# Construct and Visualize Three Fuzzy Controllers for Biological S-Systems

Shinq-Jen Wu
Department of Electrical Engineering Da-Yeh University,
68 University Rd., Dacun
Changhua 51591, Taiwan, R.O.C

## ABSTRACT
Biological S-systems use power-law-based differential equations to show *net* interactive strength between constitutes. In medium-sized or large-scale systems, dynamic constants of S-systems represent the *net* value of the action strength, rather than the actual strength. As a result, S-systems becomes the most potential model for large-scale systems. Moreover, biological systems are always subject to uncertainty and noise. Fuzzy logic controllers are developed for handing uncertainty, imprecision, and complexity in the real world. Noise, uncertainty, and the interactive information are all implied in fuzzy if-then rules. In this study, the previously proposed *optimal fuzzy controller* is used to smoothly regulate biological systems to target states with *minimum* input consumption. Then, an integrated fuzzy proportional-integral-derivative controller (*integrated fuzzy PID*) and a *pole-placement-based fuzzy controller* for biological S-systems are proposed. Additionally, these fuzzy controlled systems are all visualized in block diagrams to provide biological researchers a friendly environment. For these three kinds of fuzzy controllers, only three control rules are used to control a cascade biological system. Simulation results shows that nearly perfect results are achieved for all these three controllers.

## Keywords
T-S Fuzzy systems, pole-placement, optimal fuzzy control

## 1. INTRODUCTION
Two well-known biological systems (S-systems and Michaelis-Menten kinetic systems) are based on biochemistry system theory and described as highly nonlinear differential equations. The modelling of generalized Michaelis-Menten systems is a bottom-up process. Systems are gradually constructed from small systems [1] to medium-sized systems [2] and then expanded to large systems [3]. Parameter estimation needs a large amount of experimental data and doing experiments repeatedly. The modelling of S-systems is a top-down process and parameters are estimated through computational approaches. Good generalization properties let S-system become the most potential model for large-scale systems. Liu and coworkers used the S-system to describe p53 signaling pathway mechanism [4]. S-system modelling is a multi-objective multi-constraints optimization problem. Various intelligently computational technologies were recently developed to achieve S-system modelling [5-11].

Fuzzy set theory shows great potential in dealing with biological data and modelling *biological systems* because of the use of linguistic variables and fuzzy relationship. Luo and An took a review of fuzzy set theory in device control, biological control, classification and pattern recognition, and prediction and association [12]. Komlyama and coworkers emphasized that fuzzy interactions *always* exist in cell macromolecular nanoarchitectonics [13]. Abyad and coworkers used T-S fuzzy models to describe biomass growth processes and then optimal fuzzy control was used to control the process [14]. Bordon et al. used fuzzy logic to achieve the quantitatively modelling of repressilators with unknown kinetic data [15]. Liu and coworkers introduced fuzzy Petri nets for biological system modelling and discussed the capacities and applications [16]. They further proposed a hybrid of continuous Petri nets and fuzzy inference systems to achieve integrated modelling of biological systems with uncertainties [17]. Zhu and coworkers used fuzzy neural networks as inverse systems to achieve decoupling control of marine biological enzyme fermentation processes [18]. An adaptive T–S type neural-fuzzy scheme was proposed to achieve the fuzzy modeling of multi-inputs multi-outputs *biological systems* (small-scale genetic networks, branch pathways and cascade networks systems) [6]. The number of generated rules depends on the number of input variables of underlying systems and the division of the input space: There are $3^n$ rule numbers for an $n$-dimensional biological system with each input variable being divided into three intervals. To reduce the number of rules, researchers tried to construct *biological* fuzzy systems with a fixed number of rules. However, to determine the number of rules which are sufficient to ensure the accuracy researchers should fully understand the underlying biological systems. This will lose the essence of adaptive T–S type neural-fuzzy modeling.

Linear optimal control (LQR) gives the best possible performance for linear systems. Fuzzy modeling can mimic a real nonlinear system well. Fuzzy control supports more robust control than linear control does. Therefore, the previously proposed optimal fuzzy controller which is based on T-S fuzzy systems and is an integration of LQR and fuzzy control have practical *advantages* [19, 20]. The optimal fuzzy controller has been successfully applied to magnetic suspension system, 4-pole and 8-pole active magnetic bearing system, inverted pendulum system, half-car active suspension system, and even Taiwan *i*TS-1 experimental car.

PID controllers are still extensively used in industry process due to simple structures with only error and error derivative as inputs and model-free advantages. Parameter tuning is an essential issue for the performance of PID controllers. Ziegler-Nichols methods are usually used to estimate the PID parameters [21]. Various computational approaches are proposed to achieve optimally auto-tuning of the parameters [22-27]. Additionally, the performance of PID controllers dramatically reduces as the nonlinearity and complexity of the underlying systems increase. Fuzzy gain scheduling is a general way to improve the quality of PID control of nonlinear systems [28]. Hermassi and coworkers used fuzzy gain scheduling PID controllers to develop vector control strategy of grid-connected wind energy conversion systems [29]. Xhaiyatham and Ngamroo used this technology for the stabilization of superconducting magnetic energy storage of power system [30]. A hybrid of fuzzy logic control and PID controller is also a way to increase the robustness and accuracy of PID-controlled systems. Liu and coworkers used fuzzy PID controller to achieve position control of manipulator working space, wherein particle swarm optimization is used to online self-tuning PID parameters [25]. Phu and coworkers combined fuzzy PID controllers with fuzzy control process which is expressed as linear-form fuzzy differential equation [31]. Cao and coworkers proposed $H_\infty$ fuzzy PID control synthesis for T-S fuzzy systems [32].

Dey and Ayyagari used fuzzy number to express parametric uncertainties and discussed fuzzy pole placement technology for robust PID controller design [33]. Bai and coworkers used fuzzy Lyapunov function method to solve linear matrix inequality pole placement problem [34]. In this study, three control methods for biological systems are proposed. First, the previously proposed optimal fuzzy controller was used for biological systems and the control process was visualized to provide biological researchers a friendly environment. Second, an integrated fuzzy PID control scheme for biological or physical systems is constructed to achieve satisfactory performance. Finally, a pole-placement-based fuzzy control is proposed and examined.

# 2. OPTIMAL FUZZY CONTROL VISULIZATION

For nonlinear physical or biological systems described by T-S fuzzy models [19, 20]: ($R^i$ denotes the $i$th rule of the fuzzy model, $i = 1, \dots r$.)

