20], leading to the solution $\tau_{0*} = \frac{2*\lambda}{1-2*} * cot(\pi)$. Thus, we have the algorithm:

Input: maximum time scale size T, mRNA numbers y, proteins x, maximum steps Steps, tolerance Tol, parameters of the sensing model(coefficients of convertion c1, c2, transcription and translation rate m, n, degradation rate k1, k2, formation coefficients mu1, mu2, diffusion rate b1, b2), dt as each time increasement

Initialize:`maximum time scale`, T, maximum step number steps, tolerance Tol, numbers of mRNA after the first diffusion process xhat if necessary, initialized as one random the number in the first status and thus we start with the largest interval to cover higher possibilities, i.e. [x(0),x(0)+1,...,x[1]-1],

for do

xhat(length(xhat))< x(i+1): **repeat**

record the size T, time t, steps Steps - steps +1

set the sequence according to size T (the interval for mRNA numbers) x(1), x(2),...x(T) and generate the population number of proteins according data distribution, y(1), y(2),...y(T). T initialized as the X(i+1)-X(i).

consider Hamilton Markov (Hierarchical)[19]

if *xhat* exists(iterated from previous status) **then**

: segment the interval into several sub-sequences (X0 as the new current status, X1 as the previous status.)

end if

Note that: As we only consider up streaming, down regulation into those before the previous status is not included.

function dynamics Inputs: x and fitted y (or $x0, y0$ or $x1, y1$)

calculate degradation term $w1$ and $w2$ according to the (*1)

calculate px, py, dx, dy, conversion rate, H_θ and H_x, s, according to Appendix 2

predict multiplied mRNA and protein numbers xhat, yhat, and other Hamiltonians.

calculate updated gamma, delta

calculate tolerance for further stopping criteria as the residule of gamma and cell numbers with: $tol = abs(Gamma - gamma)/gamma; toll = mean(abs(xhat - x1)./x1 + abs(yhat - y1)./y1)$

Output: H, Hhat, HthetaX, HthetaY, HxX, HxY, HamilX, HamilY, HamilXhat, HamilYhat, xhat, yhat, sX, sY, px, py, pxhat, pyhat, actionratio, delta, gamma, Delta, Gamma, cr1, c1, c2, crhat, c1hat, c2hat, tol, toll, Txc, Tyc, Txchat, Tychat

concatenate results:

if X0, X1 exist **then**

xhat = [X0hat, X1hat]

end if;

if only X0 exist **then**

xhat = X0hat.

end if

do similar prediction regenerate the mRNA numbers X according to Y with **function dynamics** again for comparison. Results are with postfix 'L'

store the quantities of 'successful' moves with smaller tolerance and action for either from mRNA or protein numbers.

if satisifies the configuration condition **then**

$Tol = min(tol * toll, tolL * tollL)$

else

Fail++

end if

---

3 results and discussion

0.1 To have a clearer understanding of the switching process combining the binding with increasing and decreasing speed both of *hax*1 and HS_1, the two population are regarded as promoters and resistors both when activating and deactivating each other's production.(The coarse process can be briefly described as in 1, and it is briefly introduced in the previous chapter.)

3.1 Finding three critical points and explore their stability

As we have data (see Appendix C) of 15 status in all both for hax1 and HS_1 with their different cell numbers taken as X and Y in our model. For reward computation for their Markov chain, we pre-compute the their Hamiltonians, Action Potentials, mean switching time and related dynamics in the form (see availability), and the 6 upstreaming status, which is the focus of the experiment application of our model. Using the pre-computation results, we are able to discuss about some practical problems about the current model. There are three groups of quantities studied combining the action potential as well as Hamiltonian inspired by bacterial quorom sensoring, 'momentum and cell numbers', 'MTS with the SDE', and 'corresponding Hamiltonians', of each transform status in Appendix A. First, we use the taylor expansion to simplify the four ODE achieved in Appendix B: to

$$dx = \frac{C1}{1+(\frac{y}{x+y})^m} P_x - \mu1 * x * P_X)$$

$$dy = \frac{C2}{1+(\frac{x}{x+y})^n} P_y - \mu2 * y * P_Y)$$

$$dP_X = \frac{C2*m*(\frac{x}{x+y})^{m-1}}{(1+(\frac{x}{x+y})^m)^2}(P_Y - 1) - \mu1 * P_X - \mu1$$

$$dP_Y = \frac{C1*n*(\frac{y}{x+y})^{n-1}}{(1+(\frac{y}{x+y})^n)^2}(P_X - 1) - \mu2 * P_Y - \mu2$$