$R^i$:If $x_1$is $T_{1i}, \dots, x_n$is $T_{ni}$,

then $\dot{X}(t) = A_i(t)X(t) + B_i(t)u(t),\ \ Y(t) = C(t)X(t),$ (1)

where $x_1 \dots, x_n$ are system states, $T_{1i}, \dots T_{ni}$ are fuzzy terms, $u(t)$ and $Y(t) = [\,y_1, \dots y_{n'}]^T$ are, respectively, system input vectors and system output vectors. The desired rule-based fuzzy controller is in the form of

$R^i$:If $y_1$is $S_{1i}, \dots, y_{n'}$is $S_{n'i}$, then $u(t) = r_i(t),\ \ i = 1, \dots \delta,$ (2)

where $S_{1i}, \dots S_{ni}$ are the input fuzzy terms of the $i$th control rule. The asymptotically local optimal control law is [19, 20]

$$r_i^*(t) = -B_i^T \bar{\pi}_i X^*(t), \quad (3)$$

and their *blending* global minimizer, $u^*(t) = \sum_{i=1}^{r} h_i\left(X^*(t)\right) r_i^*(t)$, is able to minimize

$$J\left(u(\cdot)\right) = \int_0^\infty (X^T(t)LX(t) + u^T(t)Qu(t))dt, \quad (4)$$

where $X^T(t)LX(t)$ is state trajectory penalties and $u^T(t)Qu(t)$ is fuel consumption with $L$ and $Q$ are symmetric positive semidefinite matrices. Based on the essence of the dynamic programming, the quadratic optimization problem becomes a successively ongoing dynamic programming regarding the state resulting from the previously decision [20]. The fuzzy subsystem in Eq. (1) and the fuzzy control rule in Eq. (2) have a one-to-one correspondence ($i$th-rule to $i$th-rule), and at any time instant the overall behavior of the fuzzy system is a fuzzily blending result of all fuzzy subsystems [20]. The proposed optimal fuzzy controller has been successfully applied to magnetic suspension system, 4-pole and 8-pole active magnetic bearing system, *inverted* pendulum system, half-car active suspension system, and even Taiwan *i*TS-1 experimental car.

Now, a cascade pathway in Fig. 1 is used to examine the performance of the optimal fuzzy controller in regulating the biological system *back* to desired states. The entire controlling process is visualized in the Simulink environment for biological researchers to get the points easily and then to apply to other biological systems.

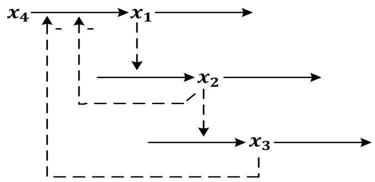


**Fig 1: Cascade pathways [35]**

The cascade pathway possesses an independent variable $x_4$ to produce mid-product $x_1$, and then to generate mid-product $x_2$ which induces the generation of the final product $x_3$. The generation reaction from the source $x_4$ to $x_1$ is inhibited by both mid-product $x_2$ and $x_3$. The dynamic behavior is described as the following S-system in Eq. (5).

$$\begin{aligned} \dot{x}_1 &= 10x_2^{-0.1}x_3^{-0.05}x_4 - 5x_1^{0.5}, \\ \dot{x}_2 &= 2x_1^{0.5} - 1.44x_2^{0.5}, \\ \dot{x}_3 &= 3x_2^{0.5} - 7.2x^{0.5}. \end{aligned} \quad (5)$$

All the reaction constants (exponent order and rate constants) are cited from Tsai and Wang's paper [35].

- ***T-S fuzzy modelling of S-systems***

Optimal fuzzy controllers show fuzzing blending behavior of the locally optimal control laws in Eq. (3) which are based on T-S fuzzy systems,

$$\begin{aligned} \dot{x}_i &= v_i^+ - v_i^- \\ &= \alpha_i \prod_{j=1}^{n+m} x_j^{g_{ij}} - \beta_i \prod_{j=1}^{n+m} x_j^{h_{ij}}, i = 1, \dots n \end{aligned} \quad (6)$$

where $g_{ij}$ and $h_{ij}$ denote the *net interactive strength* from $x_j$ on $x_i, \alpha$ and $\beta_i$ are the rate constants. The $x_i, i = 1, \dots n$ are dependent variables and $x_{n+1}, \dots x_{n+m}$ are independent variables, the values of which remains constant during a period of an experiment. To construct the corresponding optimal fuzzy controller of the cascade pathway biological system, the S-systems in Eq. (5) must transform to their corresponding T-S fuzzy systems. A neural fuzzy scheme was proposed to construct T-S fuzzy systems of biological systems [6]. A fuzzy prototype was further proposed to describe cell growth signaling mechanisms, wherein four nominal cases were concerned: resting tissue cells, resting blood vessel cells, tissue cells subjected to additional stimuli and cells in blood vessel [36]. Here, a region nonlinear transformation is proposed to get the corresponding T-S fuzzy model for achieving optimal control purpose.

The T-S fuzzy system can be realized as a nonlinear system with several linear subsystems which describing local features (regions) of the underlying system, and the local feature can be a certain organizational environment or experimental conditions in [37]. Equation (7) is a locally linearized system of the S-system [38]:

$$\dot{z}(t) = \left(A_D \circ E\right) z(t) + \left(A_I \circ F\right) u(t)$$
$$= \mathring{A} z(t) + \mathcal{B} u(t). \qquad (7)$$

whrein $\circ$ denotes Hadamard product, the perturbed dependent variables $z_i \triangleq x_i - \bar{x}_i, i = 1, \ldots n$, the perturbed independent variables $u_l \triangleq r_l - \bar{r}_l, l = 1, \ldots m$, and the system parameters $A_D = [g_{ij} - h_{ij}]_{j=1,\ldots m}^{i=1,\ldots n}$, $A_I = [g_{ij} - h_{ij}]_{l=1,\ldots m}^{i=1,\ldots n}$, $E = [\bar{v}_i^+ / \bar{x}_j]_{j=1,\ldots n}^{i=1,\ldots n}$ and $F = [\bar{v}_i^- / \bar{r}_l]_{l=1,\ldots m}^{i=1,\ldots n}$. The upper bound notation denotes variables in equilibrium. The $E, F$ relate to an operating point or an equilibrium point for linearization. Different set of independent variables (variable experimental conditions) generate different values of $E$ and $F$, and then the corresponding local dynamic behavior is also different. For the cascade pathway in Eq. (5), the system matrix $A_D = \begin{bmatrix} -0.5 & -0.1 & -0.05 \\ 0.5 & 0.5 & 0 \\ 0 & 0.5 & -0.5 \end{bmatrix}$ and $A_I = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$. Additionally, the steady state values (equilibrium points) are closely related to independent variables [38, 39]:

$$\bar{Y}_d = A_D^{\dagger} \cdot b - (A_D^{\dagger} A_I) \cdot Y_I, \quad (8)$$

where $A_D^{\dagger}$ is the inverse or the pseudoinverse of $A_D$, $y_i = \ln x_i$, $\bar{Y}_d = [\bar{y}_j]_{j=1\ldots n}$ and $Y_I = [y_j]_{j=n+1\ldots n+m}$, $\alpha\beta_i = \ln(\frac{\beta_i}{\alpha_i})$ and letting matrices $b = [\alpha\beta_i]_{i=1\ldots n}$. The $\bar{Y}_d$ denotes $Y_d$ in equilibrium. The independent variables are used as the inputs of T-S fuzzy system, and the number of fuzzy rules depends on the division number of the input space. A rule represents a subsystem, describing the locally dynamic behavior:

$\boldsymbol{R^l}$: IF $u_1$ is $T_{1l}$, $u_2$ is $T_{2l}$, …, $u_m$ is $T_{ml}$

then $\dot{z}(t) = \mathring{A}_l z(t) + \mathcal{B}_l u(t)$, (9)

where $\mathring{A}_l$ and $\mathcal{B}_l$ is the $\mathring{A}$ and $\mathcal{B}$ in Eq. (7), which depend on the values of independent variables. Figure 2 is the schematic diagram of input space segmentation and region linearization for constructing T-S fuzzy systems.

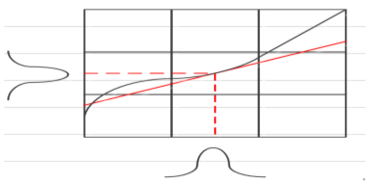

**Fig 2: Schematic diagram of region linearization**

Three experimental conditions are chosen as operating points: $x_4 = 0.1$ ( rule $R^1$ ), $x_4 = 0.75$ (rule $R^2$ ) and $x_4 = 3$ ( rule $R^3$). Based on Eq. (8), the corresponding steady state values are obtained through the following Matlab function:

```
function x_ss=Sstate(u,A_d,A_I,b)
   Yi=log(u);  %u=x4
   Yd=inv(A_d)*(b'-A_I*Yi);
   x_ss=exp(Yd);
end
```

The x_ss, A_d, A_I denote, respectively, the steady state $\bar{X}$, the system matrices $A_D$ and $A_I$. For the three experimental conditions the estimated steady states are $\bar{X} = [\bar{x}_1, \bar{x}_2, \bar{x}_3]$ =[0.0827, 0.1595, 0.0277] for $x_4 = 0.1$, $\bar{X}$= [1.8347, 3.5391, 0.6144] for $x_4 = 0.75$ and $\bar{X}$= [15.4812, 29.8633, 5.1846] for $x_4 = 3$. Here, the fuzzy system is

$\boldsymbol{R^1}$: If $x_4$ is low, then $\dot{z}(t) = \mathring{A}_1 z(t) + \mathcal{B}_1 u(t)$,
$\boldsymbol{R^2}$: If $x_4$ is medium, then $\dot{z}(t) = \mathring{A}_2 z(t) + \mathcal{B}_2 u(t)$,
$\boldsymbol{R^3}$: If $x_4$ is high, then $\dot{z}(t) = \mathring{A}_3 z(t) + \mathcal{B}_3 u(t)$, (10)

wherein

$$\mathring{A}_1 = \begin{bmatrix} -8.6953 & -0.9015 & -2.5964 \\ 3.4781 & -8.031 & 0 \\ 0 & 3.7564 & -21.6367 \end{bmatrix},$$

$$\mathring{A}_2 = \begin{bmatrix} -1.8475 & -0.1914 & -0.5511 \\ 0.7383 & -0.3827 & 0 \\ 0 & 0.7973 & -4.5927 \end{bmatrix},$$

$$\mathring{A}_3 = \begin{bmatrix} -0.6354 & -0.0659 & -0.1897 \\ 0.2542 & -0.1318 & 0 \\ 0 & 0.2745 & -1.5810 \end{bmatrix},$$

$$\mathcal{B}_1 = \begin{bmatrix} 14.3756 \\ 0 \\ 0 \end{bmatrix}, \quad \mathcal{B}_2 = \begin{bmatrix} 9.0300 \\ 0 \\ 0 \end{bmatrix},$$

$$\mathcal{B}_3 = \begin{bmatrix} 6.5577 \\ 0 \\ 0 \end{bmatrix},$$

where $\mathring{A}_1, \mathcal{B}_1$ for subsystem 1 (rule $R^1$), $\mathring{A}_2, \mathcal{B}_2$ for subsystem 2 (rule $R^2$), and $\mathring{A}_3, \mathcal{B}_3$ for subsystem 3 (rule $R^3$).

- ***Optimal fuzzy controller***

The fuzzy control law in Eq. (3) for the $i$th fuzzy subsystem is further expressed as a feedback gain $K_i$;

$$r_i^*(t) = -B_i^T \bar{\pi}_i X^*(t) = -K_i X^*(t), \quad (11)$$

and the optimal fuzzy controller for the entire system becomes

$$u^*(t) = -\sum_{i=1}^{r} h_i\left(X^*(t)\right) K_i X^*(t). \quad (12)$$

By choosing the penalty matrix of $J\left(u(\cdot)\right)$ in Eq. (4) as $L = \begin{bmatrix} 1000 & 0 & 0 \\ 0 & 1000 & 0 \\ 0 & 0 & 1000 \end{bmatrix}$ and $Q = 1$ and using the ***lqr*** function of the Matlab,

$$K_i = \boldsymbol{lqr}(A_i, B_i, L, Q), \quad (13)$$

the feedback gain $K_i = [k1, k2, k3]$ is then estimated:

$K_i = [31.1721\ 19.4427, 0.1862]$ for $x_4 = 0.1$,
$K_i = [1.8347\ 3.5391\ 0.6144]$ for $x_4 = 0.75$,
$K_i = [15.4812\ 29.8633\ 5.1846]$ for $x_4 = 3$. (14)