Note that our model here simplify the origin model where $c_i = \frac{a_i}{b_i}$, with b_i = 1 as the burst size of protein i, $\frac{x}{(x+y)} = \frac{x}{K_2*(x+y)}$ as $k_2 = 1$ is the dissociation constants standing for gene x binding on y's protein binding site. Regarding x and y as leading order variable, we apply phase analysis to consider the solution's stability around the three zero-energy points, which achieved through setting dx, dy, dP_X and dP_Y all to zero and combine the Hamiltonian's special case when H= 0 (and $H_P = 0$): P1(x, y, $\frac{\mu2*X}{C1}$, $\frac{\mu1*Y}{C2}$, where x and y are the solution of $x = \frac{C1}{\mu1*(1+(\frac{y}{x+y})^n)}$ and $y = \frac{C2}{\mu2*(1+(\frac{x}{x+y})^m)}$, P2(x, y, 0, 0), where x and y are the solution of $x = \frac{C1}{\mu1(1+(\frac{y}{x+y})^n} = -y = -\frac{C2}{\mu1(1+(\frac{x}{x+y})^m}$, and P3(0,0,0,0) As P_X and P_Y are either zero or formula can be replaced by x and y around those three convergence points. We here, consider the analysis on x and y as following: denote $dx = f(x,y)$ and $dy = g(x,y)$, and the we try to find x* and y* satisfy the $f(x,y) = 0$ and $g(x,y) = 0$ as well as holding the zero-energy points for their momentum. Thus with approximation: $dx = f_X(x*,y*)(x - x*) + f_Y(x*,y*)(y - y*)$, and $dy = g_X(x*,y*)(x - x*) + g_Y(x*,y*)(y - y*)$. We have $A = \begin{bmatrix} f_X & f_Y \\ g_X & g_Y \end{bmatrix}$

$$= \begin{bmatrix} -\mu1 * P_X & \frac{C2*n*(\frac{y}{x+y})^{n-1}}{(1+(\frac{y}{x+y})^n)^2} * P_X \\ -\mu2 * P_Y & \frac{C1*m*(\frac{x}{x+y})^{m-1}}{(1+(\frac{x}{x+y})^m)^2} * P_X \end{bmatrix}$$

where there exists the a>0, b>0 for the eigenvalue λ:

$$\lambda^2 + a * \lambda + b = 0 \quad (1)$$

$$a = -(f_X + g_Y)|_{(x*,y*)} \quad (2)$$

$$b = |A| \quad (3)$$

so that point(x*, y*) is the convergence points. Thus, we discuss about the stability of the three points as following: we denote $X = (1 + (\frac{x}{x+y})^m)$ and $Y = (1 + (\frac{y}{x+y})^n)$, compute the a and b as: $a = -(f_X + g_Y)|_{(x*,y*)} = \mu1 * P_X + \frac{C2*m(\frac{x}{x+y})^{m-1}}{(1+(\frac{x}{x+y})^m)^2} * P_Y, b = |A|_{|(x*,y*)} = \mu2 * P_X * P_Y * \frac{C1*n(\frac{y}{x+y})^{n-1}}{(1+(\frac{y}{x+y})^n)^2} - \mu1 * P_X * P_Y * \frac{C2*m(\frac{x}{x+y})^{m-1}}{(1+(\frac{x}{x+y})^m)^2}$

[1] for $P1(x, y, \frac{\mu2*X}{C1}, \frac{\mu1*Y}{C2})$, where x and y are the solution of $x = \frac{C1}{\mu1*(1+(\frac{y}{x+y})^n)}$ and $y = \frac{C2}{\mu2*(1+(\frac{x}{x+y})^m)}$,

$a = \mu1 * \frac{\mu2*x}{C1} + \frac{C2*m*(\frac{x}{x+y})^{m-1}}{(1+(\frac{x}{x+y})^m)^2} * \frac{\mu1*y}{C2} = \frac{\mu2*B^2}{B} + \frac{\mu1*C2*m*(\frac{x}{x+y})^{m-1}}{\mu2*A}$

$= \frac{\mu2^2*B^2 + \mu1*C2*A*m*(\frac{x}{x+y})^{m-1}}{\mu2*A*B}$ As in out model, $\mu1 = \mu2 = 1$ and $m > 0, C2 > 0$, we have $fracxx + y > 0$ and thus $m*(\frac{x}{x+y})^{m-1} > 0, A > 0, nominator = B^2 + C2*A*m*(\frac{x}{x+y})^{m-1} > 0$ while denominator $= A * B > 0$, which gives out $a > 0$,

$b = \mu2 * P_X * P_Y * \frac{C1*n*(\frac{y}{x+y})^{n-1}}{B^2} - \mu1 * P_X * P_Y * \frac{C2*m*(\frac{x}{x+y})^{m-1}}{A^2}$

$= \frac{\mu1*C1*n*\mu2*C2*m*P_X*P_Y}{(A^2*B^2)} * (A^2(\frac{y}{x+y})^{n-1}\mu1 * C2 * m - (B^2(\frac{x}{x+y})^{m-1}\mu2 * C1 * n), = \frac{\mu2*C1*n*A^2*(\frac{y}{x+y})^{n-1} - \mu1*C2*m*B^2*(\frac{x}{x+y})^{(m-1)}}{\mu1*C2*m*\mu2*C1*n}$

nominator = $\frac{dx}{dy} - \frac{dy}{dx}$ while denominator > 0

Thus, P1 is stable if and only if $\frac{dx}{dy} > \frac{dy}{dx}$, here means the production of hax1 is faster than HS_1.

[2] for P2(x, y, 0, 0), where x and y are the solution of $x = \frac{C1}{\mu1(1+(\frac{y}{x+y})^n} = -y = -\frac{C2}{\mu1(1+(\frac{x}{x+y})^m}$,and $P_X = P_Y = 0$