Figure 3 is the visualization graph for the optimal fuzzy controlled system, wherein the fuzzy controller (denoted by the blue dash line) and the S-system (denoted by the purple dash line) are shown in distinct block diagrams. The $x_2$ inhibits the generation of both $x_1$ and $x_3$. So, we choose $dx = x_2 - \bar{x}_2$ as the concentration deviation to estimate the feedback law $du = K \cdot dx$, where $K$ is the output of fuzzy controller. Only three simple fuzzy rules are used:

$\boldsymbol{R^1}$: If $x_4$ is low, then $K_1 = [31.1721\ 19.4427\ 0.1860]$,
$\boldsymbol{R^2}$: If $x_4$ is medium, then $K_2 = [31.4695\ 19.5353\ 0.3073]$,
$\boldsymbol{R^3}$: If $x_4$ is high, then $K_3 = [31.5500\ 19.5599\ 0.3415]$. (15)

Dry-lab experiments are conducted at $x_4 = 1$ and the initial condition $X(0) = [0.2\ 0.5\ 0.1]$. Figure 3 shows the block diagram for the controlling process of the cascade system, wherein $du = -\sum_{i=1}^{r} h_i\left(X^*(t)\right) K_i \cdot dX$ because that the controller is based on perturbed variable $dX$. The simulation results (denoted by the red solid line) are shown in the right-down corner of Fig. 3. The estimated states perfectly meet the desired states. Even the independent variables are outside of the range of [0 3], for example, in the case of $x_4 = 3.5$ the estimated states $X = [19.624543977938\ \ 37.85588479856\ \ 6.5721819724842]$ still well match the desired values $X_d = [19.624487682916\ \ 37.855879017971\ \ 6.5722012183977]$.

.

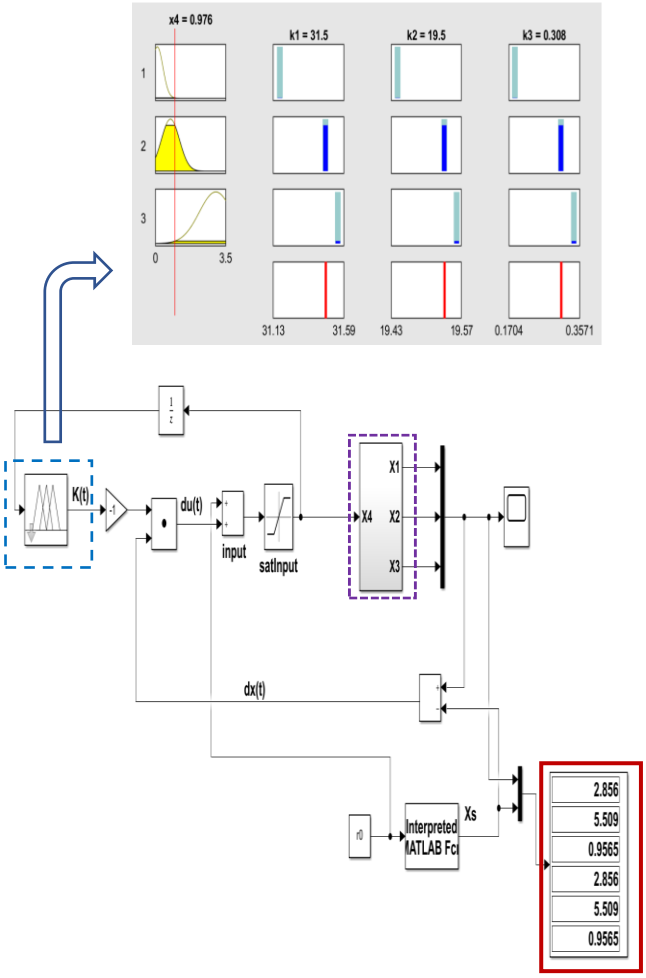


**Fig 3: Optimal fuzzy controlled systems**

## 3. INTEGRATED FUZZY PID

PID controllers, one kind of feedback control mechanisms, has been widely used in industry. PID controllers cannot guarantee system stability or optimal control. However, only the response of the measured system variable is needed. PID controllers are still broadly applicable because the controllers execute model-free control. There are two issues for using PID controllers. One is the parameter tunning, which is conceptually intuitive, but is hard in practice. Unsuitable parameters cannot stabilize the underlying systems. The other is the performance of PID controller is degraded when the underlying system is nonlinear or asymmetric. Various hybrid of PID and fuzzy controllers were proposed [22-32]. In this study, an integrated fuzzy PID is proposed, wherein each linear subsystem is stabilized by the corresponding PID control law and the fuzzily blending controller of these individual PID controller law can stabilize the underlying nonlinear system.

■ ***Local behavior (fuzzy subsystems)***

We still chose $x_4 = 0.1, 0.75$ and 3 for three basic fuzzy (linear) subsystems in Eq. (7). The three subsystems are denoted as low, medium and high, respectively. The systems matrix $\mathring{A}_l, \mathcal{B}_l, l = 1,2,3$ are in Eq. (10). Their PID parameters are automatically tuned to ensure the stability of locally closed-loop subsystems through Control System Toolbox™ in Simulink/Matlab software. Simulation results for the three subsystems are examined through Fig. 4. Table 1 shows the tuned PID parameters and the simulation results for the three subsystems.

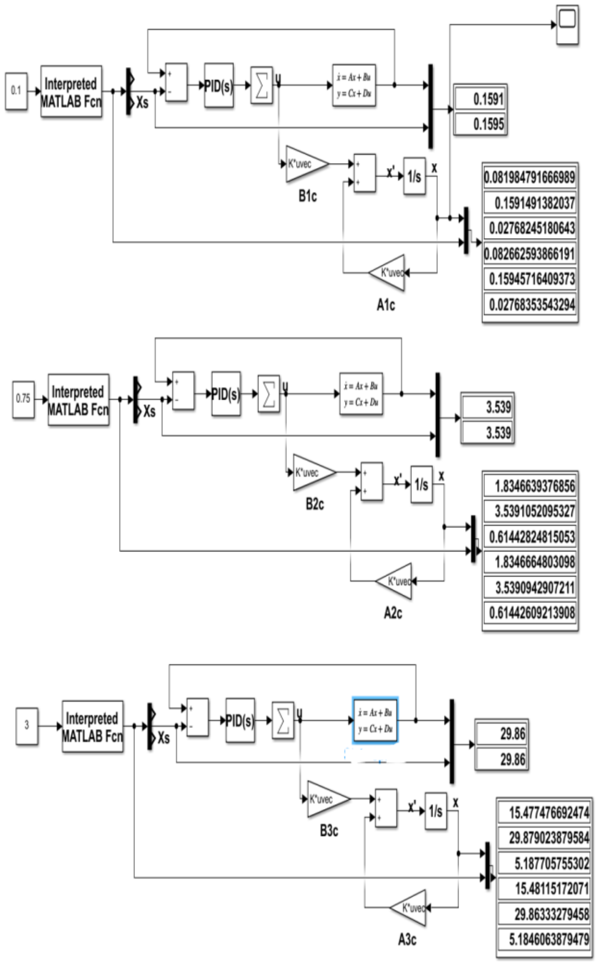


**Fig4: Local behavior for PID controllers**

**Table 1: PID parameters and simulation results for the three subsystems.**

| $x_4$ | $\begin{bmatrix} K_p \\ K_I \\ K_D \end{bmatrix}$ | state variables | |
|---|---|---|---|
| | | estimated value | target values |
| 0.1 | $\begin{bmatrix} -3.1234 \\ -1.2874 \\ -1.3919 \end{bmatrix}$ | $\begin{bmatrix} 0.08271989 \\ 0.15948652 \\ 0.02768342 \end{bmatrix}$ | $\begin{bmatrix} 0.08266259 \\ 0.15945716 \\ 0.02768354 \end{bmatrix}$ |
| 0.75 | $\begin{bmatrix} -3.4356 \\ -1.7854 \\ -1.3341 \end{bmatrix}$ | $\begin{bmatrix} 1.83466394 \\ 3.53910521 \\ 0.61442825 \end{bmatrix}$ | $\begin{bmatrix} 1.83466648 \\ 3.53909429 \\ 0.61442609 \end{bmatrix}$ |
| 3 | $\begin{bmatrix} -2.0328 \\ -0.6402 \\ -1.4185 \end{bmatrix}$ | $\begin{bmatrix} 15.47747669 \\ 29.87902388 \\ 5.18770576 \end{bmatrix}$ | $\begin{bmatrix} 15.48115172 \\ 29.86333279 \\ 5.18460639 \end{bmatrix}$ |

■ ***Global behavior***

The outputs of PID controllers of the three fuzzy subsystems are fuzzily blended as follows.

$$u^*(t) = \sum_{i=1}^{r} h_i\left(X^*(t)\right) u_{pid}^{i} \qquad (16)$$

The fuzzy-logic-control icon in the fuzzy toolbox of Simulink cannot directly used for PID control. Therefore, the membership-function icon is used to create the firing strength of each rule $w_i(X^*(t)), i = 1, \dots 3$, and then, the normalized firing strength$h_i(X^*(t))$ is obtained. The detailed scheme for constructing fuzzy PID is shown in Fig. 5. Table 2 shows the simulation results of the proposed integrated fuzzy PID controllers compared to the target values. Nearly perfect match is obtained, even the independent variable $x_4$ are far out of the range of the chosen subsystem conditions, [0.1 3].

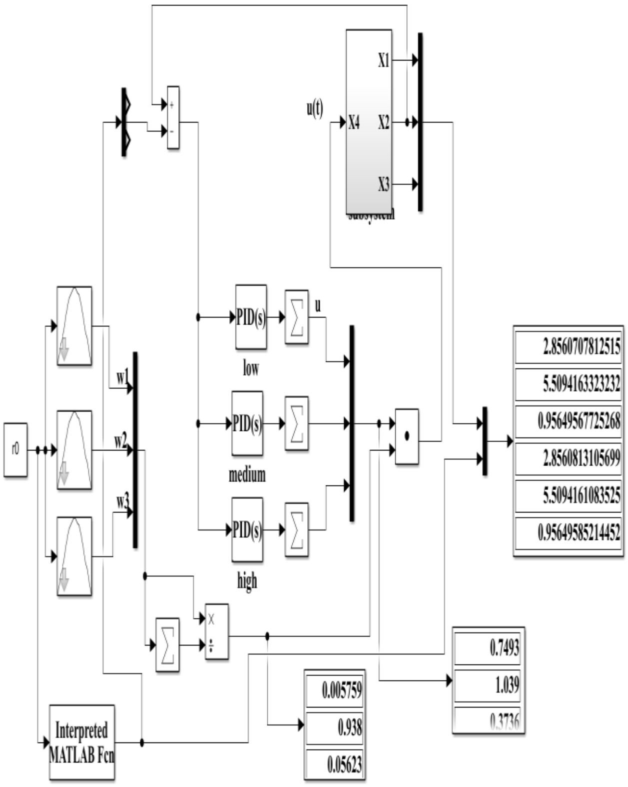


**Fig5: Integrated PID controllers scheme.**

**Table 2: Simulation results for the integrated fuzzy PID controlled system.**

| $x_4$ | state variables | |
|---|---|---|
| | **estimated value** | **target values** |
| **0.3** | $\begin{bmatrix} 0.44806572996855 \\ 0.86432432478494 \\ 0.15005630638627 \end{bmatrix}$ | $\begin{bmatrix} 0.44806572996852 \\ 0.86432432478495 \\ 0.15005630638628 \end{bmatrix}$ |
| **0.5** | $\begin{bmatrix} 0.98321273950862 \\ 1.8966295129408 \\ 0.32927595710777 \end{bmatrix}$ | $\begin{bmatrix} 0.98321273950852 \\ 1.8966295129408 \\ 0.32927595710778 \end{bmatrix}$ |
| **1** | $\begin{bmatrix} 2.8560813105706 \\ 5.5094161083524 \\ 0.9564958521445 \end{bmatrix}$ | $\begin{bmatrix} 2.8560813105699 \\ 5.5094161083525 \\ 0.95649585214452 \end{bmatrix}$ |
| **1.5** | $\begin{bmatrix} 5.3294230586996 \\ 10.280522875574 \\ 1.7848129992315 \end{bmatrix}$ | $\begin{bmatrix} 5.3294230586976 \\ 10.680522875574 \\ 1.7848129992316 \end{bmatrix}$ |
| **2** | $\begin{bmatrix} 8.2964755487825 \\ 16.004003759232 \\ 2.7784728748665 \end{bmatrix}$ | $\begin{bmatrix} 8.2964755487854 \\ 16.004003759231 \\ 2.7784728748665 \end{bmatrix}$ |
| **2.5** | $\begin{bmatrix} 11.69461597947 \\ 22.559058602369 \\ 3.9165032295778 \end{bmatrix}$ | $\begin{bmatrix} 11.694615979468 \\ 22.559058602369 \\ 3.916503229578 \end{bmatrix}$ |
| **3.5** | $\begin{bmatrix} 19.624487685415 \\ 37.855879016484 \\ 6.5722012179985 \end{bmatrix}$ | $\begin{bmatrix} 19.624487682916 \\ 37.855879017971 \\ 6.5722012183977 \end{bmatrix}$ |
| **5** | $\begin{bmatrix} 33.97105511002 \\ 65.530584641333 \\ 11.37683760975 \end{bmatrix}$ | $\begin{bmatrix} 33.971055084119 \\ 65.530584653007 \\ 11.376837613369 \end{bmatrix}$ |
| **10** | $\begin{bmatrix} 98.68075398912 \\ 190.35639375619 \\ 33.047551148158 \end{bmatrix}$ | $\begin{bmatrix} 98.680673700986 \\ 190.35623784912 \\ 33.047957959917 \end{bmatrix}$ |

# 4. POLE-PLACEMENT-BASED FUZZY CONTROL

Pole placement design is to select an optimal feedback gain matrix to place the closed-loop poles of the underlying system in pre-determined locations such that a satisfactory time and frequency domain behavior can be obtained. Given a linear subsystem $\dot{z}(t) = \mathring{A}_l z(t) + \mathcal{B}_l u(t)$ in Eq. (9) and a vector $V_l$ to denote desired closed-loop pole locations, the Matlab software provide a ***place*** function to estimate a gain matrix $K_l = [k_1, k_2, k_3]$ such that the local state feedback $r_l^*(t) = -K_l z(t)$ locates the locally closed-loop poles at the position $V_l$.

$$K_l = \boldsymbol{place}(\mathring{A}_l, \mathcal{B}_l, V_l). \qquad (17)$$

Three pole-placement-based laws are individually constructed to satisfy local dynamic behavior, and then their fuzzily blending controller is able to ensure that the global behavior of the entire cascade system possesses satisfactory dynamic behavior.

■ ***Local behavior (fuzzy subsystems)***

The same experimental conditions are used for constructing fuzzy rules: $x_4 = 0.1$ ( rule $R^1$ ), $x_4 = 0.75$ (rule $R^2$ ) and $x_4 = 3$ ( rule $R^3$ ). The corresponding desired closed-loop poles for the three subsystems are $V_1 = [-2 \pm 3j, -10]$ for subsystem one (rule $R^1$ ), $V_2 = [-1.5 \pm 0.5, -4.5]$ for subsystem two (rule $R^2$ ) and $V_3 = [-1.5 \pm 0.5, -4.5]$ for subsystem three (rule $R^3$). The principle of setting the position of the desired closed-loop poles is to let the conjugate complex poles (called dominate poles) dominate the dynamic behavior of the underlying system, and the other pole far away from the dominate poles. From an engineering viewpoint, the distance is 3 to 5 times farther.

For convenience, the one-to-one correspondence of fuzzy systems to fuzzy controllers is directly used. Therefore, for fuzzy systems in Eq. (9) the corresponding fuzzy rules are

$\boldsymbol{R^l}$: IF $u_1$ is $T_{1l}$, $u_2$ is $T_{2l}$, …, $u_m$ is $T_{ml}$

then $r_l^*(t) = -K_l X^*(t), l = 1,2,3.$ (18)

By choosing the same experimental conditions $x_4 = 0.1$, 0.75, 3 to, respectively, denote low, medium and high, and using the ***place*** function in Eq. (17) to get the feedback gain: $K_1 = [-1.2615\ 4.6424 - 24.6290]$ for $x_4 = 0.1$, $K_2 = [0.0752\ 0.2310 - 0.2322]$ for $x_4 = 0.75$ and $K_3 = [15.4812\ 29.8633\ 5.1846]$ for $x_4 = 3$. The fuzzy control rule is designed as follows.

$\boldsymbol{R^1}$: If $x_4$ is low, then $k1, k2, k3$ are all low.

$\boldsymbol{R^2}$: If $x_4$ is medium, then $k1, k2, k3$ are all medium.

$\boldsymbol{R^3}$: If $x_4$ is high, then $k1, k2, k3$ are all high. (19)

Figure 6 is the scheme for pole-placement-based fuzzy controlled system. Table 3 is the simulation results and the target values. Perfect match is obtained. However, the control failed when $x_4$ out of the range [0,3].

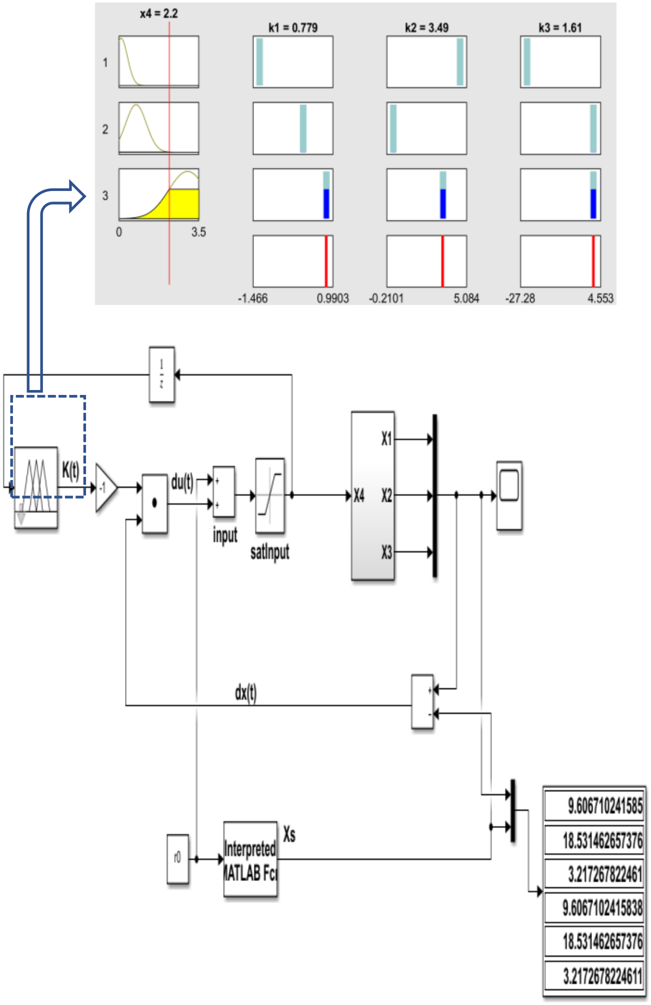


**Fig 6: Pole-placement-based fuzzy controlled systems.**

**Table 3: Simulation results of the pole-placement-based fuzzy controlled systems.**

| $x_4$ | state variables | |
|---|---|---|
| | estimated value | target values |
| 0.3 | $\begin{bmatrix}0.44806572996858\\0.86432432478484\\0.15005630638626\end{bmatrix}$ | $\begin{bmatrix}0.44806572996852\\0.86432432478495\\0.15005630638628\end{bmatrix}$ |
| 0.5 | $\begin{bmatrix}0.98321273745595\\1.8966295136645\\0.32927595728388\end{bmatrix}$ | $\begin{bmatrix}0.98321273950852\\1.8966295129408\\0.32927595710778\end{bmatrix}$ |
| 1 | $\begin{bmatrix}2.8560813105522\\5.5094161083643\\0.95649585214671\end{bmatrix}$ | $\begin{bmatrix}2.8560813105699\\5.5094161083525\\0.95649585214452\end{bmatrix}$ |
| 1.5 | $\begin{bmatrix}5.329423058697\\10.280522875574\\1.7848129992317\end{bmatrix}$ | $\begin{bmatrix}5.3294230586976\\10.280522875574\\1.7848129992316\end{bmatrix}$ |
| 2 | $\begin{bmatrix}8.2964755487825\\16.004003759232\\2.7784728748665\end{bmatrix}$ | $\begin{bmatrix}8.2964755487854\\16.004003759231\\2.7784728748665\end{bmatrix}$ |
| 2.2 | $\begin{bmatrix}9.606710241585\\10.531462657376\\3.217267822461\end{bmatrix}$ | $\begin{bmatrix}9.6067102415838\\18.531462657376\\3.2172678224611\end{bmatrix}$ |
| 2.5 | $\begin{bmatrix}11.69461597947\\22.559058602369\\3.9165032295778\end{bmatrix}$ | $\begin{bmatrix}11.694615979468\\22.559058602369\\3.916503229578\end{bmatrix}$ |

## 5. CONCLUSION

Noise and uncertainty always exist in biological systems. So, fuzzy logic control is a good choice for biological systems. A general fuzzy logic control scheme was previously proposed to regulate S-type biological systems (S-systems) [40]. In this study, three well-known control technologies (linear quadratic optimization, PID and pole-placement design) were integrated with fuzzy logic control to control biological S-systems. The entire controlling process is visualized in the Simulink environment such that biological researchers can get the point easily and modify the scheme for other biological systems. The constructed T-S fuzzy systems are based on three experimental conditions, $x_4 = 0.1, 0.75, 3$, which denotes three independent fuzzy rules or fuzzy subsystems. Simulation shows that nearly perfect results are achieved all these three controllers. However, the *testing* experimental conditions can be far out of the range of [0.1 3] for the integrated fuzzy PID control, is out of the range for optimal fuzzy control, and should in the range for pole-placement-based fuzzy control. The first reason is that the former uses current error and error derivative as the inputs of the controller, but both optimal fuzzy control and pole-placement-based control cannot use the *currently* estimated system inputs as the input of the controllers. (Simulink does not allow this kind of feedback connection.) A delay to denote the *last* time-instant input variable is used for the control input, which will reduce the performance. The second reason is that pole placement method strongly depends on the desired pole position vector $V_i$. In this study, the same performance parameters $L$ and $Q$ are used for optimal fuzzy control, but the $V_i, i = 1, 2, 3$ are set at different positions in order to achieve satisfactory local dynamic behavior. Therefore, the testing range is restricted. In the future, we shall try to compensate these two reasons. Additionally, dependent variables (state variables) of biological systems are always unmeasurable. The proposed controllers are all based on full-state feedback signals. Therefore, a soft observer or estimator is necessary to implement the proposed controllers to real biological systems. We shall develop fuzzy stochastic estimators for dealing with this issue.

## 6. ACKNOWLEDGMENTS

This research was supported by grant number MOST 109-2221-E-212-003 - from the Ministry of Science and Technology of Taiwan, R.O.C.

## 7. REFERENCES

[1] Tyson, J. J. 1991. Modeling the cell division cycle: cdc2 and cyclin interactions. Proc. Natl. Acad. Sci. USA 88:7328-7332.

[2] Tyson, J. J., Chen, K. C., and Novak, B. 2003. Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. Curr. Opin. Cell Biol. 15:221-231.

[3] Tavassoly, I., Parmar, J., Shajahan-Haq, A,, Clarke, R., Baumann, W., and Tyson, J. J. 2015. Dynamic modeling of the interaction between autophagy and apoptosis in mammalian cells. CPT Pharmacometrics Syst Pharmacol 4(4): 263-72.

[4] Liu, P. K., and Wang, F. S. 2010. Inverse problems of biological systems using multi-objective optimization. Chin. J. Chem. Eng. 39(5):399-406.

[5] Wu, S. J., Wu, C. T., and Chang, J. Y. 2012. Fuzzy-based self-interactive multi-objective evolution optimization for reverse engineering of biological networks. IEEE Trans Fuzzy Syst 20(5):865-882.

[6] Wu, S. J., Wu, C. T., and Chang, J. Y. 2013. Adaptive neural-based fuzzy modeling for biological systems. Math Biosci 242(2): 153-60.

[7] Wu, S. J., and Wu, C. T. 2013. Computational optimization for S-type biological systems: cockroach genetic algorithm. Math Biosci 245(2):299-313.

[8] Wu, S. J, and Wu, C. T. 2014. Seeding-inspired chemotaxis genetic algorithm for the inference of biological systems. Comput Biol Chem 53(2):292-307.

[9] Wu, S. J, and Wu, C. T. 2015. A bio-inspired optimization for inferring interactive networks: cockroach swarm evolution. Expert Syst Appl 42(6): 3253-3267.

[10] Wu, S. J, and Wu, C. T. 2018. Smarten up computational intelligence to decipher time series data. Appl Soft Comput 72:442-456.

[11] Liu, S., Tao, C., Huang, Z, and Huang S. 2010. Modeling of p53 signaling pathway based on S-system equations. Journal of Biomedical Engineering 27(3):505-10 (Chinese)

[12] Luo, Z. P., An, K. N. 2001. Fuzzy systems in biomedical science. Int. J. Gen. Syst. 30(2):209–217.

[13] Komiyama, M., Yoshimoto, K., Sisido, M., Ariga, K. 2017. Chemistry can make strict and fuzzy controls for bio-Systems: DNA nanoarchitectonics and cell-macromolecular nanoarchitectonics. Bull. Chem. Soc. Jpn. 90(9):967–1004.

[14] Abyad, M., Karama, A., and Khallouq, A. 2017. Modelling and control of a biological process using the fuzzy logic Takagi-Sugeno. In Proceeding of the 2017 International Renewable and Sustainable Energy Conference (IRSEC).

[15] Bordon, J., Moskon, M., Zimic, N., and Mraz, M. 2015. Fuzzy logic as a computational tool for quantitative modelling of biological systems with uncertain kinetic data. IEEE/ACM Trans Comput Biol Bioinform 12(5):1199–1205.

[16] Liu, F., Heiner, M., and Gilbert, D. 2020. Fuzzy Petri nets for modelling of uncertain biological systems. Brief. Bioinformatics 21(1):198–210.

[17] Liu, F., Sun, W., Heiner, H., and Gilbert, G. 2021. Hybrid modelling of biological systems using fuzzy continuous Petri nets. Brief. Bioinformatics 22(1):438–450.

[18] Zhu, X.L., Jiang, Z.Y., Wang, B., and He Y.J. 2018. Decoupling control based on fuzzy neural-network inverse system in marine biological enzyme fermentation process. IEEE Access, 6:36168–36175.

[19] Wu, S. J. and Lin, C. T. 2000. Optimal fuzzy controller design: local concept approach. IEEE Trans. Fuzzy Systems 8(2):171-185.

[20] Wu, S. J. 2008. Reply to "Further comment on "optimal fuzzy controller design: local concept approach"". IEEE Trans. Fuzzy Systems 16(2):547-549.

[21] Åström, K.J., Hägglund, T. 1995. PID controllers: theory, design and tuning. Instrument Society of America, North Carolina.

[22] Zhuang, M. and Atherton, D.P. 1993. Automatic tuning of optimum PID controllers. In proceeding of the IEE D.

[23] Mudi, R.K. and Pal, N.R. 1999. A robust self-tuning scheme for PI- and PD-type fuzzy controllers. IEEE Trans. Fuzzy Syst. 7(1):2–16.

[24] Li, T.H.S. and Shieh, M.Y. 2000. Design of a GA-based fuzzy PID controller for nonminimum phase systems. Fuzzy Sets Syst. 111(2):183–197.

[25] Liu, Y., Jiang, D., Yun, J., Sun, Y., Li, C., Jiang, G., Kong, J., Tao, B. and Fang, Z. 2022. Self-tuning control of manipulator positioning based on Fuzzy PID and PSO algorithm. Front. Bioeng. Biotechnol. 9:817723.

[26] Chao, C. T., Sutarna, N., Chiou, J. S. and Wang, C. J. 2019. An optimal fuzzy PID controller design based on conventional PID control and nonlinear factors. Appl. Sci. 9:1224.

[27] Elsrogy, W. M., Fkirin, M. A. and Hassan, M. A. M. 2013. Speed control of DC motor using PID controller based on artificial intelligence techniques. In Proceedings of the 2013 CoDIT.

[28] Carvajal, J., Chen, G. and Ogmen, H. 2000. Fuzzy PID controller: design, performance evaluation, and stability analysis. Inf. Sci. 123(3-4):249-270.

[29] Hermassi, M., Krim, S., Kraiem, Y., Hajjaji, M.A., Alshammari, B.M., Alsaif, H., Alshammari, A.S. and Guesmi, T. 2023. Design of vector control strategies based on fuzzy gain scheduling PID controllers for a grid-connected wind energy conversion system: hardware FPGA-in-the-loop verification. Electronics 12:1419.

[30] Chaiyatham, T. and Ngamroo, I. 2013. Optimal fuzzy gain scheduling of PID controller of superconducting magnetic energy storage for power system stabilization. Int. J. Innov. Comput. Inf. Control. 9:651–666.

[31] Phu, N.D., Hung, N.N., Ahmadian, A. and Senu, N. 2020. A new fuzzy PID control system based on fuzzy PID Controller and fuzzy control process. Int. J. Fuzzy Syst. 22:2163–2187.

[32] Cao, K., Gao, X., Lam, H. K. and Vasilakos, A. 2016. $H_\infty$ fuzzy PID control synthesis for Takagi-Sugeno fuzzy systems. IET Control. Theory Appl. 10(6):607-616.

[33] Dey, A. and Ayyagari R. 2016. Robust PID controller design using fuzzy pole placement techniques. IFAC 49(1): 789–794.

[34] Bai, J., Lu, R., Liu, X., Xue, A. and Shi, Z. 2015. Fuzzy regional pole placement based on fuzzy Lyapunov functions. Neurocomputing 167:467-473.

[35] Tsai, K. Y. and Wang, F. S. 2005. Evolutionary optimization with data collocation for reverse engineering of biological networks. Bioinformatics 21(7):1180-8.

[36] Wu, S. J., Chen, W. Y., Chou, C.H. and Wu, C. T. 2013. Prototype of integrated pseudo-dynamic crosstalk

network for cancer molecular mechanism. Math. Biosci. 243(1): 81-98.

[37] Adjei, A.A. 2001. Ras signaling pathway proteins as therapeutic targets. Curr Pharm Des, 7:1581-1594.

[38] Wu, S.J. and Wu, C. T. 2018. Simulink-based analysis for coupled metabolic systems. Appl. Comput. Intell. Soft Comput. 2018:8075051.

[39] Wu, C. T., Wu, S. J. and Chang, J. Y. 2013. Computational analysis of S-type biological systems. Int. j. eng. res. appl. 3(1): 1976-1987.

[40] Wu, S.J. 2021. Fuzzy approach to regulate S-type biological systems. Int. J. Comput. Appl. 183(10):50-55.